On Qubit Routing in Quantum Computer Architectures

Master Thesis Proposal

Mark Sobolev

Wednesday 21st May, 2025

Advisors: Prof. Dr. S. Wolf

Co-Advisors: L. Laneve, W. Schober

Faculty of Informatics, USI Lugano

The advisor(s) of this Master's Thesis Proposal confirm that the literature review within this proposal, along with the scope and content of the proposed thesis topic, are sufficient for a masters-level thesis defense in the Faculty of Informatics at Universit della Svizzera italiana..

21.05.2025

Signature of Advisor(s) & Date

Abstract

The basis of all the modern quantum algorithms was outlined by Deutsch [8]. It presented a modelling paradigm for all the quantum algorithms, later proven through the quantum Turing machine equivalence conjecture by Yao [6]. The creation of a quantum computer architecture nowadays is acting as a basic toolset for the projection and analysis of all the quantum computation algorithms. However, the realistic hardware architecture is highly different from the traditional computational model – which is de facto assumed by Deutsch, i.e. von Neumann architecture [15]. This departure in particular is happening because the qubits in the layout don't have the equivalent connectivity properties, i.e. not all the physical qubits in the circuit are made equal (see, for example, an informal discussion of IBM quantum architecture connectivity [1]). Thus, an additional constraint satisfaction at the compilation step is required, termed "qubit routing". We will formalise the process of quantum circuit conversion to the qubit interaction graph, combining time- and spatial-related information. Then we aim to explore the possibilities of the optimal placement of CNOT gates (and potential insertion of SWAP gates) in the architecture-agnostic computation graph. After that, we will take the constraints of the different quantum computer architectures and adapt our routing/mapping algorithms to those, presenting an additional set of graphs. Finally, we will port the results of the routing to the main quantum circuit compilers, e.g. qiskit, and evaluate those on the main benchmarks for the selected quantum architectures.

Contents

Contents **iii**

1 Introduction **1**

2 Related works **3**
2.1 Cowtan's solution 3
2.1.1 Problem statement 3
2.1.2 Algorithm overview 4
2.1.3 Circuit slicing 4
2.1.4 Initial mapping 4
2.1.5 Routing algorithm 5

3 Thesis Plan **7**
3.1 Introduction 7
3.1.1 A Review of Quantum hardware 7
3.1.2 Qubit routing approaches 8
3.2 Related work 8
3.2.1 Qubit routing reductions 8
3.2.2 Qubit routing analysis 9
3.2.3 Qubit routing solutions 9
3.3 Method 9
3.3.1 Terminological clarity 9
3.3.2 Problem modelling 10
3.3.3 Problem analysis 10
3.3.4 Solving the problem 10
3.4 Evaluation and discussion 11

Bibliography **13**

Chapter 1

Introduction

Qubit routing is a problem which emerged with the rise of NISQ devices and is a result of quantum hardware restrictions. The go-to model for planning and development of quantum algorithms was created by Deutsch [8]. Deutsch designed, having a quantum Turing machine in mind, a quantum circuit model, which is represented by a set of wires (each of them represents a quantum bit), and a set of quantum gates, which are ordered execution-wise. Every quantum gate can employ one or more wires (without any location requirements every combination of wires can participate in any gate). The only model restriction is that qubit cant be engaged in multiple operations, represented by quantum gates, at the same time. And from this unbounded representation comes the qubit routing problem quantum hardware is not equivalent (and probably cant) to the circuit model. In modern quantum hardware, qubits can interact in multi-qubit gates only with a fixed set of physical qubits. However unwanted, this problem could be dealt with by the SWAP gate insertion, which effectively swaps two qubits; and if we assume all-to-all qubit connectivity (more formal definition would be if the hardware topology graph is represented by one completely connected component), then this whole problem is boiled down to the efficient swap gates insertion, thats why sometimes qubit routing problem is called swap insertion. Our work aims to overview the field of qubit routing, to present a study of a problem, and to formulate a solution to it.

Chapter 2

Related works

As a good introduction to the problem, we present you a detailed review of an important paper by Cowtan [7]. Cowtan formalised the qubit routing problem, and suggested an heuristic solution to it.

