CS 698 Quantum Computing - Final Project Report
New Jersey Institute of Technology - Fall 2020
Professor – Moshiur Rahman

Sai Akhilesh Chunduri (sc2344)
Jasminkumar Patel (jp878)

Project – Quantum Computing Algorithms

Quantum Computing

Quantum Computing is the use of quantum phenomena such as superposition and entanglement to perform computation. Computers that perform quantum computations are known as quantum computers.

Introduction

In this project, we have taken quantum algorithms as a topic and with the help of course textbook and other reference materials that are available online, we were able to study and present the details of fundamental quantum algorithms. And we have also implemented the

Quantum Algorithms

Quantum computers utilize the unique characteristics of the quantum systems to process exponentially large amount of data in a very short time. Obviously, this kind of computing power has the potential application in mathematics, cryptography, and in the simulation of quantum systems themselves. A fundamental feature of many of these algorithms is that they allow a quantum computer to evaluate a function f (x) for many different values of x simultaneously.

There are 4 fundamental Algorithms in Quantum Computing

1. Deutsch’s Algorithm
2. Deutsch – Jozsa Algorithm
3. Grover’s Search Algorithm
4. Shor’s Factoring Algorithm

- In this project we have done an in-depth mathematical analysis of two of the above-mentioned algorithms namely Deutsch – Jozsa Algorithm, Grover’s Search Algorithm, and implemented them using Qiskit.

- With this report we have described basic functions of the algorithms alongside some output images that were developed from implementation of algorithms using Qiskit for an overview of the algorithms and a quick comparison between classical computation and quantum computation.

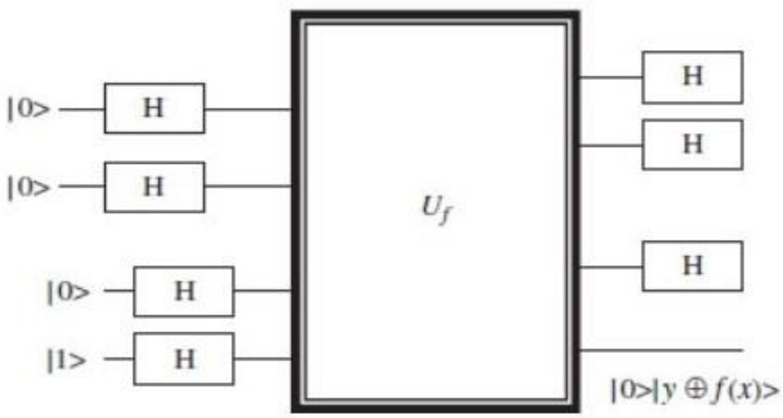
- There will be a detail documentation submitted with the report for better understanding of Quantum computation and Quantum Algorithms

Deutsch – Jozsa Algorithm

The Deutsch–Jozsa algorithm solves a black-box problem which probably requires exponentially many queries to the black box for any deterministic classical computer, but can be done with exactly one query by a quantum computer. If we allow both bounded-error quantum and classical algorithms, then there is no speedup since a classical probabilistic algorithm can solve the problem with a constant number of queries with small probability of error. The algorithm determines whether a function *f* is either constant (0 on all inputs or 1 on all inputs) or balanced (returns 1 for half of the input domain and 0 for the other half).

Deutsch's algorithm works on a single input bit in the simple case where f:(0, 1) → (0, 1). A generalization of the algorithm known as Deutsch-Jozsa algorithm can act on an n-bit function f:(0, 1) → (0, 1). Assuming a 2-bit function of the form (0, 1) → (0, 1) is provided and it is known at the outset that the function is one of the four shown below,

Input	Output	Input	Output	Input	Output	Input	Output
0 0	1	0 0	0	0 0	0	0 0	0
0 1	0	0 1	1	0 1	0	0 1	0
1 0	0	1 0	0	1 0	1	1 0	0
1 1	0	1 1	0	1 1	0	1 1	1



$$
\begin{aligned}
|\psi_0> &= |0>\cdots\cdots|0>|1> \\
&= |0>^{\otimes n}|1>
\end{aligned}
$$

Quantum Circuit for Deutsch – Jozsa Algorithm

Deutsch – Jozsa Algorithm — Implementation using Qiskit

1. Prepare two quantum registers initialized to zero. The first is an n-qubit register for querying the oracle, and the second is a one-qubit register for storing the answer of the oracle |0 . . . 0i|0i

