

Time Optimal Qubit Mapping

Chi Zhang¹, Ari Hayes², Longfei Qiu², Yuwei Jin², Yanhao Chen², and Eddy Z. Zhang²

¹University of Pittsburgh

²Rutgers University

1. Motivation

Quantum computing is a promising method to speed up a number of important applications. These applications include factoring large numbers [7], searching a database [2], and simulating quantum systems [6]. With around 100 reliable qubits, quantum computers can already solve useful problems that are out of reach for classical computers. The rapid progress of physical implementation of quantum computers paved the way for the design of tools to help users write quantum programs for any given quantum device. However, physical constraints inherent in current NISQ architectures prevent most quantum algorithms from being directly executed on quantum devices.

In the superconducting quantum computers, qubits operate in the nearest neighbor architecture, in which direct interactions form a bounded degree coupling graph. To enable two-qubit gates, logical qubits must be mapped to qubits that are physically placed next to each other. Due to the bounded degree connectivity of physical qubits on current devices, it is generally considered impossible to find an initial mapping that makes the entire circuit hardware-compliant. The common practice is to dynamically remap logical qubits to physical qubits via SWAP gates such that each two-qubit gate is mapped to two physically connected qubits. This problem is called *qubit mapping* problem.

Our paper focuses on the qubit mapping problem. The qubit mapping problem takes a logical circuit and a hardware coupling graph as input, outputs a transformed circuit. Only swap operations are allowed to be added into the transformed circuit. After transformation, all two-qubit gates must be performed on qubits that have direct links. Our goal is to let programmers develop quantum algorithms that take full advantage of the potentially disruptive computing paradigm without having to worry about low level machine details. Our effort is an important step for building the required compilation tool for near-term quantum computers.

2. Limitations of the State of the Art

Previous studies on the qubit mapping problem [3, 13, 14, 15, 10, 9] have focused on *gate-optimal* solution. They minimize the number of inserted SWAP gates. Some [15, 3] also enhance the parallelism among the inserted SWAP gates while minimizing the gate count. Zulehner *et al.* [15] propose a systematic solution using A-star paradigm for optimizing the number of swap gates for a given layer of concurrent CNOT gates in the circuit. Li *et al.* [3] formulate a multi-objective

function for exploiting the trade-off between different swap insertion strategies. The study by Siraichi *et al.* [9] models the swap-insertion problem as a subgraph isomorphism problem. Wille *et al.* [13] propose a model for gate-optimal mapping using the SAT solver. A number of studies [12, 5] note the variability of qubit error rate in IBM quantum computer and develop variability-aware qubit mapping strategies. None of the studies above focuses on time-optimal qubit mapping. That is, minimizing the execution time of the hardware-compliant circuit instead of the gate count. A SWAP operation, although can be implemented in different ways, is decomposed into quantum gate(s) as well. The process of adding SWAPs to a logical circuit involves adding extra gates into the circuit. As a result, it would alter the structure of the circuit. The inserted swap gates need to fit into the original circuit in a synergistic way to maximize execution parallelism.

Time-optimal qubit mapping maximizes parallelism. It has an additional benefit of mitigating the *decoherence* effect. Qubits are error prone. A qubit's energy decays over time. It gradually lose its state information. The phenomenon is called decoherence. The longer it takes to run a circuit, the more likely a qubit decoheres. A time-optimal solution minimizes the impact of decoherence for the qubits in the circuit, and results in higher successful trial rate for the circuit as a whole. There are two previous works [11, 1] focusing on time-optimal qubit mapping, but they impose implicit constraints. [11] uses a constrained based ILP solver, which solves for the time coordinate of each gate (including swap gate) and the qubit mapping at every time coordinate. It adopts an indirect approach for optimal depth circuit. It tests different upper bounds of the circuit depth until the ILP has a solution. The method may suffer from scalability issues when the optimal circuit time is not close to the set upper bound. Our model does not impose such a constraint. It explicitly solves for the optimal solution. [1] uses greedy depth-aware methods but it only optimizes the depth of the inserted swaps. Our mapper optimizes the depth of the entire circuit.

