COMP 4520 : Undergraduate Honours Project
Final Report

Introduction to Quantum Computing and the Single Pair Shortest Path Problem

Abhishek .
Department of Computer Science, University of Manitoba
Email : abhishek@myumanitoba.ca

Supervisor : Dr. Parimala Thulasiraman
Department of Computer Science, University of Manitoba
Email : thulasir@cs.umanitoba.ca

December 20, 2018

Abstract

The purpose of this project was to study in detail the background concepts and the recent advancements in the field of Quantum Computing and Quantum Algorithms. More precisely, we explored how concepts from general Quantum computing algorithms such as Quantum Search can be applied to computational problems in order to improve the time and query complexity of Optimal Classical Algorithms. We considered one of the most common graph optimization problems i.e the Single Source Shortest Path problem and determined how approaches from Quantum algorithms have been used to improve the efficiency of Dijkstra's Shortest Path Algorithm in the Quantum Model of Computation. We also present a discussion on how quantum algorithms can be used to improve classical algorithms and discuss an approach to solving the shortest path problem.

Acknowledgements

I would like to thank my supervisor, Dr. Parimala Thulasiraman for her useful guidance, comments and expert knowledge throughout the project. I would also like to thank the Undergraduate Honours Project Coordinator, Dr. Ruppa K. Thulasiram for allowing me to have this amazing opportunity to learn the important concepts and explore the recent advancements in the field of Quantum Computing and Information.

Contents

1 Introduction

Quantum Computing and Quantum Information is the study of information processing techniques through the use of quantum mechanical systems. The origins of the field have been followed back to early 1980s to work by pioneers such as Paul Benioff [1] and Richard Feynman [2]. In the recent decades, development of Quantum Algorithms such as Shor's Polynomial-Time Factoring [3] and Grover's Search [4] has generated a wide interest into finding efficient Quantum algorithms for problems which are classically computationally intractable due to either time or query complexity.

Graph theory and algorithms is also a very active research area in Computer Science with a wide range of real-world applications. There have been numerous advancements and development in this field and classically optimal algorithms such as Dijkstra's algorithm for finding the shortest path and Prim's algorithm for finding the minimal spanning tree in a graph have been devised.

Quantum computing models have been considered as good candidates for researching improvements to the efficiency and complexity of these algorithms. Many new quantum algorithms have been devised in the recent years [5] to provide efficient solutions to computational graph problems.

Throughout the report and in the sections below, we discuss the foundations of the field of Quantum computation that we studied in detail. We determine the characteristics of the quantum model of computation and the general techniques and approaches which are used to devise efficient quantum algorithms. We also consider the graph optimization problem of single pair shortest path and analyze how techniques from quantum computing have been used to solve this problem and how they differ from the classical algorithms. We conclude by presenting a synthesis on, how can quantum computing be used to improve existing classical algorithms and what other approaches can be used to solve the shortest path problem.

2 Theoretical Background

In the following sections, we discuss and explain the fundamental concepts learned over the duration of the project in order to build an understanding of the underlying theories of Quantum computing, Computational graphs and Quantum Parallelism which guide the research and algorithm development in the fields. We also present ideas on how these fundamental concepts make quantum computing different from classical computing.

2.1 Quantum Computing Fundamentals

In this section, we discuss the concepts of Quantum computing which were learned over the course of the project which are essential to understanding the current works in this field.

2.1.1 Quantum States

A bit is the fundamental unit of information in classical computing. It can hold the value i.e. have the state of either 0 or 1. In quantum computing, a qubit is defined as the fundamental unit of information. The two basis states used for the qubit in information processing are $|0\rangle$ and $|1\rangle$ which correspond to the states of 0 and 1 in classical computing.

The difference between the classical bit and qubit which is at the center of the field of quantum computing is that the qubit can exist in a linear combination of the basis states. The state $|\psi\rangle$ of a single qubit can be defined as :

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \tag{1}$$

where α and β are complex numbers satisfying the condition :

$$|\alpha|^2 + |\beta|^2 = 1 \tag{2}$$

In general, a qubit is a two-dimensional unit vector with complex amplitudes in the basis states.

