Enabling Scalable VQE Simulation on Leading HPC Systems

Meng Wang
The University of British Columbia
Vancouver, Canada
Pacific Northwest National Laboratory
Richland, USA
mengwang@ece.ubc.ca

Fei Hua
Rutgers University
Piscataway, USA
Pacific Northwest National Laboratory
Richland, USA
huafei90@gmail.com

Chenxu Liu
Pacific Northwest National Laboratory
Richland, USA
chenxu.liu@pnnl.gov

Nicholas Bauman
Pacific Northwest National Laboratory
Richland, USA
nicholas.bauman@pnnl.gov

Karol Kowalski
Pacific Northwest National Laboratory
Richland, USA
karol.kowalski@pnnl.gov

Daniel Claudino
Oak Ridge National Laboratory
Oak Ridge, USA
claudinodc@ornl.gov

Travis Humble
Oak Ridge National Laboratory
Oak Ridge, USA
humblets@ornl.gov

Prashant Nair
The University of British Columbia
Vancouver, Canada
prashantnair@ece.ubc.ca

Ang Li
Pacific Northwest National Laboratory
Richland, USA
ang.li@pnnl.gov

ABSTRACT

Large-scale simulations of quantum circuits pose significant challenges, especially in quantum chemistry, due to the number of qubits, circuit depth, and the number of circuits needed per problem. High-performance computing (HPC) systems offer massive computational capabilities that could help overcome these obstacles. We developed a high-performance quantum circuit simulator called NWQ-Sim, and demonstrated its capability to simulate large quantum chemistry problems on NERSC's Perlmutter supercomputer. Integrating NWQ-Sim with XACC, an open-source programming framework for quantum-classical applications, we have executed quantum phase estimation (QPE) and variational quantum eigensolver (VQE) algorithms for downfolded quantum chemistry systems at unprecedented scales. Our work demonstrates the potential of leveraging HPC resources and optimized simulators to advance quantum chemistry and other applications of near-term quantum devices. By scaling to larger qubit counts and circuit depths, high-performance simulators like NWQ-Sim will be critical for characterizing and validating quantum algorithms before their deployment on actual quantum hardware.

CCS CONCEPTS

• **General and reference → General conference proceedings**; • **Computer systems organization → Quantum computing**; *Heterogeneous (hybrid) systems*; • **Hardware → Quantum computation**; • **Computing methodologies → Quantum mechanic simulation**.

KEYWORDS

Quantum computing, quantum simulation, quantum algorithms, quantum chemistry, variational algorithms, high-performance computing, hybrid quantum-classical computing

ACM Reference Format:
Meng Wang, Fei Hua, Chenxu Liu, Nicholas Bauman, Karol Kowalski, Daniel Claudino, Travis Humble, Prashant Nair, and Ang Li. 2023. Enabling Scalable VQE Simulation on Leading HPC Systems. In *Workshops of The International Conference on High Performance Computing, Network, Storage, and Analysis (SC-W 2023), November 12–17, 2023, Denver, CO, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3624062.3624221

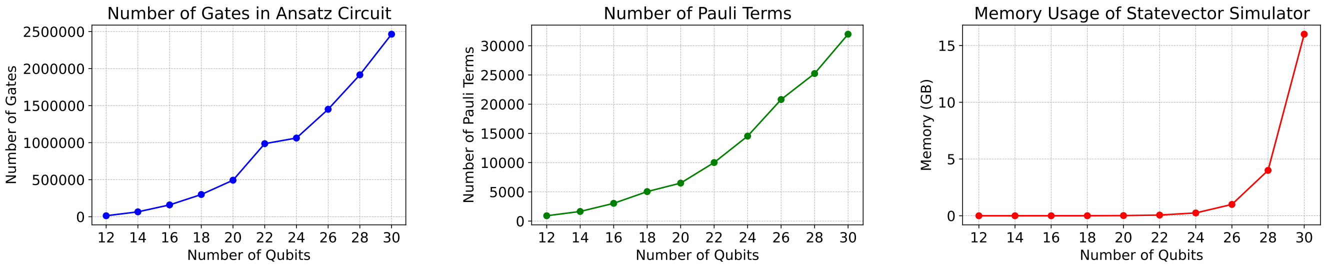
1 INTRODUCTION

Quantum computing, with its potential to efficiently simulate intricate quantum systems, heralds advancements in fields ranging from quantum chemistry to materials science and fundamental physics. Among the myriad of quantum algorithms, the Variational Quantum Eigensolver (VQE) stands out as a pivotal hybrid quantum-classical method, showing promise in unraveling the complexities of the electronic structure problem in quantum chemistry [10].

While the long-term prospects of quantum computing are undoubtedly transformative, the immediate challenges lie in the realm of near-term quantum devices, characterized by noise and other imperfections. Simulating VQE on these classical counterparts is indispensable for verification and performance benchmarking. However, this classical simulation is not without its set of challenges. The overheads, both in terms of computation and memory, scale exponentially with the size of the quantum system under investigation. For even a modestly sized quantum system—quantified by qubits

SC-W 2023, November 12–17, 2023, Denver, CO, USA

ACM ISBN 979-8-4007-0785-8/23/11…$15.00
https://doi.org/10.1145/3624062.3624221



(a) Number of Gates vs. Number of Qubits: the number of gates in the UCCSD ansatz increases with the number of qubits.

