# Simulating Quantum Systems with NWQ-Sim on HPC

In-Saeng Suh
*National Center for Computational Sciences*
*Oak Ridge National Laboratory*
Oak Ridge, Tennessee, USA
suhi@ornl.gov

Ang Li
*Physical and Computational Sciences Directorate*
*Pacific Northwest National Laboratory*
Richland, Washington, USA
ang.li@pnnl.gov

***Abstract*—NWQ-Sim is a cutting-edge quantum system simulation environment designed to run on classical multi-node, multi-CPU/GPU heterogeneous HPC systems. In this work, we provide a brief overview of NWQ-Sim and its implementation in simulating quantum circuit applications, such as the transverse field Ising model. We also demonstrate how NWQ-Sim can be used to examine the effects of errors that occur on real quantum devices, using a combined device noise model. Moreover, NWQ-Sim is particularly well-suited for implementing variational quantum algorithms where circuits are dynamically generated. Therefore, we also illustrate this with the variational quantum eigensolver (VQE) for the Ising model. In both cases, NWQ-Sim's performance is comparable to or better than alternative simulators. We conclude that NWQ-Sim is a useful and flexible tool for simulating quantum circuits and algorithms, with performance advantages and noise-aware simulation capabilities.**

***Index Terms*—Quantum Simulators, Noise Model, Variational Quantum Eigensolver, Ising Model**

## I. Introduction

In recent years, Noisy-Intermediate-Scale-Quantum (NISQ) [1] based quantum computing (QC) has made significant progress [2], [3]. However, despite these advances, the wider QC community still heavily relies on classical machine simulations for the development and validation of quantum algorithms. This is particularly true for the promising quantum variational algorithms (VQA) [4], including variational quantum eigensolvers (VQE) [5], [6], quantum neural networks (QNN) [7], [8], and quantum approximated optimization algorithms (QAOA) [9], [10].

There are several reasons for this dependence on classical simulations. Firstly, the limited availability of quantum computing resources such as those provided by IBMQ or Azure Quantum means that a large number of users have to share them, resulting in long waiting times for quantum resource allocation. Secondly, since parameterized variational circuits are typically deep, as a result, the depth of these circuits can easily exceed the maximum gate allowance of NISQ devices due to their short coherence time. Finally, current quantum computers are not quantum-error-correction (QEC) protected [11], [12], resulting in high error rates. Consequently, simulations are necessary for validating quantum algorithms and debugging circuits [13], [14].

A variety of quantum circuit simulators on classical computers have been proposed [15]. However, the majority of these simulators focus on logical qubits within an ideal isolated system setting, where complete gate fidelity can be guaranteed, rather than physical qubits within a practical, open environment that is subject to inevitable noise. Designing an efficient and scalable QC simulator on classical high-performance computing (HPC) systems presents significant challenges. In the context of today's heterogeneous supercomputers, the performance of an application largely depends on how effectively accelerators, such as GPUs, are utilized. However, users still desire a unified and easy-to-use programming interface that is suitable for their domains.

In this context, we introduce the NWQ-Sim simulator which overcomes other simulator's challenges and its implementation in simulating quantum circuits and VQE for the time-dependent transverse field Ising model. We also use a device noise model to examine the effects of errors that occur on real quantum devices. Finally, we compare the performance of the NWQ-Sim and IBM Qiskit simulators.

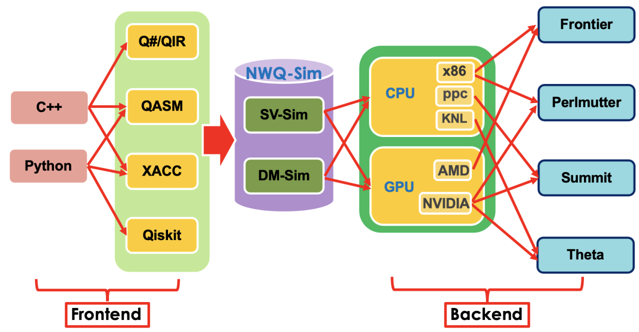


Fig. 1. The framework of the NWQ-Sim simulator

## II. NWQ-Sim simulator

NWQ-Sim is a quantum system simulation environment that runs on classical multi-node, multi-CPU/GPU heterogeneous HPC systems. It combined two different types of simulators - a state vector simulator SV-Sim for high performance quantum simulation and a density matrix simulator DM-Sim for noise-aware simulation. It supports C++, Python, Q#/QIR, Qiskit, QASM, XACC [16] as the frontends, and x86/PPC CPU, NVIDIA/AMD GPU as beckends. Fig. 1 shows the NWQ-Sim framework.

```python
#==============================================
from qiskit_nwqsim_provider import NWQSimProvider
#----------------------------------------------
nwqsim = NWQSimProvider('SVSimSimulator')
backend = nwqsim.backends['svsim_gpu']
#==============================================
```