A multiple qubit system is described by the amplitudes in the basis states which are the combined states of the component qubits. For example, a quantum state ϕ of 2 qubits has the basis $\{00, 01, 10, 11\}$. The state of ϕ is then given as :

$$|\psi\rangle = \alpha |00\rangle + \beta |01\rangle + \delta |01\rangle + \gamma |11\rangle \tag{3}$$

The state of multiple qubit systems highlights another important difference that has contributed to widespread interest in quantum computing. For a classical system, the state space grows linearly with respect to the number of components in the system whereas the state space of a quantum system grows exponentially with the number of components. This means that there are exponentially more states in a quantum system to encode information than in a classical system.

The difference in the state spaces of classical and quantum systems can be explained by the operations through which the vector spaces combine. Consider a classical computing system ϕ with 3 bits: A, B, and C, the state space of this system is the direct sum, $\bigoplus$ of the vector spaces for each of the 3 bits. The vector space for each of the 3 bits is defined by $\{0, 1\}$. The vector space of ϕ is then given by :

$$\begin{pmatrix}0\\1\end{pmatrix}_A \bigoplus \begin{pmatrix}0\\1\end{pmatrix}_B \bigoplus \begin{pmatrix}0\\1\end{pmatrix}_C = \begin{pmatrix}0_A\\1_A\\0_B\\1_B\\0_C\\1_C\end{pmatrix} \tag{4}$$

Therefore, a classical computing system with n bits has the state space of dimensions $2n$. On the other hand, consider a quantum system, ψ with 3 qubits: A, B, and C, the state space of this system is the tensor product, $\bigotimes$ of the vector spaces for each of the 3 qubits.

$$\begin{pmatrix}0\\1\end{pmatrix}_A \bigotimes \begin{pmatrix}0\\1\end{pmatrix}_B \bigotimes \begin{pmatrix}0\\1\end{pmatrix}_C = \begin{pmatrix}0_A0_B0_C\\0_A0_B1_C\\0_A1_B0_C\\0_A1_B1_C\\1_A0_B0_C\\1_A0_B1_C\\1_A1_B0_C\\1_A1_B1_C\end{pmatrix} \tag{5}$$

Therefore, a quantum computing system with n qubits has the state space of dimensions 2^n.

This suggests that quantum computing offers a more efficient model of computation which can exploit large state spaces corresponding to relatively small physical systems.

2.1.2 Quantum Gates

In the circuit model of classical computation, circuits are composed of wires and logic gates such as NOT, AND, OR etc. The voltage is the wires is used to represent classical bits of information i.e. a higher positive value of the voltage represents a '1' and a lower positive value near the ground potential of 0 V represents a '0'.

In the circuit model of quantum computation, the gates can be represented as unitary transformations that act on the quantum states of the qubits. There are various single, double and higher order gates in the circuit model of quantum computation.

The Hadamard gate is one of the most common quantum gates used in quantum algorithms. The unitary matrix defining the Hadamard gate is given by :

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix} \tag{6}$$

The Hadamard gate can be used to transform a single qubit in one of the basis states into a superposition in the basis states and vice-versa. For example, consider a qubit $|\psi\rangle$ in the state $|0\rangle$:

$$\psi = \begin{bmatrix}1\\0\end{bmatrix} \tag{7}$$

The Hadamard gate acting on $|\psi\rangle$ then transforms $|\psi\rangle$ into a superposition in the basis states $\{0, 1\}$.

$$H|\psi\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}1\\ 0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\ 1\end{bmatrix} \tag{8}$$

The condition that the quantum gates are unitary implies a reversible model of computation which adds additional functionality to the circuit model as well as new challenges in designing quantum circuits. This means the effect of a quantum gate applied to a quantum state can be reversed by applying the gates twice.

This is also a highlighting difference between quantum computing and classical computing which implies that quantum computing offers a more general model of reversible computation, and therefore should offer more techniques and approaches to solve problems than classical computing.

2.1.3 Quantum Measurement

The measurement gate often denoted M is used to perform measurement on a single qubit. Its operation on a single qubit, ψ is defined as the following :

Consider a qubit in the state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, then the measurement gate in the computation basis $\{0,1\}$ acting on $|\psi\rangle$ gives the output 0 with probability α^2 and 1 with probability β^2. The state of ψ collapses to one of the basis states corresponding to the output of the measurement.

