

# COMP 4520 – Undergraduate Honours Project Proposal

## An analysis of computational complexity and implementation requirements of quantum graph algorithms on an Universal Quantum Computer

Abhishek ., 7803917

### I Abstract:

Recent breakthroughs in Quantum Computing has enabled public access to a wide array of actual quantum hardware and simulators. With the progress to achieving "quantum supremacy" **(1)(4)** being accelerated by tech giants such as Google, IBM and Microsoft, the need for algorithms to exploit the "quantum effects" of the hardware has emerged rapidly. In this research effort, we explore and evaluate the implementations of graph algorithms such as TSP and Dijkstra's Shortest Path on simulated and actual universal quantum computers, as well as determine changes to the computational complexity and runtime speedup provided by such implementations. We use IBM's Q Platform as a base for development and execution with the help of Qiskit, an open-source quantum computing framework in Python.

### II Introduction to the topic:

The basis of information processing in Quantum Computing lies in qubits. A qubit is two-state quantum mechanical system often described in Quantum Computing literature as a superposition in the standard basis  $\{0, 1\}$ . The following is a representation of the state of the qubit in the standard basis:

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle,$$

where  $\alpha, \beta$  are probability amplitudes with  $|\alpha|^2 + |\beta|^2 = 1$ .

When measured in the standard basis, the probability of the outcome  $|0\rangle$  with the measured value 0 is  $|\alpha|^2$ , and of the outcome  $|1\rangle$  with the measured value 1 is  $|\beta|^2$ .

While the vector spaces of classical objects combine through a direct sum ( $\oplus$ ), the vector spaces associated with quantum systems combine through the tensor product ( $\otimes$ ). For example, consider  $n$  classical objects with  $m$ -dimensional vector space, the joint state space of these  $n$  objects has  $m \cdot n$  dimensions whereas for  $n$  quantum systems associated with  $m$ -dimensional vector space, the joint state space has  $m^n$  dimensions, the state space of a quantum system grows exponentially. This means there are vastly more states available to encode the information in. This behaviour of large state spaces corresponding to small physical quantum

systems is at the center of the computational speedup exploited by quantum hardware, and therefore quantum algorithms.

A basic operation on the state of a qubit is implemented through a series of unitary transformation, where a unitary transformation is described by a matrix  $\mathbf{U}$  with complex entries, which satisfies the following condition:

$$\mathbf{U}\mathbf{U}^\dagger = \mathbf{U}^\dagger\mathbf{U} = \mathbf{I},$$

where  $\mathbf{U}^\dagger$  denotes the transposed complex conjugate matrix of  $\mathbf{U}$ , and  $\mathbf{I}$  denotes the identity matrix.

Such basic operations are referred to as gates, of which {H,T,CNOT} i.e. {Hadamard, Toffoli, Controlled Not} form a universal gate set.

As described in (2), a quantum algorithm comprises of 3 important steps 1. encoding of the data, 2. sequence of quantum gates applied to the set of input qubits, and 3. measurements of one or more of the qubits at the end to obtain a classically interpretable result.

From a classical computing point of view, graph problems such as TSP are NP-hard combinatorial optimization problems with various classical approaches such as branch-and-bound algorithms as well as heuristics and approximation algorithms being used to yield good solutions (3)

We will look at several of such optimization problems and try to determine and analyse the computational complexity of the quantum implementation of such algorithms and check whether if any such implementations would belong to the BQP ( bounded-error quantum polynomial time ) complexity class.

### **III Your background preparation to do this project:**

For the past 8 months, I have carried out independent research into the building blocks of Quantum Computing through textbooks and online resources ( theoretical ) and the frameworks and libraries available for the implementation of generic quantum algorithms ( e.g. Grover's Database Search, Shor's Factoring Algorithm ) on both actual quantum hardware, and simulators running on local and remote machines ( experimental ).

Courses taken for building a background for the project are following:

PHYS 2390: Theoretical Physics 1

Relevant learnings - Mathematical methods used for understanding and explaining the interaction between physical systems.

COMP 2140: Data Structures and Algorithms

Relevant learnings - Common combinatorial graph optimization algorithms and introductory algorithm analysis.

#### COMP 2380: Quantum Physics

Relevant learnings - Mathematical methods associated with quantum systems. Proofs related to observations from quantum systems with no classical counterpart.

#### PHYS 2490: Theoretical Physics 2

Relevant learnings - Solutions to ordinary and partial differential equations in classical and quantum physics, introductory probability theorems and common distributions.

#### COMP 2080: Analysis of Algorithms

Relevant learnings - Advanced algorithm analysis using methods for analyzing the time and space requirements of the algorithms, exploring different models of computation, algorithm design using different strategies ( e.g. greedy, randomized algorithms )

#### PHYS 3380: Quantum Mechanics 2

Relevant learnings – Developing abstract linear algebraic formalism for quantum mechanical states and operators with classical measurements, solutions to 1-dimensional and 3-dimensional Schrodinger's equations for the hydrogen atom, degeneracy in quantum states.

### **IV Related work:**

Most of the initial progress in the implementation of quantum algorithms for combinatorial optimization problems was based on an adiabatic quantum computing model and the algorithms were implemented and tested on quantum annealers ( actual and simulated ) **(5) (6) (7)**. However, the quantum annealer is the least powerful and most restrictive form of quantum computers with no known advantages over conventional computing with the only known application being to optimization problems.

