Title

Introduction to the Special Issue on Software Tools for Quantum Computing: Part 1

Journal

ACM Transactions on Quantum Computing, 3(3)

ISSN

2643-6809

Authors

Alexeev, Yuri

McCaskey, Alex

De Jong, Wibe

Publication Date

2022-09-30

DOI

10.1145/3532179

Special Issue on Software Tools for Quantum Computing Part 1 – Introduction

Yuri Alexeev (yuri@anl.gov)
Computational Science Division, Argonne National Laboratory,
Alex McCaskey (amccaskey@nvidia.com),
NVIDIA,
Wibe de Jong (wadejong@lbl.gov)
Computational Science Department Applied Computing for Scientific Discovery,
Lawrence Berkeley National Laboratory

Quantum computing is emerging as a remarkable technology that offers the possibility of achieving major scientific breakthroughs in many areas. By leveraging the unique features of quantum mechanics, quantum computers may be instrumental in advancing many areas, including science, energy, defense, medicine, and finance. This includes solving complex problems whose solution lies well beyond the capacity of contemporary and even future supercomputers that are based on conventional computing technologies. As a foundation for future generations of computing and information processing, quantum computing represents an exciting area for developing new ideas in computer science and computational engineering.

Interacting with the emerging capabilities of quantum computers, including noisy-intermediate scale quantum devices, for both basic and applied research will require an end-to-end software stack, not unlike the one we rely on in classical computing. This quantum software stack plays an important role in the quantum computing ecosystem, providing quantum practitioners with the essential tools to take advantage of the quantum revolution. Critical components of a quantum software stack include programming models and languages, compilers, verification, and debugging tools, and hardware control capabilities. While advances are being made by the community, we are still far off from providing quantum practitioners with a cohesive software toolchain.

Over the last few years, there has been considerable effort to develop software tools that make quantum computing technology more accessible to the broader community. Many of those developed by industry, national laboratories, and academia are being made available as open-source software tools. Programming languages are being developed that make it easier for domain scientists to translate their science onto quantum computers. Similar to classical computing, compilers have been developed with the aim of minimizing the resource needs with respect to the number of quantum processing units (qubits, qutrits, etc.) and the number of quantum operations that need to be performed. To aid in the development and testing of new algorithms, scalable numerical simulators and resource profilers have been developed, which form a critical component of the quantum computing software ecosystem. Only recently, approaches and tools have been developed for verifying, validating, and debugging quantum computer programs and quantum computer hardware. Finally, operating on quantum computers requires a quantum control software toolset that is likely to be hardware-technology specific. Continued research and development of a broad and open-source collection of software tools and techniques will be critical to enabling the broad adoption of quantum computing in research and industry.

The purpose of this special issue is to present recent research and development accomplishments resulting in the implementation and availability of new quantum computing software tools that will make quantum computing more practical and accessible. We hope that this special issue provides a springboard for new ideas and development across the quantum computing software stack.

The topic of this special issue stems from the First International Workshop on Quantum Computing Software [1] that was held in conjunction with The International Conference for High-Performance Computing, Networking, Storage, and Analysis in 2020 (SC20). Contributions for the special issue, drawn from the SC20 community and the broader field of quantum software development, were split into two parts.

In this first part of the special issue, six papers are presented which make important progress toward the development of a quantum software ecosystem.

- “OpenQASM 3: A broader and deeper quantum assembly language” by A.W. Cross, A. Javadi-Abhari, T. Alexander, N. de Beaudrap, L.S. Bishop, S. Heidel, C.A. Ryan, J. Smolin, J.M. Gambetta, and B.R. Johnson
  This paper introduces new features in OpenQASM language that both expand and deepen the scope of quantum circuits that can be described with particular focus on their physical implementation and the interactions between classical and quantum computing. In particular, this manuscript has illustrated the primary new features introduced by OpenQASM 3 and put those features in context with examples.
- “Tools for Quantum Computing Based on Decision Diagrams” by R. Wille, S. Hillmich, and L. Burgholzer
  In this work, the authors explained how decision diagram techniques could be used for the design of quantum circuits and presented an easily-accessible tool that visualizes quantum decision diagrams as well as their applications. This approach can be benefitted by users who want to solve their problems and developers who want to enhance or integrate the tools to tackle their specific problems in quantum computing.
- “Enabling dataflow optimization for quantum programs” by D. Ittah,T. Häner, V. Kliuchnikov, and T. Hoefler
  The authors proposed to use multi-level intermediate representation (MLIR) for quantum computing that specifically targets quantum-classical co-optimization. In particular, it supports carrying out such optimizations at the application scale, allowing for optimized resource estimates of large-scale quantum programs. Moreover, it supports quantum-specific optimization passes that may fully leverage the infrastructure provided by MLIR.
- “Leveraging state sparsity for more efficient quantum simulations” by S. Jaques and T. Häner
  In this work, the authors introduced a new method for simulations of quantum circuits that exploits this sparsity to reduce memory usage and simulation runtime. The simulator was benchmarked on quantum algorithms for factoring, for computing integer and elliptic curve discrete logarithms, and for chemistry.
- “PyMatching: A Python package for decoding quantum codes with minimum-weight perfect matching” by O. Higgott

This paper introduces the software package PyMatching, a fast open-source Python package for decoding quantum error-correcting codes with the minimum-weight perfect matching algorithm. The performance of PyMatching was benchmarked for a variety of problem sizes typically considered in error correction simulations.

- “ArQTiC: A full-stack software package for simulating materials on quantum computers” by L. Bassman, C. Powers, and W.A. de Jong
  ArQTiC is an open-source, full-stack software package built for the simulations of materials on quantum computers. It currently can simulate materials that can be modeled by any Hamiltonian derived from a generic, one-dimensional, time-dependent Heisenberg Hamiltonian. Moreover, ArQTiC includes modules for generating quantum programs for real- and imaginary-time evolution, quantum circuit optimization, connection to various quantum backends via the cloud, and post-processing of quantum results.

Finally, we would like to thank ACM Transactions on Quantum Computing for the invitation to serve as the guest editors for this special issue. We would also like to thank all the contributing authors and reviewers who found time and put effort into this special issue, in spite of the pandemic.

References:

[1] https://sc20.supercomputing.org/proceedings/workshops/workshop_pages/wksp109.html