3. Key Insights

It is not trivial to achieve time optimality when addressing the qubit mapping problem. As there are so many possible permutations of circuit gates and inserted swaps. However, Time-optimal solutions, even for small size circuit, could be very useful. If an optimal solution for a logical circuit has recurring pattern, we can obtain the optimal solution for small-size inputs, and use that to deduce the generalized solution.

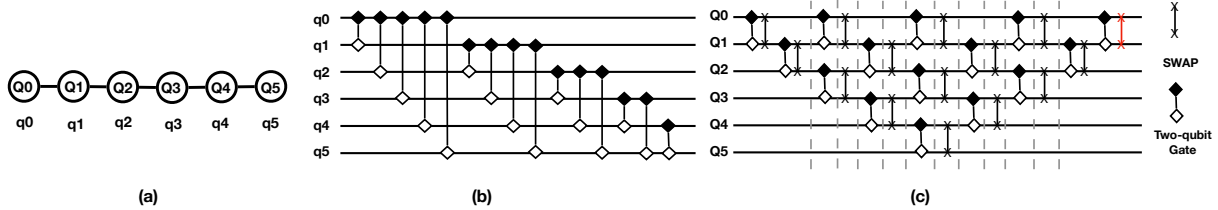


Figure 1: Adapting QFT logical circuit to LNN architecture: (a) 6-qubit LNN and initial mapping, (b) logical qft-6 circuit where each line represents a logical qubit q_i , and (c) physical qft-6 circuit where each line represents a physical qubit Q_i . The last swap gate in red in (c) is added for showing the symmetric pattern. Our solver does not have the swap in its returned solution.

We use quantum fourier transformation (QFT) to demonstrate the usefulness of our implemented time-optimal qubit mapper. QFT is at the heart of integer factorization [8] and serves as the basis for many important quantum algorithms.

It is challenging to map QFT to a bounded degree coupling graph. The logical QFT circuit requires all-to-all connection. The skeleton QFT circuit is shown in Fig. 1 (b). Even for the simplest architecture, for instance, the linear nearest neighbor architecture (LNN), it is not trivial to immediately tell what would be the best qubit mapping solution. In the LNN, qubits are placed in a conceptually straight line. An example of 6-qubit LNN is shown in Fig. 1 (a).

We use the time-optimal qubit mapper developed in this paper to solve for 6-qubit QFT on LNN. Our qubit mapper returns a physical circuit as shown in Fig. 1 (c). A very clear pattern for the optimal solution manifests itself in this solution. We find that this pattern can be generalized to arbitrary size QFT. Our generalized solution happens to be the same as the solution by Maslov *et al.* [4]. However, the result by Maslov *et al.* is found manually. It is published in physics literature as a standalone paper. Our automatic qubit mapper finds the optimal solution for 6-qubit QFT on LNN in < 1 second.

Moreover, for QFT on more complicated architectures, we are not aware of any concrete optimal solution discovered. Maslov *et al.* predicted lower bound time complexity of transformed QFT circuit. In this paper, using our automatic qubit mapping framework, we find and generalize a solution for QFT on arbitrary size $2 \times N$ architectures. Our transformed circuit complexity matches the lower bound complexity proved by Maslov *et al.*. Hence it is verified to be optimal, not only for small number of, but also for arbitrary number of qubits.

4. Main Artifacts

We develop a simple and natural model to represent the search space of all hardware-compliant executions for the same logical circuit. We develop a qubit mapping algorithm based on the A* paradigm. Our algorithm is both optimal and complete. We further build a framework of techniques to reduce the search complexity in the exponential search space. We also build an approximated non-optimal qubit mapper.

The algorithm and the framework of techniques are implemented using C++. They are evaluated on important benchmarks including quantum fourier transformation and logical

reversible circuits for synthesis. They are compared against the theoretical result of lower bound complexity of the evaluated benchmark in physics literature [4] and two best qubit mapping algorithms we are aware of [3, 15].

