Hybrid Quantum-Classical Computing Architectures

Martin Suchara, Yuri Alexeev, Frederic Chong, Hal Finkel, Henry Hoffmann,
Jeffrey Larson, James Osborn, and Graeme Smith

Abstract—We describe how classical supercomputing can aid unreliable quantum processors of intermediate size to solve large problem instances reliably. We advocate using a hybrid quantum-classical architecture where larger quantum circuits are broken into smaller sub-circuits that are evaluated separately, either using a quantum processor or a quantum simulator running on a classical supercomputer. Circuit compilation techniques that determine which qubits are simulated classically will greatly impact the system performance as well as provide a tradeoff between circuit reliability and runtime.

✦

1 INTRODUCTION

In the past decade, quantum hardware has achieved dramatic improvements in gate fidelities, qubit coherence times, and qubit counts. Small quantum circuit fragments have been experimentally realized with superconducting qubits [21] as well as with ion traps [7]. However, due to fundamental engineering limitations, the number of qubits on a chip, gate fidelities, and qubit coherence times will remain modest in the near future. While quantum algorithms achieve exponential speedup for some problems [17], [22], they require deep quantum circuits involving hundreds of qubits to solve problem instances that are intractable for classical computers [14]. Moreover, unreliable quantum hardware can only evaluate such large and deep circuits with the help of quantum error-correcting codes [10] that add several orders of magnitude of space and time overhead [20]. This paper explores a hybrid quantum-classical architecture that allows the use of small, unreliable quantum processors to evaluate large quantum circuits.

Our earlier theoretical work [5] shows how a quantum computation requiring n qubits can be completed using $n-k$ qubits and additional classical computation resources, as shown in Figure 1. This approach was later generalized [12] to allow breaking a larger quantum circuit into smaller sub-circuits that can be evaluated separately. In such a decomposition, the classical computing costs scale exponentially with the number of gates between the sub-circuits. For this reason it is important to minimize the number of gates between the sub-circuits.

Our approach presents a tradeoff between quantum and classical computing resources. Some circuits can be evaluated more efficiently on classical computers and some on quantum processors. Classical computers are particularly efficient at evaluating sub-circuits consisting of Clifford gates [11] or mostly Clifford gates [4]. Near-term quantum hardware, on the other hand, is expected to excel in evaluating sub-circuits that are shallow, but potentially densely connected. Shallow circuit depths allow evaluation without the use of quantum error correcting codes and avoid the associated overhead.

Another tradeoff exists between computational cost and fidelity. To increase the fidelity we suggest to partition quantum

- *Martin Suchara, Yuri Alexeev, Hal Finkel, Jeffrey Larson and James Osborn are with Argonne National Laboratory, Lemont, IL, U.S.A. E-mail: msuchara@anl.gov*
- *Frederic Chong and Henry Hoffmann are with the University of Chicago, Chicago, IL, U.S.A.*
- *Graeme Smith is with the University of Colorado, Boulder, CO, U.S.A.*

circuits to allow small, fidelity-critical sub-circuits to be simulated on classical supercomputing resources that offer nearly perfect reliability. Partitioning that minimizes the number of gates between sub-circuits may reduce the classical computing overhead but possibly at the cost of reducing the fidelity of the computation. New compiler analysis techniques will be needed to allow suitable quantum circuit partitioning, and numeric simulations with representative quantum circuits can help us understand and optimize the outlined performance tradeoffs.

We argue that using hybrid quantum-classical architectures will be necessary to solve large computational problems with near term quantum hardware. In this position paper we describe this architecture, identify existing tools that can be used to build the system, and discuss some open questions.

2 BUILDING A QUANTUM-CLASSICAL ARCHITECTURE

Figure 2 depicts a hybrid architecture that uses a classical supercomputer and quantum processors in the cloud. The computation in Figure 2 consists of the following steps:

(1) Compiling a quantum algorithm into a quantum circuit: The quantum algorithm must be first programmed in a quantum programming language and compiled. We note that the compiled circuit is not fault-tolerant yet and could be successfully run only on reliable hardware. The Scaffold programming language and ScaffCC compiler [13] provide scalable environment capable of compiling large quantum circuits. A variety of quantum circuits suitable for performance evaluation and benchmarking of the proposed architecture are available in Scaffold [13], and additional quantum circuits solving practical problems can be obtained e.g. from OpenFermion [15].

(2) Partitioning of the circuit into suitable sub-circuits: Partitioning into sub-circuits to be run on quantum and classical processors will require new compiler tools. These tools need to achieve a long list of conflicting goals. To reduce the time overhead, the compiler must minimize the number of gates connecting the sub-circuits. The sub-circuits must be small enough to fit on the available quantum or classical hardware. The compiler should also identify which sub-circuits can be simulated efficiently on classical computers. Finally, to improve the fidelity of evaluating the quantum circuit, qubits sensitive to errors should be simulated on reliable classical hardware.