The measurement gate is usually used at the end of a quantum circuit after information processing on a set of n qubits have been completed using the unitary transformations described by reversible quantum gates.

2.2 Quantum Parallelism

In order to understand how numerous quantum algorithms offer time or query complexity advantages over optimal classical algorithms, it is important to understand the concept of quantum parallelism. We studied the theory of quantum parallelism in detail in order to understand how these algorithms exploit the quantum mechanical properties of Quantum computers.

We discuss the quantum mechanical phenomena which lead to quantum parallelism and an algorithm which exploits the quantum property of interference between qubit states in order to achieve quantum parallelism.

2.2.1 Introduction to Quantum Parallelism

Quantum parallelism is a fundamental property of many quantum algorithms. In simplified terms, it allow for quantum computer to evaluate a function $f(x)$ simultaneously for multiple values of x which are encoded in the qubit states. [6]

Quantum parallelism relies greatly on the quantum phenomenon of superposition of the qubits as shown in Equations (1) and (3).

The simplest example to understand how superposition allows for quantum parallelism is given below :

Consider a 2-qubit quantum computer with the initial state $|\psi\rangle = |0\rangle|0\rangle$. Consider a binary function operator f(x) with one bit domain and range i.e. f(x) : $\{$0,1$\} \mapsto \{$0,1$\}$. It can be shown that a unitary operator, U_f can be constructed using quantum gates where the operation of U_f on $|\phi\rangle = |x\rangle|y\rangle$ is defined by :

$$U_f |\phi\rangle = U_f |x\rangle |y\rangle = |x\rangle |y \oplus f(x)\rangle \tag{9}$$

Consider we apply the Hadamard gate (Equation 8) to the first qubit of $|\psi\rangle$, the application of the Hadamard gate, H acting on $|0\rangle$ generates a superposition of the single qubit i.e.:

$$H |0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \tag{10}$$

The state of the 2-qubit computer can now be described as :

$$|\psi\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \bigotimes |0\rangle = \frac{|00\rangle + |10\rangle}{\sqrt{2}} \tag{11}$$

The operation of U_f on the $|\psi\rangle$ is now defined by :

$$U_f |\psi\rangle = U_f \frac{|00\rangle + |10\rangle}{\sqrt{2}} = \frac{|0\rangle |0 \oplus f(0)\rangle + |1\rangle |0 \oplus f(1)\rangle}{\sqrt{2}} = \frac{|0\rangle |f(0)\rangle + |1\rangle |f(1)\rangle}{\sqrt{2}} \tag{12}$$

The above example shows how the superposition of a single qubit can be used to evaluate the function $f(x)$ over both 0 and 1 using only two operators, H and U_f. Although, this information is not directly useful through measurement, this approach of using quantum phenomena such as superposition, entanglement and interference is the foundation of quantum parallelism.

2.2.2 Deustch-Jozsa Algorithm

The Deustch-Jozsa Algorithm [7] is one of the first quantum algorithms which were determined to be exponentially faster than any classical algorithm. Although the problem solved by the Deustch-Josza algorithm was of little practical significance, the algorithm was essential to generating wide-spread interest in quantum computing and paving way for other more practical and useful exponentially and quadratically quantum algorithms to be developed later.

The Deustch-Jozsa Algorithm uses the property of quantum interference to solve a type of hidden subgroup problem where the goal is to determine whether a binary function $f(x)$ is balanced or constant. The classical limit of this problem is $O(2^n)$ queries whereas the Deustch-Jozsa Algorithm algorithm offers a query complexity of $O(1)$.

This represents the massive potential that quantum model of computation offers to solve present classically computationally intractable problems.

2.3 Graph Theory and Query Models

In this section, we discuss the basic computational theory of graphs and the query models commonly used in the quantum variants of the classical shortest path algorithm.

In computational graph theory, a graph is described by a set of vertices $V = \{0, 1, ..., n\}$ and a set of edges between the vertices $E = \{< i, j >\}$, where $i, j \in U$ and $U \subseteq V$. A strongly connected, un-weighted graph is then defined by $G = (V, E)$.