5. Key Results and Contributions

Our work enhances the understanding of time-optimal qubit mapping problem in the following ways:

- We present the first theoretical model for time-optimal qubit mapping. It lends itself to optimality guarantee, flexible extension, and practical algorithm design.
- We present a search framework based on our time-optimal mapping model. It consists of space pruning components including redundancy elimination and comparative filter. It significantly reduces the time complexity of the search, and makes our qubit mapping method feasible.
- We present exact analysis by implementing our theoretical model for solving quantum applications with regular execution patterns. We find a time-optimal solution for the quantum fourier transformation (QFT) program on linear nearest neighbor architecture. We note our time-optimal solution is the same as an existing solution documented in the physics literature by [4]. However, we find it automatically within 1 seconds while [4] found it by hand.
- We further use the exact analysis program to solve for QFT on two dimensional quantum architecture. We find an optimal solution with complexity $3n + \mathcal{O}(1)$ which no one has found before as far as we know. [4] proved that a solution's lower bound is $3n + \mathcal{O}(1)$ but did not provide a solution. Therefore we confirmed our solution is indeed optimal.
- We present a scalable extension of our theoretical model that can be used to solve large circuits. The search space is significantly pruned to rule out the branches that are unlikely to yield a good solution. Our practical implementation is an approximation of the optimal model, however, it still outperforms state-of-the-art qubit mappers by a speedup ranging from 1.09x to 2.04X, by 1.62x on average, for representative benchmarks from RevLib, Qiskit, and Scaffold.

References

- [1] Andrew M Childs, Eddie Schoute, and Cem M Unsal. Circuit transformations for quantum architectures. *arXiv preprint arXiv:1902.09102*, 2019.
- [2] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, STOC '96, pages 212–219. New York, NY, USA, 1996. ACM.
- [3] Gushu Li, Yufei Ding, and Yuan Xie. 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*, pages 1001–1014. ACM, 2019.
- [4] Dmitri Maslov. Linear depth stabilizer and quantum fourier transformation circuits with no auxiliary qubits in finite-neighbor quantum architectures. *Physical Review A*, 76(5), Nov 2007.
- [5] Prakash Murali, Jonathan M. Baker, Ali Javadi-Abhari, Frederic T. Chong, and Margaret Martonosi. Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '19, pages 1015–1029. New York, NY, USA, 2019. ACM.
- [6] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. In *Nature Communications*, volume 5, page 4213, 2014.
- [7] Peter W Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. Ieee, 1994.
- [8] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, October 1997.
- [9] Marcos Yukio Siraichi, Vinícius Fernandes dos Santos, Caroline Collange, and Fernando Magno Quintão Pereira. Qubit allocation as a combination of subgraph isomorphism and token swapping. *Proc. ACM Program. Lang.*, 3(OOPSLA), October 2019.
- [10] Marcos Yukio Siraichi, Vinícius Fernandes dos Santos, Sylvain Collange, and Fernando Magno Quintão Pereira. Qubit allocation. In *Proceedings of the 2018 International Symposium on Code Generation and Optimization*, pages 113–125. ACM, 2018.
- [11] Bochen Tan and Jason Cong. Optimal layout synthesis for quantum computing. In *2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, pages 1–9. IEEE, 2020.
- [12] Swamit S. Tannu and Moinuddin K. Qureshi. Not all qubits are created equal: A case for variability-aware policies for nisq-era quantum computers. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '19, pages 987–999. New York, NY, USA, 2019. ACM.
- [13] Robert Wille, Lukas Burgholzer, and Alwin Zulehner. Mapping quantum circuits to ibm qx architectures using the minimal number of swap and h operations. In *Proceedings of the 56th Annual Design Automation Conference 2019*, page 142. ACM, 2019.
- [14] Alwin Zulehner, Stefan Gasser, and Robert Wille. Exact global reordering for nearest neighbor quantum circuits using A*. In *International Conference on Reversible Computation*, pages 185–201. Springer, 2017.
- [15] Alwin Zulehner, Alexandru Paler, and Robert Wille. Efficient mapping of quantum circuits to the ibm qx architectures. In *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 1135–1138. IEEE, 2018.