(3) Copying circuit descriptions and inputs: The complied sub-circuits and inputs are distributed to the quantum processors (that may be located off-site) and to the classical simulators
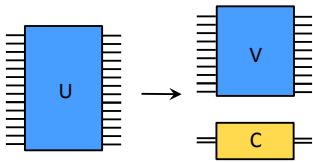
Fig. 1. A compiled quantum circuit U is decomposed into a smaller quantum circuit V and some classical computation C.
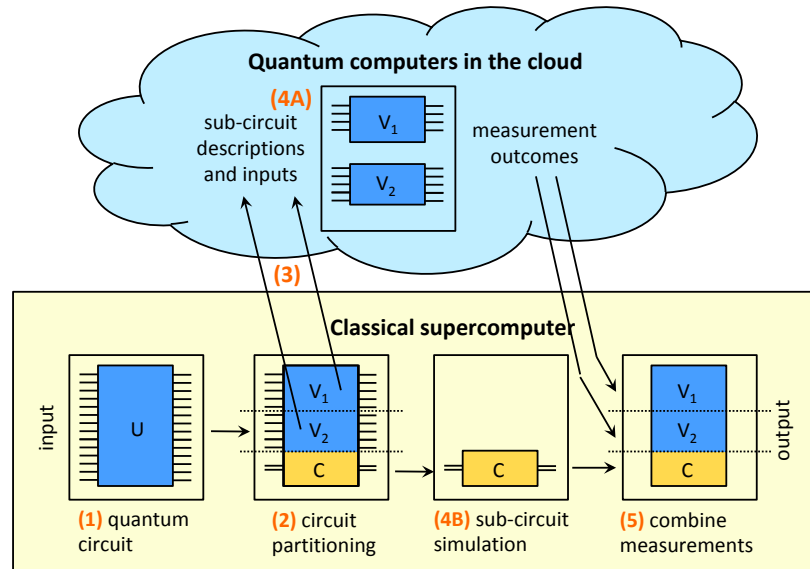

Fig. 2. The quantum circuit U is decomposed into a classical circuit C and quantum circuits $V_1...V_n$. The quantum circuits are evaluated on quantum computers in the cloud and the classical circuit on a classical supercomputer.

(that reside at the supercomputing site). Prior to running the circuits on quantum hardware, it must be determined if the hardware is reliable enough and the circuit is shallow enough so that the computation can succeed with high enough probability. If higher success probability is needed, the circuits must be made fault tolerant by using a suitable quantum error correcting code. The theory of error correction is well developed [8], [10] and small circuits that demonstrate error correction were realized experimentally [6].

(4A) Computing on quantum processors: We envision that the quantum processors will be accessed remotely. For example, IBM made its superconducting quantum computer with 20 qubits accessible in the cloud [2]. A growing number of other experimental devices [1] and simulators [19] also allow remote access. The outputs of quantum circuits are qubit measurement outcomes represented by classical bits, and can be therefore transmitted efficiently back to the supercomputing site for further processing. The only drawback of remote quantum processor access is increased latency.

(4B) Classical computing on a supercomputer: Some sub-circuits are evaluated on a classical supercomputer. Classical circuits can be evaluated using traditional techniques, and quantum circuits can be evaluated by a quantum simulator. A suitable simulator is e.g. Intel-QS [18] capable of evaluating general quantum circuits with up to approximately 50 qubits. Intel-QS takes full advantage of multi-core and multi-node architectures and was ported to the Cray/Intel supercomputer Theta [3].

(5) Combining the results: In the final step the results feeding from the quantum and classical circuits are combined, and steps 3 and 4 are possibly repeated with the same or different circuit inputs.

3   DISCUSSION AND OPEN QUESTIONS

The overall system performance critically depends on the choice of sub-circuits that are simulated classically. This is a difficult problem as there are exponentially many choices of

which qubits should be simulated and when. Additional complication is a tradeoff between running time and circuit fidelity. Classically simulated qubits are perfect whereas physical qubits are subject to loss and decoherence. At one extreme we can classically simulate qubits that require the highest reliability without regard to the connectivity of the circuit. This approach will optimize the circuit fidelity at the cost of a potential exponential increase in running time. Another extreme is to find the most natural separation of the quantum circuit into sub-circuits to minimize the number of gates connecting the sub-circuits. This approach minimizes the running time but does not consider the fidelity of the computation. Benchmarking of various strategies will be needed to determine the appropriate tradeoff.

The discussed architecture leverages small geographically distributed quantum processor prototypes to solve larger computational tasks. An interesting question is how can computation proceed effectively if there is a communication delay of tens of milliseconds between the processors and circuits have to be partitioned into very small sub-circuits.

Yet another question is which algorithms are best suited to benefit from hybrid quantum-classical computing. The methods in [5] and [12] require clean circuit separation into sub-circuits without cutting too many two-qubit gates. Fortunately, there are promising practical problems that have quantum circuits with low depth and sparse connectivity. Such separable circuits include Quantum Approximate Optimization Algorithms (QAOA) [9] and some quantum machine learning algorithms.