Fig. 2. Importing the SV-Sim simulator from NWQ-Sim

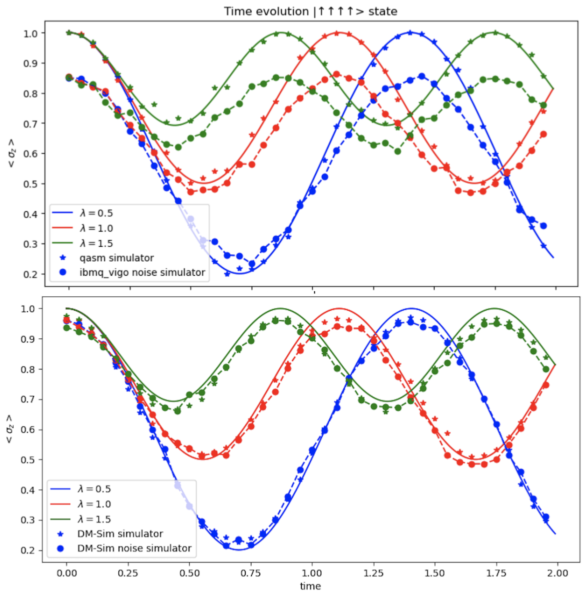


Fig. 3. Time evolution simulation of transverse magnetization, $\langle\sigma_z\rangle$, for the state $|\uparrow\uparrow\uparrow\uparrow\rangle$ of a $n = 4$ Ising spin chain. Solid line represents the exact result in comparison with the experimental simulations on Qiskit Aer and NWQ-Sim noisy simulators represented by scatter points.

To run a quantum circuit on NWQ-Sim, first one will need to import NWQSimProvider as shown in Fig. 2 and then specify the backend to use. The backend corresponds to the simulator that we want to use and is defined by the provider.

## III. Quantum Simulations with a Noisy Model

As a first use case, a simulation was employed to model the magnetization and temporal variations of a spin chain governed by Ising-type interactions. Here, we will use these precise outcomes as a reference point for our quantum simulation of the Transverse Field Ising Model (TFIM). For simple experiments, let's consider $n = 4$ spin chain. We optimize the quantum circuit and implement it on IBM Quantum Qiskit and NWQ-Sim simulators.

Fig. 3 shows the time evolution of transverse magnetization $\langle\sigma_z\rangle$. We also use the device backend noise model to do noisy simulations of the quantum circuits to investigate the effects of errors which occur on real quantum devices in both Qiskit Aer and NWQ-Sim. The noise model includes depolarization noise, thermal relaxation noise, and measurement noise. We use real noise data for an IBM Quantum device using the data for ibmq_vigo stored in Qiskit Terra [17].

In the VQE simulation experiments, we use the COBYLA method as a minimizer [18]. By comparing the noise VQE simulations with IBM Qiskit and NWQ-Sim simulators, we can further investigate the effect of noise on real quantum

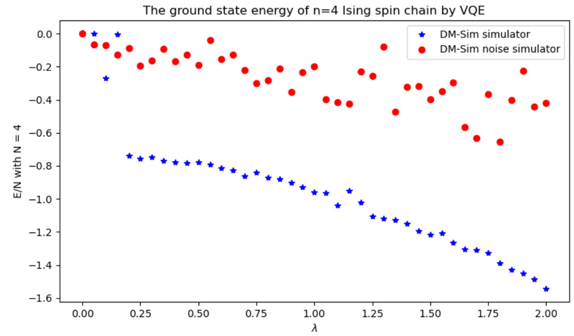


Fig. 4. VQE simulation in no-noise (star) and noise (solid circle) with NWQ-Sim

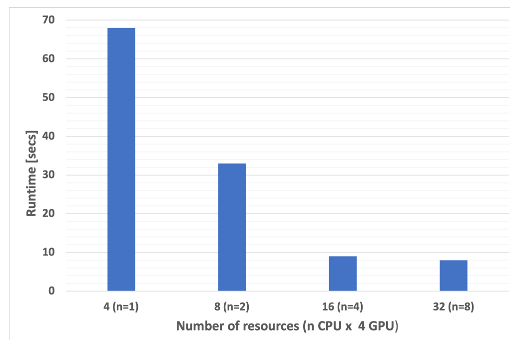


Fig. 5. QASMBench [22] with NWQ-Sim on OLCF Summit

computers. Fig. 5 shows QASM benchmark result for NWQ-Sim simulator on the Summit supercomputer. We perform all simulations on the Frontier [19] and Summit supercomputer at ORNL [20] and the Perlmutter supercomputer at NERSC [21].