2.1 Cowtan's solution

Cowtan and the team have presented a way to abstract mathematically the qubits routing, and then use a basic heuristic to solve it. They start by stating that the current quantum architectures are quite distant from the traditional von Neumann architecture, since qubits usually need to be neighbouring to perform operations on them. This is done by inserting swap operations to transfer qubits to different places of the circuit. This problem is known to be infeasible, and there are some go-to solutions available, the authors state. In the tket (Cambridge quantum compiler), both the translation and the routing to the given set of hardware qubits are implemented.

2.1.1 Problem statement

The authors then represent a quantum computer as a graph, where the nodes are qubits and edges are possible interactions of the qubits. The quantum circuit model by default is a fully connected graph; thus, they needed to produce a quantum circuit respecting the hardware constraints. As an example of a ring hardware connectivity graph, it is shown that it is beneficial to slice the circuit by the timesteps first. Then the authors suggest mapping the nodes to qubits in a smart way. Finally, for every timestep, you check if there is a two-qubit gate on non-connected qubits. In such a case, you insert a swap gate chain to get to the required neighbouring pair.

The authors state that the task is bounded by n^2 swap operations, going from one combination to any other. A recent study has shown that it can be

decreased to $\log n$ [9]. But it is obvious that it is needed to map the circuit globally and this one was per one timestep.

Another mentioned problem type is the fact that swap gate, being more or less an elementary operation, can be problematic on some hardware architectures. In example, the superconducting ones only have one elementary building block, so an ansatz is required. Also, for IBM architectures sometimes only one directional CNOT gates are available.

2.1.2 Algorithm overview

After that, the routing process is described by the authors. They start with an incomplete initial mapping of hardware qubits, and then go through all the timesteps. In a given timestep, if they approach yet not mapped qubit they add it to a nearest neighbour of its two-qubit gate vis-à-vis. If after this operation all the constraints are satisfied, they proceed to the next slice, otherwise, they need to start building the chain of SWAP gates.

They then present an overview of the algorithm, namely: 1) the timestep decoposition, 2) an initial placement determination, 3)the routing itself, and 4) a cleanup phase.

2.1.3 Circuit slicing

The cutting of the circuit into timesteps is a greedy algorithm, which slices the circuit by the two-qubit gates. If a two-qubit gate requires the qubit to be present, which is already participating in the current timestep, then a new timestep is formed. Authors also present a density metric, which represents the fraction — how many two-qubit gates out of the maximum possible one happen in the timestep. They then remark that they are okay with sparsity, but there is a possibility of constraining the sparsity by this density metric.

2.1.4 Initial mapping

The next phase is initial mapping: it is suggested to be constructed by a heuristic analysis, iterating through timesteps, an edge between two qubits only if they interact in this timestep, and for both of them, it is less than the third interaction in the whole circuit. As a result of this operation, authors claim that there can be either a ring or a line shape of the initial-mapping graph. The ring is broken then by removing a random connection. That way of initial mapping guarantees that most of the first two interaction gates (except that one which was chosen to break the ring) are happening without any swaps. After this, you select a subgraph of the hardware connectivity graph with the highest average degree and lowest diameter, and then map your mapping as a one long line, starting from the highest degree hardware qubit, and moving to the next highest degree qubit from there. If the initial

mapping is impossible on the given architecture, you can split the mapping line into several line segments.

2.1.5 Routing algorithm

The routing algorithm also runs step by step. It is suggested to provide the initial mapping and the sliced circuit as an input to the algorithm. If there are two-qubit gate qubits in the timestep, which are yet to be mapped, they are just mapped to a nearest mapped gate neighbour. All the one-qubit gates and the executable two-qubit gates are immediately removed from the sliced circuit. Then the swap-distance vectors are constructed. Authors suggest starting from the set of all available swaps in the circuit; then, you take all swaps that maximally reduce the distance to the circuit being executable. They repeat the construction until the path is unique, or just select a random one in a cutoff fashion. Authors argue that the diameter of the graph should strictly decrease under this construction.

Sometimes the algorithm gets stuck due to high symmetricity of the hardware connectivity graph: then it is suggested to try to have two-swaps in the construction phase; or even a brute force full longest two-qubit path once can be applied.

In practice, they do the slicing dynamically, so no pre-slicing of the circuit is needed. Also, at the end of the routing, a small clean-up pass happens either to get a hardware mapping or to remove excessive gates.

