Thesis Proposal: Low-depth quantum architectures

Paul Pham
University of Washington
ppham@cs.washington.edu

April 26, 2013

1 Committee

- Aram Harrow (chair)
- Paul Beame
- Mark Oskin
- Boris Blinov (GSR)

2 Abstract

Building upon my generals exam, I combine the common threads of realistic architectural constraints (e.g. nearest-neighbor interactions), quantum circuit construction techniques (e.g. parallelization), quantum compiling, and a new circuit resource called circuit coherence. I develop these threads in the context of the well-known Shor factoring algorithm and then generalize them to a new algorithm, Hamiltonian simulation. I calculate closed formulae for the resources needed to run these algorithms on input sizes of interest. Thus, I characterize the possibility of trading space (circuit width, up to billions of qubits, for time (circuit depth, up to centuries of time).

Thesis Statement: Advances in nearest-neighbor quantum architectures, quantum compilation, and quantum circuit analysis can help us design quantum computers to solve human problems within human lifetimes.

3 Updated and Proposed Timeline

February 21 I send this proposal to committee, accept revisions.

February 22 I begin writing, at least on non-controversial parts of proposal, until we come to a consensus on the whole proposal. This 6 week period is broken down by chapter in the next section.

April 5 I finish writing and remaining research and send a penultimate dissertation draft to the reading committee. Beginning of two week reading period.

April 19 End of two week reading period. Reading committee gives me feedback and decides whether thesis is ready to be defended. I schedule my final exam with the graduate school (tentatively set for May 3rd), make requested changes to dissertation, and prepare my talk.

May 3 Tentative final exam with full committee present, in CSE 503, 10am - 12:30pm.

June 7 Tentative final exam with full committee present, in CSE 503, 10:30am - 12:30pm.

4 Proposed Chapters

1. **Factoring on a 2D nearest-neighbor architecture.**

   In this chapter, we discuss the work Pham-Svore 2012 [17], where we mapped Shor's factoring algorithm to a 2D nearest-neighbor quantum architecture (with classical controller) in polylogarithmic depth, an exponential improvement over the previous best-known architecture (in 1D). Furthermore, we will improve these results to be constant-depth, which is optimal even if we allow an architecture with arbitrary interactions. We will use the techniques, namely the unbounded quantum fanout of Høyer-Špalek 2002 [13], the quantum threshold circuit given in Takahashi-Tani 2011 [22], and the original classical circuit given in Siu et al. 1993 [21]. We will calculate the circuit resources required by our factoring architectures and compare them to previous implementations, especially the best-known 1D nearest-neighbor algorithm by Kutin 2006 [16].

   This chapter will also include a background of quantum architecture, and physical justification for the realism and utility of our chosen models. It will also review other work on mapping quantum gates of arbitrary connectivity to a nearest-neighbor architecture such as Rosenbaum 2012 [18] and Biels et al. 2012 [5].

2. **Quantum compiling on a nearest-neighbor architecture.**

   This section is primarily a pedagogical literature review which builds upon my quals project.

   Quantum compiling is the approximation of any quantum gate to arbitrary precision using gates from a universal, finite, discrete set. Along with error correction, it provides one of the biggest overheads in the realistic implementation of a quantum algorithm. In this case, we limit ourselves to approximating single-qubit gates, which is sufficient to approximate multi-qubit gates using known decompositions given in Kitaev-Shen-Vyalyi 2002 [1], Svore-Aho 2003 [3], and Saeedi-Markov 2011 [19].

   There has been much recent work in approximating single-qubit gates and improving empirical performance using heuristics, such as Amy et al. 2012 [4], Booth 2012 [8], Kliuchnikov et al. 2012 [15], Eastin 2012 [12] Selinger 2012 [20], Bocharov-Svore 2012 [7]. An interesting model which uses the technique of Kitaev-Shen-Vyalyi 2002 [1] to prepare programmable ancillae offline and then apply them in constant depth at runtime is given in Jones et al. [14]. There has also been an asymptotic improvement by Duclos-Cianci-Svore 2012 [11]. These all improve upon the original results by Solovay-Kitaev 1995-1997 as formulated in Dawson-Nielsen 2005 [10] and a later result by Kitaev-Shen-Vyalyi 2002 [1].

   We will give a pedagogical review all of these works and compare their relative performances on architectures with arbitrary interactions. As well, we will compute the circuit resource overhead in mapping one or more of these methods to our 2D nearest-neighbor architecture with classical controller, as time allows.

3. **Quantum coherence versus measurement patterns.**

   In this chapter, I re-introduce a new circuit resource first mentioned in my generals report which I now call *circuit coherence*. Roughly defined, it is the time-space product of the coherent computation state that ends in measurement to produce a circuit's final classical output. It has units of *qubits· timestep*. Intuitively, it measures the amount of experimental labor and error-correction needed to maintain a coherent, entangled state for computation.

   Circuit size is counted as the number of two-qubit (nearest-neighbor) gates in a circuit, and a timestep is the unit depth of a two-qubit gate. Then circuit coherence is upper-bounded by the product of circuit depth times width, which in the worst case means that all qubits in a quantum computer must remain entangled for the entire runtime of the circuit. It is lower-bounded by the circuit size, which in the worst case means that every two-qubit gate is alternately entangling or disentagling a qubit from the computation state.

   The questions examined in this section are as follows:

(a) Is circuit coherence a well-defined circuit resource? Are there pathological cases to be handled, and if so, what are the workarounds to the definition given above?

(b) Are there transformations to a circuit that decrease circuit coherence while still computing the same function? What are some specific examples? Can we generalize some principles, properties, or pseudocode that would allow us to automatically determine that a circuit is "canonical"? That is, a canonical circuit has the minimum circuit coherence among all its equivalent representations.