If the graph is weighted, then a cost function $C = (c_{ij})$ is used where c_{ij} is the weight of the directed edge from vertex i to vertex j and $c_{ij} = \infty$ is often used when there does not exist a directed edge from vertex i to vertex j. The weighted, directed graph G is then defined by $G = (V, E, C)$.

Two query models are generally considered when discussing the algorithms for graph optimization problems.

1. Adjacency Matrix Model : A graph with n vertices is described by an adjacency matrix M of size $n \times n$ where $M_{ij} = 1$ if and only if $< i, j > \in E$ and the graph is un-weighted or $M_{ij} = c_{ij}$ if and only if $< i, j > \in E$ and the graph is weighted.

2. Adjacency List Model : A graph is described by a collection of n lists where each list describes the set of neighbours for a vertex. For a weighted graph, each item in the list has an attached weight.

3 Quantum Algorithms

In this section, we discuss the quantum algorithms that were comprehensively studied during the project in order to understand the underlying principles of these algorithms and understand the approaches which can been applied to solve computational problems in other areas.

In general, there are 3 primary classes of quantum algorithms to which all efficient quantum algorithms which offer advantages over classical algorithms belong to [6]:

1. Quantum Search Algorithms : Quantum algorithms which use the technique of amplitude amplification and a query oracle from Grover's Search in order to improve the time and/or query complexity of classical algorithms. The theory and important concepts behind these algorithms are discussed in Section 3.1.

2. Quantum Fourier Transform : Quantum algorithms which use the quantum variant of the classical Fourier transform to provide advantage over classical algorithms which widely use the Fourier transform. It was shown in [3] by Peter Shor that quantum computers can be used to efficiently implement the Quantum Fourier Transform.

   The quantum fourier transform has a circuit complexity of $\Theta(n^2)$ which is exponentially faster than the classical algorithm of Fast Fourier Transform with a gate complexity of $\Theta(n2^n)$ [6].

   The fourier transform is used for the procedure of phase estimation which is at the center of many algorithms such as Shor's algorithm for factoring and order-finding [4].

3. Quantum Simulation : Quantum algorithms which exploit the natural behaviour of Quantum computers to simulate quantum mechanical systems.

The 3 classes of algorithms above comprehend most of the quantum algorithms which have been conceived to date.

3.1 Quantum Search Algorithms

One of the most important algorithms in Quantum computing is the Grover's search algorithm. It was originally devised by Lov K. Grover in 1995 and published in 1996 [4]. Many of the efficient quantum algorithms devised to date use the techniques from Grover's search algorithm in order to improve the query and/or time complexities of classical algorithms.

3.1.1 Grover's Search Algorithm

Grover's search algorithm presents a quantum approach to the problem of finding an item in an unstructured database. In the classical computing model, this has the query complexity of $O(n)$ where n is the number of items in the database. Grover's algorithm uses the technique of amplitude amplification to provide a quadratic speedup over the classical limit. The algorithm has the query complexity of $O(\sqrt{\frac{n}{m}})$ with a probability of the output being the correct solution of $\frac{1}{2}$ where n is the number of items in the database and m is the number of items being searched for.

The query complexity of Grover's search was proven to be optimal for any quantum turing machine by Charles Bennett et al. in [8].

The quantum oracle U which is an important component of Grover's search is generally described as a black box where the action the action of U on a basis state $|x\rangle$ is defined by:

$$U|x\rangle = (-1)^{f(x)}|x\rangle \tag{13}$$

This action translates to a phase shift of the the basis state $|x\rangle$ if the item at index x of the database is the item being searched for.

The Grover's operator, G is usually used to denote the quantum sub-routine of Amplitude Amplification. The steps of the operator are given by :

1. Apply the oracle operator, U
2. Apply the Hadamard transform, $H^{\bigotimes N}$
3. Apply the conditional phase shift to the every computational basis except $|0\rangle$: $|x\rangle = -(-1)^{\delta_{x0}}|x\rangle$
4. Apply the Hadamard transform, $H^{\bigotimes N}$

In operator notation, the Grover's Operator is defined as :

$$G = (2|\psi\rangle|\psi\rangle - I)U \tag{14}$$

where, $|\psi\rangle$ is the superposition of the computational basis states which denote the indices of the elements in the unstructured database.