Besides more efficient computing, hybrid quantum-classical architectures can be also useful for validation of the outputs of a quantum computer [16]. For some computation, we can replace a potentially noisy physical qubit with a perfectly performing classically simulated qubit. If the output is unaffected by this replacement, it suggests that any noise on the physical qubit can be considered benign. If we replace a logical qubit protected by error correcting code by a perfect classical qubit, matching output can also verify that the gate sequence that performs the error correction was correct and effective.

REFERENCES

[1] "Alibaba cloud and CAS launch one of the worlds most powerful public quantum computing services," March 2018. [Online]. Available: https://www.alibabacloud.com/press-room/alibaba-cloud-and-cas-launch-one-of-the-worlds-most

[2] (2018) IBM Q Experience. [Online]. Available: https://quantumexperience.ng.bluemix.net/qx/experience

[3] Y. Alexeev, "ANL/ALCF -18/2 Evaluation of the Intel-QS performance on Theta supercomputer," Argonne National Laboratory - Leadership Computing Facility, Tech. Rep., May 2018. [Online]. Available: https://www.alcf.anl.gov/files/ANL.technical.report.v6.pdf

[4] S. Bravyi and D. Gosset, "Improved classical simulation of quantum circuits dominated by Clifford gates," *Phys. Rev. Lett.*, vol. 116, p. 250501, Jun 2016.

[5] S. Bravyi, G. Smith, and J. A. Smolin, "Trading classical and quantum computational resources," *Phys. Rev. X*, vol. 6, p. 021043, Jun 2016.

[6] A. D. Córcoles, E. Magesan *et al.*, "Demonstration of a quantum error detection code using a square lattice of four superconducting qubits," *Nature Communications*, vol. 6, pp. 6979 EP –, Apr 2015.

[7] S. Debnath, N. M. Linke *et al.*, "Demonstration of a small programmable quantum computer with atomic qubits," vol. 536, pp. 63–66, Aug 2016.

[8] E. Dennis, A. Kitaev *et al.*, "Topological quantum memory," *Journal of Mathematical Physics*, vol. 43, no. 9, pp. 4452–4505, 2002.

[9] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv:1411.4028*, Nov 2014. [Online]. Available: https://arxiv.org/abs/1411.4028

[10] D. Gottesman, "Stabilizer codes and quantum error correction," Ph.D. dissertation, California Institute of Technology, 1997.

[11] ——, "The Heisenberg representation of quantum computers," *arXiv:quant-ph/9807006*, Jul 1998. [Online]. Available: https://arxiv.org/abs/quant-ph/9807006

[12] A. Harrow, M. Ozols *et al.*, "Simulating large quantum circuits on a small quantum computer," Nov 2016. [Online]. Available: https://pdfs.semanticscholar.org/3ed0/c125055f2143ffc9be9972b5cb97efa72fac.pdf

[13] A. JavadiAbhari, S. Patil *et al.*, "ScaffCC: A framework for compilation and analysis of quantum computing programs," *Proceedings of the 11th ACM Conference on Computing Frontiers, CF 2014*, May 2014.

[14] N. C. Jones, R. Van Meter *et al.*, "Layered architecture for quantum computing," *Phys. Rev. X*, vol. 2, p. 031007, Jul 2012.

[15] J. R. McClean, I. D. Kivlichan *et al.*, "OpenFermion: The electronic structure package for quantum computers," *arXiv:1710.07629*, Jun 2018. [Online]. Available: https://arxiv.org/abs/1710.07629

[16] S. Z. Robert Rand, Jennifer Paykin, "QWIRE practice: Formal verification of quantum circuits in Coq," *arXiv:1803.00699*, Mar 2018. [Online]. Available: https://arxiv.org/abs/1803.00699

[17] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM J. Comput.*, vol. 26, no. 5, pp. 1484–1509, Oct. 1997.

[18] M. Smelyanskiy, N. P. D. Sawaya, and A. Aspuru-Guzik, "qhipster: The quantum high performance software testing environment," *CoRR*, vol. abs/1601.07195, 2016.

[19] R. S. Smith, M. J. Curtis, and W. J. Zeng, "A practical quantum instruction set architecture," 2016.

[20] M. Suchara, A. Faruque *et al.*, "Comparing the overhead of topological and concatenated quantum error correction," *arXiv:1312.2316*, Dec 2013. [Online]. Available: https://arxiv.org/abs/1312.2316

[21] M. Takita, A. D. Córcoles *et al.*, "Demonstration of weight-four parity measurements in the surface code architecture," *Phys. Rev. Lett.*, vol. 117, p. 210505, Nov 2016.

[22] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, "Simulation of electronic structure hamiltonians using quantum computers," *Molecular Physics*, vol. 109, p. 735–750, Mar 2011.