Chapter 3

Thesis Plan

In short, this master's thesis should present the problem of qubit routing. First, we expect to explain the problem by listing the connectivity requirements of the modern hardware, and then showing the existing solutions to the problem. Then we expect to outline the most prominent ways of the algorithmic analysis of the problem, the main ways of problem trivialisation (i.e. reduction to a known subproblem) and the known heuristic solutions to the problems, especially those with the components of the serious theoretical treatment. As a main body of work, we aim to present the terminological overview of the subfield, present a complete mathematical model of the problem, and then a solution to this mathematical construction. A final step would include an evaluation of the presented solution, and the benchmarking and comparative analysis in the contemporary state of the field.

3.1 Introduction

In this part we would like to discover all the available solutions and analyses in the field, as well as the state of the hardware connectivity.

3.1.1 A Review of Quantum hardware

In this section, we would like to present the known hardware implementations of the quantum computers; namely, the topology of the connectivity properties is of interest. We would like also to present some insights about possible modifications of the quantum qubit routing problem — see the neutral atom [16], or race track [13] trapped ion architecture, which are an unorthodox approach to the problem if we'd compare it with other contenders (the whole notion of the transformation of the qubit connectivity space is problematic for our suggested analysis). Thus, we'd like to convey as much information as possible on the hardware implementation topics. Well-known quantum architecture topologies include the IBM hexagonal [1]

and Google bowtie grid [3]. The additional complexity of this part arises from the fact that the implementation is often a part of a trade secret, and only singular examples of such unveil exist. As an output of this section, we would like to present the extensive review of the most prominent quantum hardware implementations.

3.1.2 Qubit routing approaches

The problem of qubit routing is natural to the way modern NISQ devices are implemented. Thus, a lot of possible solutions to the problem were created in the last decade. The first of the modern outlooks and attempts of formalisation of the problem is the [7], presented in detail in the related work section. It presents both a way to analyse the problem (both token swapping and graph isomorphism ways of thinking), a way to solve the problem, and a set of benchmarks. In this section, we would like to present all the aforementioned elements of the successful work, collecting the information about the methods of analysis, ways of solving the problem and benchmarking. We would like to deeply explore the SABRE family of algorithms [11]—which are considered state-of-the-art, and used in the IBM Qiskit compiler set. A central paper in the algorithmic sense is the [10], which provides a rich example of how to model the qubit routing problem, with a huge set of reasonable conjecture results. Finally, there are some algorithms designed specifically for an architectural paradigm, like [17]. It would be good to outline their ideas and present a compatibility study of them. A good result of this section would be to create a thorough understanding of the modern qubit routing solving approaches.

3.2 Related work

This section is devoted to more particular analysis of solutions to the problem.

3.2.1 Qubit routing reductions

One of the most prominent ways of solving the problem efficiently is reducing it to a well-studied type of problem, and then use an existing problem solver, or enhance the analysis by this inclusion. Some attempts of doing so for the qubit routing problem were made by [5]; some other papers include reduction to MaxSat or token swapping. Some alternative well-known graph structures were suggested: cyclic butterfly, hypercube and others. This section aims to provide an overview of trivialisations, their pros and cons, effects, and possible modifications or reductions to be made.

3.2.2 Qubit routing analysis

In this section, the most looked upon part in the whole qubit routing task will be overviewed. The algorithmic analysis of the problem suggests analysing various problem formulations, problem solutions and the characteristics of these solutions. An excellent example of such analysis is presented in [10]. We aim to collect and entwine the analyses (often incomplete) and present a set of possible problem formulations, useful conjectures and lemmas.

3.2.3 Qubit routing solutions

This section will outline the most important heuristic and algorithmic approaches to solving the routing problem. As even the problem formulation is different in many sources, this will require splitting the existing solutions into hierarchical components (responsible for different parts of the problem) and matching those together. A good overview of the contemporary is available at ([4]). There are many sources of confusion in the literature, for example, some are differentiating so-called qubit mapping and qubit routing (that wouldnt be a problem, but this sometimes acts as a complexity well analysing only one part, but moving a lot of steps in the other). Most of the approaches follow the timestep slicing as a main technique, and dynamic analysis of each timestep, like [7]. However, there are also machine learning solutions ([14]), reductionist solutions ([12]) and many other groups. An ideal outcome of this section would be a solutions taxonomy, their shared characteristics and different approaches.

3.3 Method

This part will be related to our approach to solving the qubit routing problem.