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle, n = \lceil log_2 N \rceil \tag{15}$$

for an unstructured database of size N.

The Grover's algorithm is then described as following:

Algorithm 1 Grover's Algorithm

```
function GROVER'S ALGORITHM(Quantum Oracle U)
    Initialize $|\psi\rangle = |O\rangle^{\bigotimes N}$
    Apply the Hadamard Transform, $H^{\bigotimes N}$ to $|\psi\rangle$
    for $O(\sqrt{\frac{n}{m}})$ do
        Apply the Grover's Operator, $G = (2|\psi\rangle|\psi\rangle - I)U$ to $|\psi\rangle$
    Measure $|\psi\rangle$ in the superposition of the computational basis $H^{\bigotimes N}|0\rangle^{\bigotimes N}$
    return The result of the measurements with the highest m probabilities
```

3.1.2 Amplitude Amplification

The technique of amplitude amplification is at the core of Quantum search. In simplified terms, consider the Grover's Operator G acting on the superposition state, $|\psi\rangle$, the oracle acting on the state $|x\rangle$ where, x denotes the index of the item we are searching for in the unsorted array, applies a phase shift to the amplitude of this state. Applying the hadamard operator then decreases the average amplitude of the state due to the negative amplitude of the state $|x\rangle$. The conditional phase shift operator then inverts this amplitude with respect to the lowered mean amplitude and therefore, the amplitude of the basis state $|x\rangle$ is boosted. Applying this operation $O(\sqrt{\frac{n}{m}})$ times allow for the amplitude to be boosted enough to get the correct result from the measurement with probability $\frac{1}{2}$.

This technique makes the Grover's search algorithm generalizable to many applications other than searching for an item in as unsorted database such as Quantum counting algorithms where the goal is to determine the number of solutions to a search problem.

3.1.3 Quantum Minimum Finding Algorithm

The technique introduced by Grover's search algorithm in [4] described in the sections above has also been used by Christoph Durr and Peter Hoyer in [9] in order to devise the Quantum Minimum Searching Algorithm. This algorithm has been widely used in various optimization algorithms since in many optimization problems, a cost or an weight function is often being minimized or maximized.

4 Problem background

4.1 Shortest Path Optimization Problem

Shortest path problem is one of many graph optimization problems. The single-pair shortest path problem can be described as the following :

Given a graph $G = (V, E, C)$, a source node s and a destination vertex t, where :

1. $V = \{0, 1, ..., n\}$, a set of n vertices.
2. $E = \{< i, j >\}$, $i, j \in U$ and $U \subseteq V$, a set of edges between the vertices.
3. $C = (c_{ij})$ is used where c_{ij} is the weight of the directed edge from vertex i to vertex j and $c_{ij} = \infty$ is often used if there does not exist a directed edge from vertex i to vertex j

The goal is to find the path from the source vertex, s to destination vertex t which has the minimum accumulated cost of all possible paths from s to t.

The problem also has the following generalizations for which other efficient algorithms have been devised :

1. Single-source shortest path problem : Given a graph $G = (V, E, C)$ and a source vertex, t, find the shortest paths to all other vertices in V from t.
2. Single-destination shortest path problem : Given a graph $G = (V, E, C)$ and a destination vertex t, find the shortest paths from all other vertices in V to the destination vertex, t.
3. All-pairs shortest path problem : Given a graph $G = (V, E, C)$, find the shortest paths for every pair of vertex (s, t), where $s, t \in V$

5 Analysis of Existing Works

In the following section, we provide an analysis of the classical and quantum algorithms which have been used in order to solve the single-pair shortest path problem.

5.1 Classical Algorithms

The original classical algorithm for finding the shortest path between vertices in a graph with non-negative weights was devised by E.W. Dijkstra and presented in 1959 [10]. It is a greedy algorithm and the original algorithm which did not use a min-priority queue or a min-heap has a run-time of $O(n^2)$.

A classical optimization to Dijkstra's algorithm was introduced by Fredman and Tarjan in 1984. [11] The optimization using a min-priority queue implemented by a Fibonacci heap had a run-time complexity of $O(|E|+|V|log|V|)$ for a graph with $|E|$ edges and $|V|$ vertices.