(b) Number of Pauli Terms vs. Number of Qubits: the number of Pauli terms in the downfolded (effective) observable of the H_2O (water) molecule using the cc-pV5Z basis set.

(c) Memory Usage vs. Number of Qubits: the memory usage of the state vector (in gigabytes) increases exponentially with the number of qubits.

Figure 1: Scaling overhead of the Variational Quantum Eigensolver. As the number of qubits grows, VQE faces rapidly increasing demands on quantum and classical resources. (a) The gate count of UCCSD ansatz circuits substantially increases, resulting in greater circuit depths and optimization difficulties. (b) The combinatorial growth in Pauli operator measurements imposes higher execution time and sampling costs. (c) The memory for state vector representation scales exponentially, exceeding current classical limits. These overhead trends motivate research into optimized VQE implementations and quantum-classical architectures. Managing VQE's scaling demands is critical for advancing practical applications.

numbering fewer than 20—the depth of the quantum circuit and the sheer number of circuit repetitions necessary for each problem instance present formidable bottlenecks.

Figure 1a underscores this challenge, showing a pronounced increase in the number of gates in the UCCSD ansatz with the growth of qubits. This escalating gate count affects circuit depth and extends the simulation or runtime of the quantum circuit.

The number of Pauli terms, representing the various circuits essential for energy evaluation of the current parameter set, is depicted in Figure 1b. Here, we specifically focus on the downfolded (effective) observable of the H_2O (water) molecule within the confines of the cc-pV5Z basis set. As the graph elucidates, the rise in qubits leads to a substantial surge in Pauli terms, adding to the computational overhead.

Figure 1c further accentuates the memory overhead challenge. To simulate a quantum system classically, the state vector's representation becomes pivotal, and its size grows exponentially with the quantum system's qubit count. This graph paints a stark reality: even for a quantum system with a modest qubit count, the memory requirements, quantified in gigabytes, can quickly outpace the capabilities of conventional computational setups.

While VQE's potential in the realm of quantum simulation is undeniable, it brings with it inherent scalability challenges. Addressing these, especially in the context of classical simulations, remains at the forefront of quantum computing research. To address these scaling challenges, we have developed a comprehensive, end-to-end execution workflow for the VQE algorithm. This workflow is optimized for deployment on state-of-the-art High-Performance Computing (HPC) systems. A high-level overview of the execution flow is shown in Figure 2. The execution flow is composed of three core components, which work together seamlessly:

The execution flow starts with a coupled cluster downfolding [1] process, which downfolds the targeting Hamiltonian to a smaller one that only contains the active Hamiltonian components. Then, the XACC [9], a quantum-classical framework, is used to process the downfolded Hamiltonian to gate-model quantum computing compatible format and execute the VQE algorithm. Quantum circuit simulation is a core part of the overall execution of VQE. This is laid over to NWQ-Sim [6, 7]. NWQ-Sim is a high-performance quantum circuit simulator built for extensive simulations on advanced HPC systems, such as ORNL Summit and NERSC Perlmutter. It supports various execution backends, including CPU, GPU, and multi-node CPU GPU backends. In this work, beyond the ordinary quantum circuit simulation capability, we added a chemistry simulation mode to NWQ-Sim that is specifically optimized for the execution flow of VQE.

In this work, we demonstrate how integrating NWQ-Sim with XACC can enhance the efficiency of VQE simulations on state-of-the-art HPC systems. By employing this VQE execution flow, we conduct simulations on downfolded quantum chemistry systems, achieving significant speedup with smaller instances. While we have not directly demonstrated the capability for larger-scale VQE algorithms, the observed speedup with smaller cases strongly suggests the potential to handle larger instances within a defined time budget. These results underscore the potential benefits of harnessing optimized simulation capabilities with HPC resources, signposting a hopeful path forward for quantum chemistry and other imminent quantum applications.

2 COUPLED CLUSTER DOWNFOLDING

The recently introduced coupled cluster (CC) downfolding techniques [1] provide a powerful framework for systematically reducing the dimensionality of challenging quantum many-body problems. CC downfolding recasts the CC formalism as a renormalization procedure to construct effective Hamiltonians confined to a small, active subspace of the full Hilbert space. This active space encompasses the most important degrees of freedom, while the remaining external space is integrated.

