# Quantum Circuit Optimilization Using Monte Carlo Tree Search and Reinforcement Learning

Author: Liza Darwesh
liza.darwesh@hva.nl

Supervisors:
Ariana Torres, SURF
Damian Podareanu, SURF
Dick Heinhuis, HvA

Hogeschool van Amsterdam

A Preliminary Bachelor's Thesis
Graduation Assignment for SURF BV
Department of Innovation
Amsterdam University of Applied Science

April 3, 2022

# Chapter 1

# Introduction

Quantum computers are expected to make a breakthrough in chemical and biological engineering through the discovery and manipulation of molecules, encryption for cybersecurity, the processing of very large quantities to aid in artificial intelligence and the pricing of complex assets in finance [1]. The quantum computers from nowadays may be able to perform tasks which surpass the capabilities of today's classical computers. Yet the world is not there yet.

Decoherence is a phenomenon that must be taken into account. Think of a spinning coin with information: either heads of tails. Eventually the coin will stop spinning and land on one of the sides. This is the same for a quantum system. The algorithms that are written for a quantum computer contain noise, which can ensure that the information provided is unreliable. The way algorithms can be operated is by writing them into quantum gates. They are the same as the classical logic gates, but with quantum phenomena, which will be explained later in the paper. So basically, mathematical algorithms are transformed into circuits of operations. These circuits are written in Quantum Assembly, a language that the quantum computer understands. Each circuit has a defined number of logical quantum bits (qubits). These qubits are what the gates are interacting with. In the moment of processing, the logical qubits in the circuit will then be mapped into the topology of the quantum computer, which is a connectivity architecture of the qubits inside the quantum computer, also refereed as physical qubits. The problem here lies in the connectivity matching. When programming a circuit, one must satisfy the topology. For instance, say two qubits need to interact with each other through a gate. If these qubits are not connected to each other, this operation will be inoperable, which means the whole circuit cannot be executed. So there needs to be a way to make circuits operable while satisfying the connectivity constraints of the quantum computer. A way to do this is by adding a so called SWAP-gate in the circuit, that causes the qubits to flip. This flip ensures that the qubits come closer to each other or even become connected. The problem with the swap gate is that it is not a known logic gate for the quantum computer. The swap gate is performed by placing 3 CNOT gates one behind the other. This ensures that the circuit becomes larger in depth and therefore also more sensitive to decoherence. So there needs to be an optimization in placing these SWAP gates to ensure that the information is reliable.

The research of this thesis is done for the organisation SURF. A cooperative association of Dutch educational and research institutions. Universities, universities of applied sciences, MBO institutions, UMCs and research institutes work together within SURF to purchase or develop the best possible digital services. The company owns supercomputers which are available to complete high performance processes. Within the company there are multiple departments including quantum innovation, machine learning, and high performance computing. Which will be useful for this research.

Currently, there are already a number of studies done in this area, for example a state-of-the-art research was the one of Pozzi, Herbert, Sengupta and Mullins [2], where he used Reinforcement Learning and something called Quantum Annealing to optimize the number of SWAP-gates in a circuit. His research was based on a random logical qubit allocation on the physical qubits. When creating an initial qubit placement, the number of SWAP-gates can be reduced. Another recently published study is done by Sinha, Azad and Singh [3], where they managed to minimize the number of SWAP gates for a random qubit allocation using Monte Carlo Tree Search in combination with Reinforcement Learning.

Both papers discussed the need to look into initial qubit placement, which will also ensure that the SWAP gates are minimized in addition to using Reinforcement Learning. With this in mind, research will be conducted on quantum circuit optimization using the Monte Carlo Tree Search in combination with Reinforcement Learning. An initial qubit placement will be used for this.

This paper answers the main question: 'How can quantum circuit routing be optimized using Reinforcement Learning applied on Monte Carlo Tree Search?' This question will be answered by means of the following sub-questions:

- What is quantum circuit routing?
- What has already been done to successfully accomplish quantum circuit routing?
- How can the initial qubit placement procedure be performed?
- How does the Monte Carlo Tree Search work on optimizing the number of SWAP-gates?

This research paper is organised as follows. Section ?? offers a description of basic knowledge about quantum computing phenomenon and techniques needed for understanding the main question. Section ?? demonstrates what research has already been done by others. Section ?? describes the sub-problem initial qubit placement and how to solve it. Section ?? provides an overview of the experimental setup, which is how the Monte Carlo Tree Search is put together with the Reinforcement Learning. Section ?? shows the results of what the reinforcement learning model delivered. Finally, in Section ?? and ?? states the discussion about the research that is performed and the final conclusion.

# Bibliography


[1] F. Bova, A. Goldfarb, and R. Melko, "Quantum computing is coming. what can it do?" Jul 2021. [Online]. Available: https://hbr.org/2021/07/quantum-computing-is-coming-what-can-it-do 1

[2] M. G. Pozzi, S. J. Herbert, A. Sengupta, and R. D. Mullins, "Using reinforcement learning to perform qubit routing in quantum compilers," Jul 2020. [Online]. Available: https://arxiv.org/abs/2007.15957 1

[3] A. Sinha, U. Azad, and H. Singh, "Qubit routing using graph neural network aided monte carlo tree search," Mar 2022. [Online]. Available: https://arxiv.org/abs/2104.01992 1