5.1.1 Dijkstra's Original Shortest Path Algorithm

The original single pair shortest path algorithm for a directed graph with non-negative weights operates by maintaining two sets - a set of visited vertices and a set of unvisited vertices. The steps of Dijkstra's algorithm are as following:

1. The set of visited vertices is initialized to contain source vertex, s and the set of unvisited vertices is initialized to $V - s$, where V is the set of all vertices.

2. Initially the cost of the source vertex s is set to 0 and the cost of all the vertices in the unvisited set is set to ∞.

3. The current vertex is initialized to the source vertex.

4. The steps below are repeated until the unvisited set is empty i.e. all nodes have been visited or the destination vertex, t has been visited.

   (a) The current vertex is added to the visited set.

   (b) We consider all all adjacent vertices of the current vertex. A distance of the an adjacent vertices of the current vertex is updated if distance to the current vertex plus the distance from the current vertex to the adjacent vertex is lower than the current tentative distance of the adjacent vertex.

   (c) The current vertex is then set to the vertex from the unvisited set with the lowest tentative distance.

5.2 Quantum Algorithms

In this section, we discuss the quantum algorithms which are based on the concepts and principles described in the theoretical and quantum algorithm sections above to solve the shortest path problem. We explored these algorithms to further understand how can quantum computing principles be applied

Primarily, two major approaches have been used in order to find efficient quantum algorithms or modify existing classical algorithms for many optimization problems. The first approach is to simply replace the classical search component of the algorithms by the quadratically faster quantum search through the use of the techniques of Grover's search algorithm [4]. The second approach which is also widely used for optimization problems is the use of adiabatic quantum computing which is a subclass of a general class of meta-heuristic algorithms known as Quantum Annealing. [12]

The earliest work on developing an efficient quantum algorithm for the shortest path problem was considered by Mark Heiligman in 2003 [13]. Using the techniques from Grover's search, a variant of Dijkstra's algorithm i.e. Dijkstra's algorithm with periodic updating is used. Heiligman initially described how the quantum algorithm for finding the minimum of an unordered set [9] can be used to reduce the iteration cost at step 4(c) (refer to Section 5.1.1) to $O(\sqrt{|V| - |VS|}$ where $|VS|$ is the cardinality of the visited set. However, the entire cost of the algorithm still remains $O(n^2)$ due to the cost at step 4(b) which is still of the order $O(|V| - |VS|)$. This highlights one of the most common problems when quantum optimizations for classical algorithms are considered. Simply replacing the classical components with their quantum counterparts does not necessarily lead to an improvement in the algorithm.

The improvement to the time-complexity of the algorithm to $O(n^{1.75})$ is achieved in [13] through a trade-off between the search and the update sections of Dijsktra's algorithm since the search section is where the quantum algorithm for finding the minimum [9] can be used for run-time improvement. An iteration step k is used in the loop of Step 4 in Section 5.1.1 i.e. the tentative distance of the vertices in the unvisited set is updated after every k steps.

A similar approach to the single source shortest path problem is described in [14] which offers the bounded error query complexity of single source shortest path in the adjacency array and the matrix model. The quantum algorithm for finding the minimum [9] is again used to improve the query complexity over the classical algorithm. Through this modification, the upper-bounds on the quantum complexity of the single source shortest path problem is determined to be $O(n^{\frac{3}{2}}log^2n)$ for the matrix model and $O(\sqrt{nm}log^2n)$ for the adjacency model.

There also exists a different model of Quantum computation using adiabatic quantum computing. Although not universal and with applications to only optimization problems, it is still a widely used model for Quantum algorithms. For example, the method discussed in [15] suggest the use of an Ising model for the shortest path problem.