There are two primary variants of the CC downfolding approach:

(1) **Non-Hermitian downfolding**: Based on single-reference CC theory, this approach generates non-Hermitian effective Hamiltonians H_{eff} whose eigenenergies in the active space exactly match the CC energies E_{CC} for the full system:

$$H_{\text{eff}}|\Psi_{\text{CAS}}\rangle = E_{\text{CC}}|\Psi_{\text{CAS}}\rangle \tag{1}$$

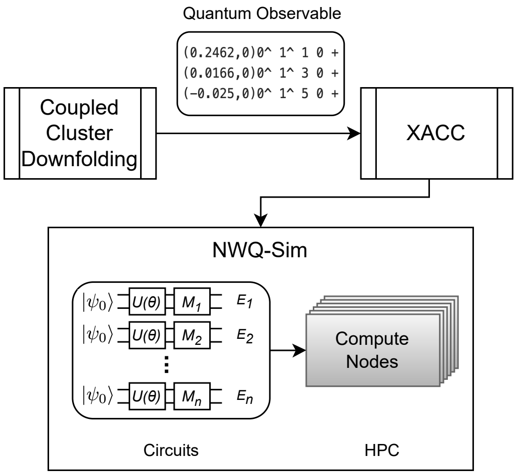
Here, $|\Psi_{\text{CAS}}\rangle$ is the wavefunction confined to the complete active space. The external cluster amplitudes outside the active space are integrated out to renormalize H_{eff}.

(2) **Hermitian downfolding**: Based on unitary CC theory, this approach produces Hermitian downfolded Hamiltonians suitable as inputs for quantum computational algorithms. The anti-Hermitian external cluster operators σ_{ext} are approximately integrated out through systematic finite commutator expansions:

$$H_{\text{eff}} = e^{-\sigma_{\text{ext}}} H e^{\sigma_{\text{ext}}} \approx H + [H, \sigma_{\text{ext}}] + \frac{1}{2}[[H, \sigma_{\text{ext}}], \sigma_{\text{ext}}] + \ldots \tag{2}$$

Higher-order commutators systematically improve the accuracy. Initial applications truncating at second order have demonstrated promise.

A critical theoretical result is the proof of an equivalence theorem showing that solving the standard CC equations is entirely equivalent to solving a set of coupled active space eigenvalue problems defined by downfolded Hamiltonians H_{eff}. This provides a systematic pathway to reduced-scaling CC methods that circumvent exponential scaling costs. It also enables quantum algorithms to leverage modest qubit resources instead of requiring entire configuration interaction (FCI)-scale registers.



```
(0.2462,0)0^ 1^ 1 0 +
(0.0166,0)0^ 1^ 3 0 +
(-0.025,0)0^ 1^ 5 0 +
```

Figure 2: The schematic illustration of the execution flow for the VQE algorithm. It begins with the Coupled Cluster Downfolding, reducing the complexity of the chemistry problem. The downfolded Hamiltonian then feeds into the XACC, generating quantum circuits for execution. Finally, NWQ-Sim, an optimized quantum circuit simulator, conducts large-scale simulations.

Practical applications have focused on using downfolding to compress quantum chemistry simulations, particularly when combining with variational quantum eigensolver (VQE) algorithms. For example, downfolded Hamiltonians based on double commutator expansions give highly accurate potential energy surfaces for breaking chemical bonds, reducing active space errors by orders of magnitude compared to bare Hamiltonian diagonalization [1].

Downfolding provides a flexible framework for exploiting sparsity and locality in quantum simulations. Flow-based algorithms built from coupled H_{eff} eigenvalue problems can target only the most significant degrees of freedom. This avoids the need for explicit global wavefunctions. Overall, CC downfolding delivers a systematic avenue to address the pressing challenges of complexity and scalability across quantum chemistry, materials science, and beyond. Ongoing work is focused on integrating downfolding with reduced-scaling CC methods, analyzing the breakdown of locality approximations, and applying downfolding to periodic solid-state systems.

3 XACC FRAMEWORK

The eXtreme-scale ACCelerator (XACC) is a cutting-edge programming framework that is a linchpin for the seamless integration and execution of quantum-classical algorithms [9]. A standout feature of XACC is its hardware-agnostic nature, empowering researchers to compile quantum programs for a wide range of supported backends. This includes physical quantum processors and simulators, establishing a pathway for high-level algorithms to be initially tested on simulators and subsequently transitioned to actual quantum hardware when ready.

3.1 XACC in Quantum Chemistry

In the quantum chemistry domain, XACC emerges as a pivotal tool, especially for implementing hybrid algorithms like the Variational Quantum Eigensolver (VQE). The typical workflow encapsulating this process is as follows:

(1) XACC takes a molecular Hamiltonian H combined with a chosen ansatz quantum circuit $U(\theta)$ as its primary input.
(2) The input Hamiltonian transforming quantum logic gates and circuits, which are for expectation values, represented as
$U^{\dagger}(\theta)HU(\theta)$.
(3) A specified quantum backend, in our case, the NWQ-Sim simulator, undertakes the execution of these circuits. The outcome is the derivation of expectation values corresponding to particular parameter sets denoted by θ.
(4) A classical optimizer then processes these expectations. The core objective is to refine and optimize θ, thereby minimizing the energy, culminating in the finalization of the VQE process.
(5) This procedure undergoes repetitive cycles until convergence towards an optimized minimum energy.

3.2 Quantum-Classical Co-Processing in XACC

Central to XACC's functionality is its execution model premised on quantum-classical co-processing. Here, quantum circuits, either on NISQ devices or simulators, furnish the much-needed expectation values that feed into the classical optimization procedure. What sets XACC apart is its intrinsic flexibility. It swiftly integrates novel quantum backends, state-of-the-art algorithms, and cutting-edge chemistry methodologies.