## IV. Conclusion

NWQ-Sim is a state-of-the-art quantum system simulation environment that offers advanced features for simulating quantum circuits and algorithms on classical multi-node, multi-CPU/GPU heterogeneous HPC systems. Its versatile simulation infrastructure, including the state vector simulator (SV-Sim) and the density matrix simulator (DM-Sim), provides high-performance quantum simulation as well as noise-aware simulation. NWQ-Sim's efficient inter-communication management capabilities make it particularly suitable for large-scale simulations.

In this work, we have provided an overview of NWQ-Sim and its implementation in simulating quantum circuit applications, showcasing its ability to examine the effects of errors on real quantum computers using a device noise model. For deep quantum circuit simulations, NWQ-Sim simulator shows comparable or better performance to the IBM Qiskit simulator under specific environments. Furthermore, NWQ-Sim's performance in implementing variational quantum algorithms, such as the variational quantum eigensolver (VQE) for the Ising model, gives better performance than other simulators. For example, it shows about $50\%$ faster on the GPU-based supercomputer.

## Acknowledgment

The research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration. 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 (QSC). This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. 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 Awards ERCAP0023224 and ERCAP0023053.



## References


[1] John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018.

[2] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[3] Sergio Boixo, Sergei V Isakov, Vadim N Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J Bremner, John M Martinis, and Hartmut Neven. Characterizing quantum supremacy in near-term devices. *Nature Physics*, 14(6):595–600, 2018.

[4] Quantum computing: Qubit and entanglement. *Review of the Universe: Structures, Evolutiuons, Observations, and Theories*.

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

[6] Jonathan Romero, Ryan Babbush, Jarrod R McClean, Cornelius Hempel, Peter J Love, and Alán Aspuru-Guzik. Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz. *Quantum Science and Technology*, 4(1):014008, 2018.

[7] Kerstin Beer, Dmytro Bondarenko, Terry Farrelly, Tobias J Osborne, Robert Salzmann, Daniel Scheiermann, and Ramona Wolf. Training deep quantum neural networks. *Nature communications*, 11(1):808, 2020.

[8] Samuel A Stein, Betis Baheri, Daniel Chen, Ying Mao, Qiang Guan, Ang Li, Shuai Xu, and Caiwen Ding. Qucla
ssi: A hybrid deep neural network architecture based on quantum state fidelity. *Proceedings of Machine Learning and Systems*, 4:251–264, 2022.

[9] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

[10] Gian Giacomo Guerreschi and Anne Y Matsuura. Qaoa for max-cut requires hundreds of qubits for quantum speed-up. *Scientific reports*, 9:1–7, 2019.

[11] Rami Barends, Julian Kelly, Anthony Megrant, Andrzej Veitia, Daniel Sank, Evan Jeffrey, Ted C White, Josh Mutus, Austin G Fowler, Brooks Campbell, et al. Superconducting quantum circuits at the surface code threshold for fault tolerance. *Nature*, 508(7497):500–503, 2014.

[12] Daniel A Lidar and Todd A Brun. Quantum error correction. In *Quantum error correction*. Cambridge university press, 2013.

[13] Jonathan Carter, David Dean, Greg Hebner, Jungsang Kim, Andrew Landahl, Peter Maunz, Raphael Pooser, Irfan Siddiqi, and Jeffrey Vetter. Ascr report on a quantum computing testbed for science. Technical report, USDOE Office of Science (SC), Washington, DC (United States). Advanced . . . , 2017.

[14] Tyson Jones, Anna Brown, Ian Bush, and Simon C Benjamin. Quest and high performance simulation of quantum computers. *Scientific reports*, 9(1):1–11, 2019.

[15] List of qc simulators. https://www.quantiki.org/wiki/list-qc-simulators.

[16] Alexander J McCaskey, Dmitry I Lyakh, Eugene F Dumitrescu, Sarah S Powers, and Travis S Humble. Xacc: a system-level software infrastructure for heterogeneous quantum–classical computing. *Quantum Science and Technology*, 5(2):024002, 2020.

[17] Qiskit: An open-source framework for quantum computing. htts://qiskit.org.

[18] Lohit Potnuru. Finding the ground state of the transverse Ising model. https://github.com/LohitPotnuru/TransverseIsingModelQiskit.git, 2021.

[19] Frontier at ORNL. https://www.olcf.ornl.gov/frontier/.

[20] Summit at ORNL. https://www.olcf.ornl.gov/summit/.

[21] Perlmutter at NERSC. https://docs.nersc.gov/systems/perlmutter/.

[22] Ang Li, Samuel Stein, Sriram Krishnamoorthy, and James Ang. {rm QASMBench: A low-level quantum benchmark suite for nisq evaluation and simulation. *ACM Transactions on Quantum Computing*, 4(2), feb 2023.