To the best of my knowledge, the research into the development of optimization algorithms for universal quantum computer hardware is still in very early stages, particularly due to the impending restriction on the number of qubits being available of both the simulators and the developed quantum hardware.

The groundwork for building such algorithms have been laid by the development of quantum algorithms that are quadratically or exponentially faster than the theoretical limit for classical algorithms such as Grover's search algorithm **(8)** and Shor's factoring algorithm **(9)**. Many optimization and quantum machine learning algorithms rely on Grover's search or one of its variants. We would use the corresponding pre-defined operators and call oracles if necessary for the development and implementation of the quantum graph algorithms.

## **V Problem Statement:**

Problem: What is the feasibility and computational complexity of quantum graph algorithms, and how much resources are required to develop, implement, and execute such algorithms on actual and simulated universal quantum computers ?

Goal: Thorough analysis of the implementation, deployment and execution of the developed algorithms.

## **VI Methodology:**

The methodology used will be adapted from my first co-op term at Online Business Systems which I have found to be very useful to stay on track as well as quantify any progress made. The first step will be to organize an HMW ( How Might We? ) session with any individuals involved in this research effort to determine the major stages and checkpoints in the project as well as identify any foreseeable challenges and raise objections to the deliverables set forward.

The project vision will be divided into a series of iterative step-projects, one building upon the other. The series is dynamic and will evolve as any unforeseen challenges emerge. Here is an initial overview of what the step-projects will look like :

Step #	Step Project overview	Predicted completion dates
1	Determination of graph algorithms most viable for implementation on a universal quantum computer. Development and research into improving the efficiency and complexity of quantum algorithms.	September 30, 2018
2	Setup of development and execution environments for the quantum simulators and actual machines ( including any optimizations to the simulators inferred from the design of the quantum graph algorithms such as gate delays, transformation sequencing etc. )	October 20, 2018
3	Execution and analysis of the results from the executions. Research into computational complexity classes and determining the computational complexity and feasibility of the implementations	November 10, 2018
4	Retrospective and compilation of the findings and results including meeting reporting requirements for the course as well as analysis of the goals met, and challenges faced.	December 05, 2018

In addition, each of the step-projects will be accompanied by an initial objective brief and an ending retrospective document compiling and addressing the progress made during the step-projects.

## **VII Infrastructure and facilities and expert personnel required:**

The following are the initial resources expected for successful completion of the project:

### Computing Resources

1. Cloud Access and research priority to the IBMQ5, IBMQ16 and IBM Q20 machines for the implementation, testing and analysis of the quantum algorithms on actual quantum hardware. ( Due to the restrictiveness of the access to the actual quantum hardware, the following could be fulfilled using simulators on high performance machines with the loss of analytics corresponding to the execution of the algorithm on the actual hardware. )

### Academic Resources

1. Access to articles and journals actively covering the most recent advancements in Quantum Computing ( partly fulfilled by UofM library access.)
2. Active correspondence with experts in the field of Quantum Algorithms and Computations as well as academic resources available in the computer science department at the University of Manitoba in the fields of high performance computing ( for universal quantum computer simulators ) and algorithm development ( for developing efficient algorithms for deployment ).

## **VIII Outcome and Deliverables:**

Outcome: Brief analysis of implementations of the graph algorithms on actual and simulated universal quantum computers including the deployment feasibility on actual quantum hardware, limitations due to error correction methods currently used, computational complexity of such algorithms and an overview of near-future applications.

Deliverables: Deployment of the theorized quantum algorithms on actual and simulated universal quantum computers. Retrospective of the important learnings worthy of publication and continued future research.

Note: The findings from this research effort will provide a very good head start for the proposed research project for COMP 4560 – Undergraduate Industrial Project in the field of Quantum Machine Learning.

## **References:**

- (1) Aaronson, S., Chen L., "Complexity-theoretic foundations of quantum supremacy experiments", DOI: <http://dx.doi.org/10.4230/LIPIcs.CCC.2017.22>
- (2) Coles, P.J. et al., "Quantum Algorithm Implementations for Beginners", arxiv: <https://arxiv.org/abs/1804.03719>.
- (3) Rego, C. et al. (2011), "Traveling salesman problem heuristics: leading methods, implementations and latest advances", DOI: <https://doi.org/10.1016/j.ejor.2010.09.010>
- (4) Calude, C.S. , Calude E., "The Road to Quantum Computational Supremacy", arxiv: <https://arxiv.org/pdf/1712.01356>
- (5) Martonak, R. et al., "Quantum annealing of the traveling-salesman problem", doi: <https://doi.org/10.1103/PhysRevE.70.057701>
- (6) Stella, L. et al., "Optimization by quantum annealing: Lessons from simple cases", doi: <https://doi.org/10.1103/PhysRevB.72.014303>
- (7) McGeoch, C.C., Wang C., "Experimental evaluation of an adiabatic quantum system for combinatorial optimization", doi: <https://doi.org/10.1145/282767.2482797>
- (8) Grover, L.K., "A fast quantum mechanical algorithm for database search", arxiv: <https://arxiv.org/abs/quant-ph/9605043v2>
- (9) Shor, P.W., "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer", arxiv: <https://arxiv.org/abs/quant-ph/9508027v2>

## **Textbook references:**

- i. Rieffel, E., Polak, W., "Quantum Computing : A Gentle Introduction", doi: <https://doi.org/10.1063/PT.3.1442>
- ii. Wittek, P., "Quantum Machine Learning", doi: <http://dx.doi.org/10.1016/B978-0-12-800953-6.00002-5>