3.3.1 Terminological clarity

This section would provide an overview of the employed terminology, marking the start of creative contributions to the field. The terminological situation in the qubit routing (as youve probably noted in the previous section) is often a source of confusion. What is [17] names allocation, is routing in [7], swap insertion in [10] etc. The problem statements (e.g. some include connectivity graphs in statement, some just a general overview), operations definitions (do we include the one-qubit gates, how do we define a timestep) are only a small fraction of terminological problems. The ideal result of this section would be a simple guideline for qubit routing problems, terminology and interface between different schools and research groups.

3.3.2 Problem modelling

This chapter would outline our understanding of the qubit routing problem, providing a robust mathematical description of the problem. We plan to take the best of both worlds — theoretical and practical — and formulate the problem most clearly and concisely. Finding the theoretical representation of the problem is the demand of the analytical part of our work, and finding an effective practical translation of it is the demand of the practical part. Therefore, this part is also amplified since it lays the foundation of future chapters: solving the problem, evaluating the result and others. As a small insight into the planned approach, we had an idea of using Cayley diagram [2] representation of the problem as the theoretical basis of the analysis. However powerful, it is also exponential in the qubit count; thus, for actually solving the problem, it can be rendered highly impractical. Maybe some composite solution would be possible, however, the other problem would be — if we hurt the analytical stiffness of the problem statement, we might lose all the expressive power of our framework. The ideal output of this chapter would be a robust mathematical statement of the problem and its practical application.

3.3.3 Problem analysis

This section aims to present an all-sided analysis of the qubit routing problem. First part of it algorithmic analysis, similar to the approach suggested by [10]. Wed like to analyse the approach in the lenses of our problem statement, optimally obtaining the bounds for the algorithmic complexity, and analysis of the problem hardness for different classes of hardware and circuit layouts. The possible reductions of the problem will also be considered (as in MaxSat [12]), as a path to use a bigger class of universal algorithmic solvers. An ideal result of this chapter would be complete results about the algorithmic properties of our problem model and possible reductions of the problem to other problems.

3.3.4 Solving the problem

This is the most important part of the whole thesis. Here we would like to outline the algorithm for solving the problem stated and analysed in the previous chapters. As it was outlined in the previous sections, the approach to problem solving, although dependent on the analysis, could also be heuristic, reductionist or based on a problem extension. It would be good to present multiple approaches, as all of the aforementioned ways of solving have their differences, e.g. heuristic methods are usually robust and easy to implement, see [18]. If we obtain a new reduction model, well use an existing solver and explain why we used it. If it will be an extension of a problem, e.g. machine learning approach, then well show the intuition behind the model, and the

cost of this extension. Nevertheless, an ideal result of this chapter would be at least one approach to solving the presented problem, with an analysis (if possible) and its motivation.

3.4 Evaluation and discussion

The last planned part is an evaluation part. We want to analyse our problem solution from multiple sides. First, a comparison of analytical results with state-of-the-art solutions would be needed, including time complexity, scaling analysis, etc. Second, a practical benchmarking should be applied, a set of useful benchmarks could be found, for example in [7]. The ablation study of the solution of the problem is also desired, to see which parts affect the performance of the algorithm; also, verification of theoretical results is required. An analysis of the comparative performance could be a good addition to this section, together with a scalability study. An ideal output of this section would be plots of scalability, and a table of benchmark performance.

Bibliography

[1] The IBM Quantum heavy hex lattice | IBM Quantum Computing Blog. https://www.ibm.com/quantum/blog/heavy-hex-lattice.

[2] S.B. Akers and B. Krishnamurthy. A group-theoretic model for symmetric interconnection networks. *IEEE Transactions on Computers*, 38(4):555–566, April 1989.

[3] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Burkett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowler, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Mike Lindmark, Erik Lucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xiao Mi, Kristel Michielsen, Masoud Mohseni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, October 2019.

[4] Harrison Barnes. A Survey of Qubit Routing Algorithms. *Theses*, April 2023.

[5] Stephen Brierley. Efficient implementation of Quantum circuits with limited qubit interactions, September 2016.

[6] A. Chi-Chih Yao. Quantum circuit complexity. In *Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science*, pages 352–361, Palo Alto, CA, USA, 1993. IEEE.