5.3 Extensions

In addition to the quantum algorithms for the single-pair shortest path discussed in Section 5.2, an efficient non-trivial quantum algorithm for the All-pairs shortest path problem has been presented in [16] which has a time complexity of $O(n^{2.5-\epsilon})$ for some constant $\epsilon > 0$. Their work follows the approach that is common among the algorithms discussed in Section 5.2 in which the key components of classical algorithms are replaced by their faster and more efficient quantum counterpart. In the case of [9], this is achieved by replacing the brute classical matrix product calculation with time complexity of $O(n^{3-\epsilon})$ with its more efficient quantum counterpart using Grover's search with a complexity of $O(n^{2.5-\epsilon})$ and replacing the classical Dijkstras algorithm with a query-complexity of $O(n^2)$ with the quantum variant discussed in Section 5.2 from [14] with the query-complexity of $O(n^{\frac{3}{2}}log^2n)$.

6 Synthesis

6.1 Using quantum algorithms to improve classical algorithms

In this section, we present a discussion of the common techniques we observed and propose how general optimal classical algorithms can be potentially optimized using concepts from quantum computing.

Throughout our study of how techniques from Quantum Computing have been used to improve classical algorithms, we found a general trend that has been used in optimizing many classical algorithms.

The common trend we observed among many approaches was that in order to optimize a classical optimal classical algorithm using concepts from quantum computing, we first have to determine the separate components of the algorithm such as the searching, the operations on data, the transformations etc.

Once the separate components are realized, the next step is to try to replace the classical components by their quantum counterparts, for ex. the search components can be easily replaced by techniques from Grover's search algorithm and the generalized quantum search, similarly in algorithms involving periodic functions, the classical fast fourier transform can be replaced by the quantum fourier transform in order to improve the complexity of the algorithm.

The classical algorithm may need to be modified with some trade-off between the classical components and the quantum components in order to actually improve the complexity of the algorithm.

However, the discussion above only applies to improving classical algorithms which is different than solving classical problems using quantum computing. The ingenuity of quantum algorithms does not lie in modifying existing quantum algorithms but to use the concepts we have discussed in Sections 2. and 3. to solve the problems using the beneficial quantum mechanical characteristics of quantum computers.

6.2 A discussion on an approach to solving the shortest path problem

In addition to the application of quantum algorithms to the shortest path problem, we also explored the problem of encoding information about a graph or a similar connected data structure. We tried an approach similar to [17] in which the distance between the vertices in encoded in the phases of an unitary matrix. This allow for the use of the phase estimation algorithm using Quantum Fourier Transform to estimate the eigenstates corresponding to routes in a graph which can be used in conjunction with the Quantum algorithm for finding the minimum to get the best route.

Since, the algorithms we studied for the shortest path problem were hybrid classical-quantum algorithms i.e. they used classical components for the graph encoding, updating etc. and the quantum component was only used for searching the minimum. We tried the approach of using the distances of the graph encoded, however we realized :

1. Firstly even if the query component i.e. searching for the minimum has a query complexity of $O(\sqrt{n})$, currently the quantum fourier transform has the complexity of $O(n^2)$ which is equivalent to the original time complexity of Dijkstra's algorithm.

2. Secondly, we determined that the Dijkstra's algorithm is an inherently sequential algorithm, therefore in order to fully exploit the massive parallelism that is available

from quantum computers and quantum mechanical phenomena such as superposition, entanglement etc., we need to devise a completely new approach to solving the shortest path problem which was beyond the scope of this project.

This also suggests that in order for classical algorithms to directly exploit the property of quantum parallelism, the functions and different components of the algorithms have to be easily parallelizable.

7 Discussion

7.1 Major Learnings

Over the course of the project, we learned in detail about the fundamental concepts in Quantum computing and how these concepts allow for the quantum model of computation to be a more generalized model of computation.

We studied and learned about the inner components of two of the primary classes of quantum algorithms i.e. quantum search algorithms and quantum fourier transform. We determined how generalizable techniques from Grover's search algorithm can be applied to various optimization problems to improve the efficiency of classical algorithms.

We learned about how quantum computing algorithm can be used to optimize classical algorithms and some of the requirements of solving graph problems on quantum machines.

7.2 Challenges

One of the major challenges we faced in the field of Quantum algorithms is that there are very few general techniques which have been devised and are known to be applicable to solving problems. Most of the current quantum algorithms which offer advantages over classical algorithms are known to be based on these few techniques. This means in order to understand how quantum algorithms can be applied to solve these computational problems efficiently, it is important to understand the underlying concepts and principles of these techniques which requires a detailed study of the basics and fundamentals of Quantum computing.