The ongoing research landscape is buzzing with endeavors to amplify XACC's capabilities. The goal is clear: to facilitate scalable and pinpoint-accurate quantum chemistry simulations, harmonizing the strengths of both classical and quantum hardware.

4 NWQ-SIM OPTIMIZATIONS FOR VQE

The NWQ-Sim simulator provides high-performance quantum circuit simulation capabilities by leveraging massively parallel GPU hardware architectures [6]. In particular, NWQ-Sim is designed to maximize the benefits of GPUs for simulating quantum circuits, including:

- Maintaining quantum state representations in fast GPU memory to minimize latency
- Distributing parallel simulation of gates and state updates across thousands of GPU cores
- Batching independent calculations together to enhance GPU utilization

This GPU-centric architecture enables NWQ-Sim to rapidly simulate state preparation, gate applications, and measurements for quantum circuits.

Crucially, NWQ-Sim also incorporates optimizations targeting more efficient and accurate simulations of the variational quantum eigensolver (VQE) algorithm.

4.1 Caching and Reusing Post-Ansatz States

At the heart of the VQE is a need to measure quantum states in various bases to evaluate expectation values accurately. But first, let's establish what we mean by different *basis measurements* and why they are crucial in quantum computing.

4.1.1 Basis Measurements in Quantum Computing. In classical computing, we think of bits: 0s and 1s. In quantum computing, however, qubits can exist in a superposition of states. The 'basis' refers to the set of states against which a qubit's state is measured. The most familiar basis is the computational (or Z-basis), where measurement yields either a $|0\rangle$ or $|1\rangle$.

However, for many quantum algorithms, measuring qubits in different bases is essential. For instance, X and Y bases are other commonly used ones.

4.1.2 Handling Different Bases. In quantum computing, the most common measurement bases are the Pauli bases: X, Y, and Z. These bases are essential for quantum computations and play a pivotal role in determining the outcomes of quantum measurements.

The Z-basis is the standard computational basis, and if we prepare a qubit in the state $|0\rangle$ or $|1\rangle$ and measure it, the outcome will correspond to one of these states.

If we want to measure a qubit initially prepared in a Z-basis state for the X-basis, we apply a Hadamard gate (H) before measurement. This gate transforms the qubit from the Z-basis to the X-basis. The outcomes here are represented by $|+\rangle$ (which is a superposition of $|0\rangle$ and $|1\rangle$) and $|-\rangle$ (a superposition with a relative phase).

For the Y-basis, a combination of Pauli-X and Pauli-Z gates (specifically, a S† gate followed by a Hadamard gate) will transform the qubit from the Z-basis to the Y-basis. The outcomes in the Y-basis are typically represented by the states $|y+\rangle$ and $|y-\rangle$, which are complex conjugates of each other.

4.1.3 Computing Expectation Values in VQE. To compute the expectation value of a Hamiltonian H in the state $|\psi(\theta)\rangle$, the expression is:

$$\langle H\rangle = \langle\psi(\theta)|\, H\, |\psi(\theta)\rangle \tag{3}$$

Often, H is a sum of terms acting on different bases. Hence, to evaluate $\langle H\rangle$, the state $|\psi(\theta)\rangle$ may need to be measured in several different bases.

For example, consider a toy Hamiltonian:

$$H = Z \otimes Z + X \otimes X \tag{4}$$

For a 2-qubit system, this Hamiltonian comprises terms in both Z and X bases. If our state is $|\psi(\theta)\rangle$ and we're measuring in the Z-basis, we don't need any additional operations. However, we'd apply a Hadamard gate to each qubit to measure the second term to switch to the X-basis before measurement. These repeated measurements in various bases necessitate multiple executions of the ansatz circuit $U(\theta)$ to prepare the state $|\psi(\theta)\rangle$.

4.1.4 Efficiency through Caching. Recognizing the recurring need to reapply the ansatz, NWQ-Sim introduces a novel approach. After the initial simulation of

$$|\psi(\theta)\rangle = U(\theta)\, |0\rangle^{\otimes n} \tag{5}$$

the resulting state's amplitudes are cached in the GPU memory. This pre-computed state now serves as a ready reference for all subsequent measurements, eliminating the need for repeated ansatz executions.

However, quantum states can have large memory requirements, especially for systems with many qubits. Thus, if the GPU's memory capacity surpasses, NWQ-Sim seamlessly transitions to CPU memory storage. While this might introduce some performance delays compared to the rapid GPU access times, it ensures scalability and continuity of the simulation process.

In essence, by caching post-ansatz states, NWQ-Sim addresses the significant challenge posed by the VQE's multi-basis measurement needs, presenting a practical solution to a complex quantum problem.

4.2 Direct Expectation Value Calculation

When working with quantum systems, obtaining accurate information about the system's properties often requires calculating expectation values. Traditionally, these values have been estimated using a sampling approach, wherein the quantum system is measured multiple times, and the results are averaged to infer the expectation value. However, this method can be computationally intensive and might not provide exact values due to statistical fluctuations.