(c) Is circuit coherence asymptotically separated from the upper and lower bounds given above? If not, what are specific examples where there is a separation between coherence and depth $\times$ width?

(d) Is circuit coherence different from the measurement patterns defined in Broadbent-Kasheffi 2007 [9] ?

4. **Mapping the Hamiltonian simulation algorithm to a 2D nearest-neighbor architecture.**

   A very different flavor of quantum algorithm is Hamiltonian simulation, the original problem Richard Feynman proposed for which a quantum mechanical machine might be exponentially faster than a classical machine. Hamiltonian simulation, or the simulation of an inherently quantum physical system with many-body interactions, has been attracting a lot of attention recently. We can conceivably demonstrate quantum speedups in experimental implementations of Hamiltonian simulation in the near future that match classical supercomputer performance on modest input sizes. We cannot say the same for factoring for at least many more years.

   This chapter is partly a review of related work on Hamiltonian simulation but is primarily original research on mapping a k-sparse Hamiltonian to our 2D nearest-neighbor architecture with classical controller. We also calculate the circuit resources for such a mapping. This gives a concrete implementation for the basic building block of the Hamiltonian simulation steps in Berry et al. 2005 [6] and Aharanov and Ta-Shma 2003 [2].

   This chapter is not meant to be a comprehensive survey, but rather a novel and concrete application of the previous three chapters (architectural lessons learned from the factoring algorithm, quantum compiling as a subroutine to quantum algorithms, and circuit coherence) to a new algorithm.

5. **Conclusion.** We conclude with interesting open problems and ways to extend the work of these four chapters in the future.

References

[1] M.N. Vyalyi A. Yu. Kitaev, A.H. Shen: *Classical and Quantum Computation*. American Mathematical Society, Providence, Rhode Island, 2002.

[2] Dorit Aharonov and Amnon Ta-Shma: Adiabatic Quantum State Generation and Statistical Zero Knowledge. p. 35, January 2003. [arXiv:0301023].

[3] Alfred V. Aho and Krysta M. Svore: Compiling Quantum Circuits using the Palindrome Transform. p. 17, November 2003. [arXiv:0311008].

[4] Matthew Amy, Dmitri Maslov, Michele Mosca, and Martin Roetteler: A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits. p. 20, June 2012. [arXiv:1206.0758].

[5] Robert Beals, Stephen Brierley, Oliver Gray, Aram Harrow, Samuel Kutin, Noah Linden, Dan Shepherd, and Mark Stather: Efficient Distributed Quantum Computing. p. 28, July 2012.

[6] Dominic W. Berry, Graeme Ahokas, Richard Cleve, and Barry C. Sanders: Efficient Quantum Algorithms for Simulating Sparse Hamiltonians. *Communications in Mathematical Physics*, 270(2):359–371, December 2006. [doi:10.1007/s00220-006-0150-x, arXiv:0508139].

[7] Alex Bocharov and Krysta M. Svore: A Depth-Optimal Canonical Form for Single-qubit Quantum Circuits. p. 10, June 2012. [arXiv:1206.3223].

[8] Jeffrey Booth: Quantum Compiler Optimizations. June 2012. [arXiv:1206.3348].

[9] Anne Broadbent and Elham Kashefi: Parallelizing Quantum Circuits. April 2007. [arXiv:0704.1736].

[10] Christopher M. Dawson and Michael A. Nielsen: The Solovay-Kitaev algorithm. p. 15, May 2005. [arXiv:quant-ph/0505030].

[11] Guillaume Duclos-Cianci and Krysta M. Svore: A State Distillation Protocol to Implement Arbitrary Single-qubit Rotations. p. 10, October 2012. [arXiv:1210.1980].

[12] Bryan Eastin: Distilling one-qubit magic states into Toffoli states. p. 8, December 2012. [arXiv:1212.4872].

[13] Peter Høyer and Robert Špalek: Quantum Circuits with Unbounded Fan-out. *Theory of Computing*, 1(5):81–103, August 2005. [arXiv:0208043].

[14] N. Cody Jones, James D. Whitfield, Peter L. McMahon, Man-Hong Yung, Rodney Van Meter, Alán Aspuru-Guzik, and Yoshihisa Yamamoto: Simulating chemistry efficiently on fault-tolerant quantum computers. p. 33, April 2012. [arXiv:1204.0567].

[15] Vadym Kliuchnikov, Dmitri Maslov, and Michele Mosca: Practical approximation of single-qubit unitaries by single-qubit quantum Clifford and T circuits. p. 9, December 2012. [arXiv:1212.6964].

[16] Samuel A. Kutin: Shor's algorithm on a nearest-neighbor machine. August 2006. [arXiv:0609001].

[17] Paul Pham and Krysta M. Svore: A 2D Nearest-Neighbor Quantum Architecture for Factoring. p. 15, July 2012. [arXiv:1207.6655].

[18] David Rosenbaum: Optimal Quantum Circuits for Nearest-Neighbor Architectures. April 2012. [arXiv:1205.0036].

[19] Mehdi Saeedi and Igor L. Markov: Synthesis and Optimization of Reversible Circuits - A Survey. p. 34, October 2011. [arXiv:1110.2574].

[20] Peter Selinger: Efficient Clifford+T approximation of single-qubit operators. December 2012. [arXiv:1212.6253].

[21] K.-Y. Siu, J. Bruck, T. Kailath, and T. Hofmeister: Depth efficient neural networks for division and related problems. *IEEE Transactions on Information Theory*, 39(3):946–956, May 1993. [doi:10.1109/18.256501].

[22] Yasuhiro Takahashi and Seiichiro Tani: Constant-Depth Exact Quantum Circuits for the OR and Threshold Functions. p. 17, December 2011. [arXiv:1112.6063].