In addition, practical quantum algorithms to solve real-world problems are still yet to be fully discovered due to the constraints on current quantum computer implementations in terms of circuit depth and qubit coherence. Most algorithm research so far has been primarily theoretical due to this constraint.

7.3 Future directions

This project served as a comprehensive introduction to the theory of Quantum computation and information and how techniques from the field can be applied to solve computation problems. However, the problem of finding new techniques which exploit the massive parallelism and natural quantum phenomena to improve the existing and devise new algorithms is still a very active field of research.

The graph algorithms which were explored in the project were hybrid classical-quantum algorithms. The open problem we would like to address in the near-future is whether information about a graph or other graph-like structures such as networks can be encoded in qubits using the circuit model of quantum computation.

8 Conclusion

In this project, we studied in detail the theoretical background of the field of Quantum computation and information. We learned how principle concepts from the two of three general classes of quantum algorithms can be used to devise efficient quantum algorithms for solving problems. We also considered the graph optimization problem of single source shortest path and discussed how concepts from quantum computing have been used in order to develop more efficient quantum algorithms to solve this problem. We presented a general outline on how quantum algorithms can be used to improve classical algorithms and discussed an approach to solving the shortest path problem on a quantum computer.

References

[1] P. Benioff, "The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines," *Journal of Statistical Physics*, vol. 22, pp. 563–591, May 1980.

[2] R. P. Feynman, "Simulating physics with computers," *Internantional Journal of Theoretical Physics*, vol. 21, pp. 467–488, 1982. [,923(1981)].

[3] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Review*, vol. 41, no. 2, pp. 303–332, 1999.

[4] L. 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, (New York, NY, USA), pp. 212–219, ACM, 1996.

[5] P. J. Coles, S. Eidenbenz, S. Pakin, A. Adedoyin, J. Ambrosiano, P. Anisimov, W. Casper, G. Chennupati, C. Coffrin, H. Djidjev, D. Gunter, S. Karra, N. Lemons, S. Lin, A. Lokhov, A. Malyzhenkov, D. Mascarenas, S. Mniszewski, B. Nadiga, and W. Zhu, "Quantum algorithm implementations for beginners," 04 2018.

[6] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[7] D. Deutsch and R. Jozsa, "Rapid solution of problems by quantum computation," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 439, pp. 553 – 558, 1992.

[8] C. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, "Strengths and weaknesses of quantum computing," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1510–1523, 1997.

[9] C. Durr and P. Hoyer, "A quantum algorithm for finding the minimum," *CoRR*, vol. quant-ph/9607014, 07 1996.

[10] E. W. Dijkstra, "A note on two problems in connexion with graphs," *NUMERISCHE MATHEMATIK*, vol. 1, no. 1, pp. 269–271, 1959.

[11] M. L. Fredman and R. E. Tarjan, "Fibonacci heaps and their uses in improved network optimization algorithms," in *25th Annual Symposium onFoundations of Computer Science, 1984.*, pp. 338–346, Oct 1984.

[12] D. de Falco and D. Tamascelli, "An introduction to quantum annealing," *RAIRO - Theoretical Informatics and Applications*, vol. 45, 07 2011.

[13] M. Heiligman, "Quantum algorithms for lowest weight paths and spanning trees in complete graphs," 2003.

[14] D. Christoph, H. Mark, H. Peter, and M. Mehdi, "Quantum query complexity of some graph problems," *SIAM J. Comput.*, vol. 35, pp. 1310–1328, June 2006.

[15] J. S. C. O. E. B. Christian Bauckhage, Rafet Sifa and K. Cvejoski, "Towards shortest paths via adiabatic quantum computing," in *Proceedings of the 14th International Workshop on Mining and Learning with Graphs (MLG)*, 2018.

[16] A. Nayebi and V. V. Williams, "Quantum algorithms for shortest paths problems in structured instances," *CoRR*, vol. abs/1410.6220, 2014.

[17] K. Srinivasan, S. Satyajit, B. Behera, and P. Panigrahi, "Efficient quantum algorithm for solving travelling salesman problem: An ibm quantum experience," 05 2018.