4.2.1 Traditional Sampling vs. Direct Calculation. The quantum system is prepared and measured repeatedly in the traditional sampling method. Each measurement collapses the quantum state to a particular basis state, and the outcomes are collected over many runs. These outcomes are then statistically processed to estimate the expectation value of an operator.

NWQ-Sim, on the other hand, offers a more direct approach. Instead of relying on statistical averages, it calculates the exact expectation value using the full knowledge of the quantum state.

4.2.2 Mathematical Insight. For a clearer understanding of direct expectation value calculation, let's consider the simple 2-qubit toy Hamiltonian from Equation 4. This Hamiltonian consists of two terms: one that operates in the Z-basis and another in the X-basis. This Hamiltonian can be represented as a matrix in the computational basis. Each term of the Hamiltonian would have its own matrix representation. The overall matrix H is the sum of these individual matrices. Let's focus on the $Z \otimes Z$ term for simplicity. Its matrix representation is:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{6}$$

Imagine we have a quantum state $|\psi\rangle$ prepared using some ansatz circuit. This state can be expressed in the computational (Z) basis as:

$$|\psi\rangle = \sum_{x=0}^{3} c_x |x\rangle \tag{7}$$

Where the coefficients c_x are the amplitudes of the state and $|x\rangle$ can be one of the four basis states $|00\rangle$, $|01\rangle$, $|10\rangle$, or $|11\rangle$ for a 2-qubit system. With the state and operator expressed in the same basis, the expectation value for the $Z \otimes Z$ term can be computed using:

$$\langle Z \otimes Z \rangle = \sum_{x=0}^{3} \sum_{y=0}^{3} c_x^* (Z \otimes Z)_{xy} c_y \tag{8}$$

A similar procedure applies for the $X \otimes X$ term, but the state first needs to be transformed to the X-basis (using Hadamard gates) before this computation. The total expectation value for H is then the sum of the expectations for each term.

4.2.3 Efficiency through Parallelization. While the direct calculation method is deterministic, it can be computationally intensive for large quantum systems due to the double summation over all possible basis states. However, this challenge is tackled head-on by NWQ-Sim's architecture.

The nested sums in the above equation can be efficiently parallelized, especially since each term in the sum is independent of the others. NWQ-Sim divides these calculations across thousands of GPU cores, allowing simultaneous computation of multiple terms. The algorithm maximizes the computational throughput by batching iterations over the $|x\rangle$ states.

As the system size grows, the advantages of this method become increasingly apparent. For large-scale quantum simulations, the direct expectation calculation in NWQ-Sim significantly outpaces the traditional sampling approach, chiefly due to the parallel processing capabilities of modern GPU hardware.

The direct expectation value calculation in NWQ-Sim offers a blend of precision and efficiency. By replacing traditional probabilistic methods with deterministic calculations and fully leveraging GPU acceleration, NWQ-Sim ensures rapid and accurate evaluations of quantum systems, paving the way for future more complex and insightful quantum simulations.

4.3 Gate Fusion

In quantum circuit simulations, especially those involving many gates, there is an opportunity to optimize the computation by fusing multiple consecutive gates into a single gate. This technique, called gate fusion, can provide significant computational advantages in simulation over executing each gate separately.

4.3.1 Gate Fusion in Simulation. While quantum hardware devices may have constraints regarding which gates can be fused, especially due to available basis gates and qubit connections, a simulator, like NWQ-Sim, is not limited by such physical constraints. Any sequence of consecutive gates acting on the same qubit(s) can be fused in simulation. The resulting fused gate is represented as a matrix, computed by taking the matrix product of the individual gate matrices in their sequential order.

NWQ-Sim natively supports single and two-qubit gates. To strike a balance between optimization and computational feasibility, NWQ-Sim fuses gates only up to two qubits. This design decision is rooted in computational efficiency. Consider a scenario where we have four gates acting on four individual qubits. If we fuse all of these gates into a single gate, we will have a matrix of dimensions $2^4 \times 2^4$. On the other hand, if we choose to fuse them into two pairs of gates, each acting on two qubits, we get two matrices, each of size $2^2 \times 2^2$. The combined dimensionality of the smaller matrices is much more manageable than that of the larger matrix.

Given the exponential growth of matrix dimensions with the number of qubits, the computational cost to manipulate and apply larger matrices, especially considering the parallel processing capabilities of GPUs, can quickly become infeasible. By limiting gate fusion to produce only up to 2-qubit gates, NWQ-Sim ensures an optimal trade-off between reduced operations and computational complexity.

Overall, gate fusion offers substantial performance improvements. By reducing the total number of operations, NWQ-Sim can execute simulations more rapidly, leveraging the parallelism of modern GPUs to handle the matrix operations efficiently.

These GPU-focused architectures and VQE-specific optimizations enable NWQ-Sim to maximize performance on quantum chemistry simulations relevant to quantum computing. Ongoing work is expanding the capabilities to larger qubit counts and circuit depths.

