Outline

Eric Schinwald

27th February 2020

1 Abstract

Currently more and more Quantum Algorithms are being written for Quantum Computers. Most of these are still theoretical. They either use too many Qubits or too many gates which leads to a very high error rate. Some of the written algorithms can already satisfy those two constraints. There is however, another constraint which leads to almost no Quantum Circuits being able to be run directly. IBM QX architectures currently allows for CNOT gates only at certain Qubits. In order to overcome this limitation we can insert SWAP gates to satisfy the constraints given by the IBM QX architectures.

This leads us to an interesting question: Can we design an architecture in a way that benefits a certain class of Quantum Algorithms? Benefiting in this case would mean that we have to insert a lower amount of SWAP gates in order to satisfy the constraints given by the architecture. This is the question that this paper is going to try to answer.

First, we go over basic background knowledge in Quantum Computing that is required to understand this paper fully. Then we answer why it is important to design good architectures. In the implementation section we go into detail, how we divided existing Quantum algorithms into different sets to optimise architectures for. After that we will go into detail how design the architectures. After that we will have a section where we present the results of our generated architectures in terms of number of SWAP gates we have had to insert to make the Quantum Circuit theoretically run on our generate architectures. Lastly we will summaries our results.

2 Introduction

3 Background

4 Motivation

5 Implementation

5.1 Lessons learned

6 Summary

7 Related Work