[7] Alexander Cowtan, Silas Dilkes, Ross Duncan, Alexandre Krajenbrink, Will Simmons, and Seyon Sivarajah. On the qubit routing problem. *LIPIcs, Volume 135, TQC 2019*, 135:5:1–5:32, 2019.

[8] David Elieser Deutsch. Quantum computational networks. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 425(1868):73–90, September 1989.

[9] Steven Herbert. On the depth overhead incurred when running quantum algorithms on near-term quantum computers with limited qubit connectivity, July 2020.

[10] Takehiro Ito, Naonori Kakimura, Naoyuki Kamiyama, Yusuke Kobayashi, and Yoshio Okamoto. Algorithmic Theory of Qubit Routing, May 2023.

[11] Gushu Li, Yufei Ding, and Yuan Xie. Tackling the Qubit Mapping Problem for NISQ-Era Quantum Devices, May 2019.

[12] Abtin Molavi, Amanda Xu, Martin Diges, Lauren Pick, Swamit Tannu, and Aws Albarghouthi. Qubit Mapping and Routing via MaxSAT, August 2022.

[13] S. A. Moses, C. H. Baldwin, M. S. Allman, R. Ancona, L. Ascarrunz, C. Barnes, J. Bartolotta, B. Bjork, P. Blanchard, M. Bohn, J. G. Bohnet, N. C. Brown, N. Q. Burdick, W. C. Burton, S. L. Campbell, J. P. Campora III, C. Carron, J. Chambers, J. W. Chan, Y. H. Chen, A. Chernoguzov, E. Chertkov, J. Colina, J. P. Curtis, R. Daniel, M. DeCross, D. Deen, C. Delaney, J. M. Dreiling, C. T. Ertsgaard, J. Esposito, B. Estey, M. Fabrikant, C. Figgatt, C. Foltz, M. Foss-Feig, D. Francois, J. P. Gaebler, T. M. Gatterman, C. N. Gilbreth, J. Giles, E. Glynn, A. Hall, A. M. Hankin, A. Hansen, D. Hayes, B. Higashi, I. M. Hoffman, B. Horning, J. J. Hout, R. Jacobs, J. Johansen, L. Jones, J. Karcz, T. Klein, P. Lauria, P. Lee, D. Liefer, C. Lytle, S. T. Lu, D. Lucchetti, A. Malm, M. Matheny, B. Mathewson, K. Mayer, D. B. Miller, M. Mills, B. Neyenhuis, L. Nugent, S. Olson, J. Parks, G. N. Price, Z. Price, M. Pugh, A. Ransford, A. P. Reed, C. Roman, M. Rowe, C. Ryan-Anderson, S. Sanders, J. Sedlacek, P. Shevchuk, P. Siegfried, T. Skripka, B. Spaun, R. T. Sprenkle, R. P. Stutz, M. Swallows,

R. I. Tobey, A. Tran, T. Tran, E. Vogt, C. Volin, J. Walker, A. M. Zolot, and J. M. Pino. A Race Track Trapped-Ion Quantum Processor. *Physical Review X*, 13(4):041052, December 2023.

[14] Matteo G. Pozzi, Steven J. Herbert, Akash Sengupta, and Robert D. Mullins. Using Reinforcement Learning to Perform Qubit Routing in Quantum Compilers, July 2020.

[15] J. Von Neumann. First draft of a report on the EDVAC. *IEEE Annals of the History of Computing*, 15(4):27–75, 1993.

[16] Hanrui Wang, Pengyu Liu, Daniel Bochen Tan, Yilian Liu, Jiaqi Gu, David Z. Pan, Jason Cong, Umut A. Acar, and Song Han. Atomique: A Quantum Compiler for Reconfigurable Neutral Atom Arrays, November 2024.

[17] Robert Wille and Lukas Burgholzer. MQT QMAP: Efficient Quantum Circuit Mapping. In *Proceedings of the 2023 International Symposium on Physical Design*, pages 198–204, Virtual Event USA, March 2023. ACM.

[18] Henry Zou, Matthew Treinish, Kevin Hartman, Alexander Ivrii, and Jake Lishman. LightSABRE: A Lightweight and Enhanced SABRE Algorithm, September 2024.

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

ON QUBIT ROUTING IN QUANTUM COMPUTER ARCHITECTURES

Authored by (in block letters):

For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
MARK SOBOLEV	

With my signature I confirm that

- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date

21.05.25

Signature(s)

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.