5 RESULTS

In this section, we present results from using NWQ-Sim to simulate variational quantum algorithms. The performance improvements enabled by the NWQ-Sim workflow are quantified through comparative simulations. Furthermore, additional results demonstrate the accuracy and effectiveness of NWQ-Sim for practical quantum computational tasks. Together, these two categories of simulations provide evidence of NWQ-Sim's capabilities for efficiently simulating intermediate-scale quantum circuits and algorithms relevant to near-term applications.

5.1 Caching Post-Ansatz State

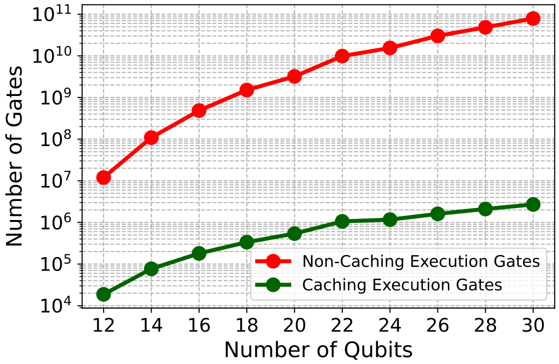


Figure 3: Comparison of the number of gates required for VQE energy evaluation using non-caching vs. caching execution. The y-axis shows the number of gates on a logarithmic scale, and the x-axis represents the number of qubits. Caching the post-ansatz state leads to a significant reduction in the number of required gates.

The results in Figure 3 highlight the significant reduction in gate requirements enabled by caching the post-ansatz state. Without caching, the energy evaluation circuit must repeatedly prepare the ansatz state followed by basis transformation gates for each term in the Hamiltonian. This incurs a high gate cost on the order of 10^7 to $10^1$1 gates. By contrast, caching allows the post-ansatz state to be prepared only once. Subsequent basis transformations and measurements to compute the partial expectations can be applied with just 10^4 to 10^6 additional gates. Thus, caching provides a gate savings of approximately 3 to 5 orders of magnitude. This substantial efficiency improvement is especially impactful as system size increases. Caching fundamentally changes the scaling behavior of VQE by avoiding redundant state preparations. These findings demonstrate the value of caching techniques for reducing the quantum resources needed for VQE computations.

5.2 Gate Fusion

The gate count reductions in Figure 4 highlight the optimization provided by gate fusion in NWQ-Sim. Across 4, 6, and 8-qubit UCCSD ansatz circuits, fusing neighboring single and two-qubit gates consistently decreases the number of operations by over 50%. For example, the 8-qubit circuit gate count reduces from 10,809 to 5,208 gates after fusion, an approximately 52% improvement. The 6-qubit circuit experiences a comparable drop from 4,158 to 1,954 gates. Even small can circuits benefit from significant gate savings via fusion.

By consolidating quantum operations during simulation, gate fusion reduces the number of discrete gate applications. This decreases the computational resources required to simulate circuit

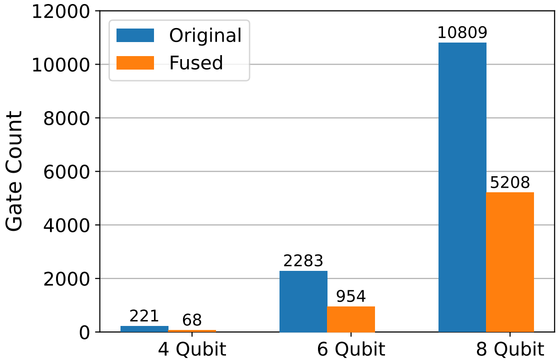


Figure 4: Comparison of gate counts for 4, 6, and 8-qubit UCCSD ansatz circuits before and after gate fusion.

execution. The substantial gate count reductions accelerate the simulation of quantum circuits within NWQ-Sim. As circuit sizes scale up for simulating larger quantum chemical systems, gate fusion will become increasingly crucial for feasible simulations.

5.3 Adapt-VQE Execution of Water Molecule

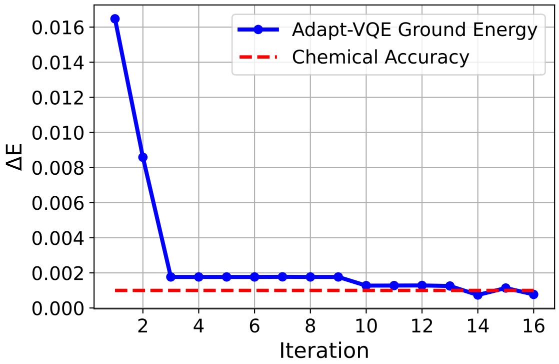


Figure 5: Convergence of adaptive VQE for computing the ground state energy of a downfolded 6-orbital H_2O molecule. The energy difference ΔE from the true ground state is plotted against the VQE iteration. At around 16 iterations, adaptive VQE reaches the 1 milli-hartree chemical accuracy.

Adaptive VQE [4, 16, 17] is an extension of the standard VQE algorithm that iteratively improves the ansatz by adapting the circuit structure based on measurement outcomes. Compared to fixed VQE ansatzes, adaptive VQE circuits can better approximate the ground state with fewer parameters and shallower circuits. However, adaptively growing the ansatz increases the classical optimization burden. In this work, we apply adaptive VQE to efficiently compute the ground state of a 6-orbital H_2O molecule simulated on 12 qubits.

As shown in Figure 5, adaptive VQE is able to converge the ground state energy to within 1 milli-hartree chemical accuracy in just 16 iterations. Furthermore, each adaptive iteration increases the ansatz depth by only 1 layer. This demonstrates efficient ground-state convergence for the adaptive VQE approach. These results validate the proposed workflow for efficiently finding the ground state energy for quantum chemical simulations.

6 DISCUSSION

6.1 Related works

A large number of prior works have looked at optimization for the execution/simulation of the Variational Quantum Eigensolver (VQE) algorithm. Hardware-efficient ansatz designs [5] have been explored to reduce the circuit depth and number of parameters. Cafqa [11] proposed using an efficient classical simulation of Clifford circuits to expedite the VQE optimization. Adaptive VQE methods [4, 16, 17] have been developed to significantly reduce the depth of the ansatz circuit.

Beyond those, general quantum circuit optimizations are also proposed to reduce gate count, circuit depth, etc. Sabre [8] is a compiler that optimizes and maps quantum circuits to IBM Q devices. It performs circuit rewriting using gate cancellation, commutation, and fusion. Siraichi et al.[14] developed qubit mapping techniques to minimize SWAP gates needed for circuits on hardware with limited qubit connectivity. Faster Schrödinger-Feynman algorithm[3] fuses sequences of gates to avoid recomputing temporary intermediate states.

In addition to VQE-specific optimizations, recent works have started investigating larger-scale quantum simulations relevant to quantum chemistry. Cao et al.[2] provided a comprehensive review of the progress and challenges of quantum computational chemistry in the NISQ era. They discussed required theoretical foundations, algorithmic developments, as well as the outlook of near-term quantum devices for chemical simulations. Along this line, Shang et al.[12] explored tensor network simulation of quantum chemistry problems on a supercomputer.

These works have targeted various performance improvements for VQE and quantum circuits and provide valuable insights on enhancing VQE execution, which can be integrated with NWQ-Sim's backend.

6.2 Future imrprovements

There are several promising directions to further accelerate VQE simulation in the future. One area is batch execution, where multiple VQE iterations or circuits could be simulated simultaneously. Within a GPU, multiple compute kernels could be launched concurrently to utilize more cores [13]. Across multiple GPUs, independent circuits can be distributed to enhance parallelism.

Another major bottleneck is the classical optimization procedure. The number of tunable parameters in VQE ansatzes often ranges from tens to thousands. This creates a vast search space that is computationally expensive for classical optimizers to traverse. Specialized optimization algorithms that leverage problem structure could help mitigate this overhead. Incremental optimization is another approach where the optimal parameters from the previous executions can be used to warm start the next round.

Additionally, there are opportunities to optimize the co-design of classical and quantum resources for VQE. Hybrid algorithm-architecture techniques like EQC [15] that efficiently distribute computations across available quantum and classical hardware can lead to higher throughput and faster convergence. As quantum devices continue to scale up, developing holistic co-designs that coordinate the classical and quantum components will become increasingly important.

7 CONCLUSION

In conclusion, this work demonstrates an integrated workflow for efficient VQE simulation on high-performance computing systems. By combining coupled cluster downfolding, the XACC quantum programming framework, and the optimized NWQ-Sim simulator, we establish an end-to-end pipeline for practical VQE applications. NWQ-Sim's ability to cache the post-ansatz states avoids redundant circuit executions during VQE energy evaluation. Through selective gate fusion and direct expectation value calculation, NWQ-Sim further boosts the performance of simulating quantum circuits. Our comparative simulations quantify the substantial gate count reductions and accuracy achieved by NWQ-Sim. In addition, we showcase the application of this workflow by computing the ground state energy of a water molecule using adaptive VQE.

This research underscores the benefits of leveraging HPC resources and optimized simulators to tackle the scaling demands of VQE and advance quantum chemistry simulations. While directly demonstrating larger-scale VQE capabilities remains an ongoing effort, the optimizations presented already accelerate smaller instances. As quantum systems continue to grow, high-performance simulators like NWQ-Sim will be instrumental in verifying quantum algorithms before deployment on real hardware.

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Quantum Science Center. This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231 using NERSC award ERCAP0023224 and ERCAP0023053. This material is also supported by the National Research Council Canada grant AQC 003 and by the Natural Sciences and Engineering Research Council of Canada (NSERC) [funding reference number RGPIN-2019-05059].

REFERENCES

[1] Nicholas P. Bauman and Karol Kowalski. 2021. Coupled Cluster Downfolding Theory: towards efficient many-body algorithms for dimensionality reduction of composite quantum systems. arXiv:2111.03215 [quant-ph]

[2] Yudong Cao, Jonathan Romero, Jonathan P Olson, Matthias Degroote, Peter D Johnson, Mária Kieferová, Ian D Kivlichan, Tim Menke, Borja Peropadre, Nicolas PD Sawaya, et al. 2019. Quantum chemistry in the age of quantum computing. *Chemical reviews* 119, 19 (2019), 10856–10915.

[3] A. Fatima and I. L. Markov. 2021. Faster Schrödinger-style simulation of quantum circuits. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE Computer Society, Los Alamitos, CA, USA, 194–207. https://doi.org/10.1109/HPCA51647.2021.00026

[4] Harper R. Grimsley, Sophia E. Economou, Edwin Barnes, and Nicholas J. Mayhall. 2019. An adaptive variational algorithm for exact molecular simulations on a

quantum computer. *Nature Communications* 10, 1 (jul 2019). https://doi.org/10.1038/s41467-019-10988-2

[5] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta. 2017. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature* 549, 7671 (sep 2017), 242–246. https://doi.org/10.1038/nature23879

[6] Ang Li, Bo Fang, Christopher Granade, Guen Prawiroatmodjo, Bettina Hein, Martin Rotteler, and Sriram Krishnamoorthy. 2021. SV-Sim: Scalable PGAS-based State Vector Simulation of Quantum Circuits. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*.

[7] Ang Li, Omer Subasi, Xiu Yang, and Sriram Krishnamoorthy. 2020. Density Matrix Quantum Circuit Simulation via the BSP Machine on Modern GPU Clusters. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*.

[8] Gushu Li, Yufei Ding, and Yuan Xie. 2019. Tackling the Qubit Mapping Problem for NISQ-Era Quantum Devices. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems* (Providence, RI, USA) *(ASPLOS '19)*. Association for Computing Machinery, New York, NY, USA, 1001–1014. https://doi.org/10.1145/3297858.3304023

[9] Alexander J McCaskey, Dmitry I Lyakh, Eugene F Dumitrescu, Sarah S Powers, and Travis S Humble. 2020. XACC: a system-level software infrastructure for heterogeneous quantum–classical computing. *Quantum Science and Technology* 5, 2 (feb 2020), 024002. https://doi.org/10.1088/2058-9565/ab6bf6

[10] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. 2014. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications* 5, 1 (jul 2014). https://doi.org/10.1038/ncomms5213

[11] Gokul Subramanian Ravi, Pranav Gokhale, Yi Ding, William Kirby, Kaitlin Smith, Jonathan M. Baker, Peter J. Love, Henry Hoffmann, Kenneth R. Brown, and Frederic T. Chong. 2022. CAFQA: A Classical Simulation Bootstrap for Variational Quantum Algorithms. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 1* (Vancouver, BC, Canada) *(ASPLOS 2023)*. Association for Computing Machinery, New York, NY, USA, 15–29. https://doi.org/10.1145/3567955.3567958

[12] Honghui Shang, Li Shen, Yi Fan, Zhiqian Xu, Chu Guo, Jie Liu, Wenhao Zhou, Huan Ma, Rongfen Lin, Yuling Yang, Fang Li, Zhuoya Wang, Yunquan Zhang, and Zhenyu Li. 2022. Large-Scale Simulation of Quantum Computational Chemistry on a New Sunway Supercomputer. In *SC22: International Conference for High Performance Computing, Networking, Storage and Analysis*. 1–14. https://doi.org/10.1109/SC41404.2022.00019

[13] S.-Kazem Shekofteh, Hamid Noori, Mahmoud Naghibzadeh, Holger Fröning, and Hadi Sadoghi Yazdi. 2020. cCUDA: Effective Co-Scheduling of Concurrent Kernels on GPUs. *IEEE Transactions on Parallel and Distributed Systems* 31, 4 (2020), 766–778. https://doi.org/10.1109/TPDS.2019.2944602

[14] Marcos Yukio Siraichi, Vinícius Fernandes dos Santos, Caroline Collange, and Fernando Magno Quintão Pereira. 2018. Qubit allocation. In *Proceedings of the 2018 International Symposium on Code Generation and Optimization*. 113–125.

[15] Samuel Stein, Nathan Wiebe, Yufei Ding, Peng Bo, Karol Kowalski, Nathan Baker, James Ang, and Ang Li. 2022. EQC: Ensembled Quantum Computing for Variational Quantum Algorithms. In *Proceedings of the 49th Annual International Symposium on Computer Architecture* (New York, New York) *(ISCA '22)*. Association for Computing Machinery, New York, NY, USA, 59–71. https://doi.org/10.1145/3470496.3527434

[16] Ho Lun Tang, V.O. Shkolnikov, George S. Barron, Harper R. Grimsley, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. 2021. Qubit-ADAPT-VQE: An Adaptive Algorithm for Constructing Hardware-Efficient Ansätze on a Quantum Processor. *PRX Quantum* 2, 2 (apr 2021). https://doi.org/10.1103/prxquantum.2.020310

[17] Linghua Zhu, Ho Lun Tang, George S. Barron, F. A. Calderon-Vargas, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. 2022. An adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer. arXiv:2005.10258 [quant-ph]