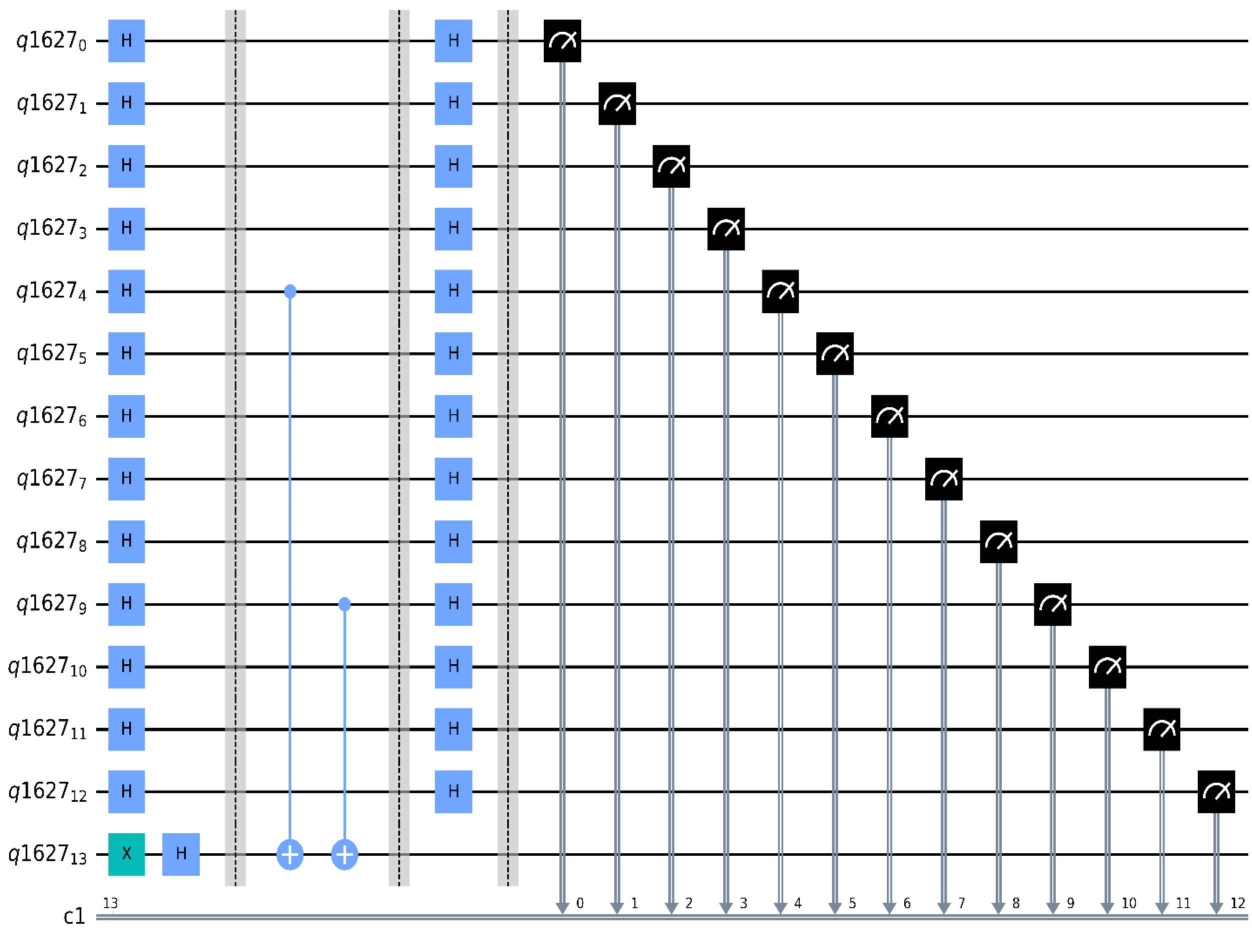
2. Create the superposition of all input queries in the first register by applying the Hadamard gate to each qubit. H ⊗n |0 . . . 0i|0i = 1 √ 2 n 2 n−1 ∑ i=0 |ii|0i

3. Flip the second register and apply the Hadamard gate. This is to store the answer of the oracle in the phase. 1 √ 2 n 2 n−1 ∑ i=0 |ii|0i → 1 √ 2 n+1 2 n−1 ∑ i=0 |ii(|0i − |1i)

4. Query the oracle 1 √ 2 n+1 2 n−1 ∑ i=0 |ii(|0i − |1i) → 1 √ 2 n+1 2 n−1 ∑ i=0 (−1) f(i) |ii(|0i − |1i)

5. Apply the Hadamard gate to the first register

6. Measure the first register. If it is non-zero, then conclude that the hidden Boolean function is balanced. Otherwise, conclude that it is constant.



Deutsch – Jozsa Algorithm — Implementation using Qiskit and Output

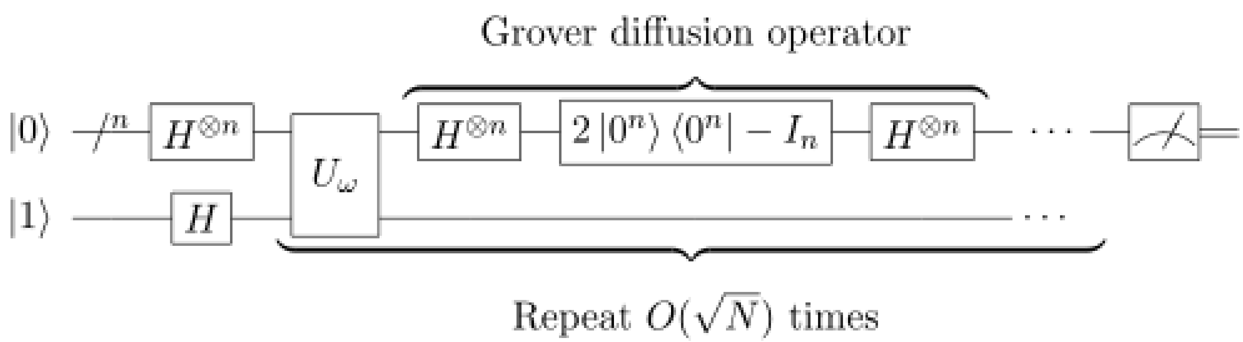
Grover's Search Algorithm

Grover's Algorithm searches an unstructured database with N entries for a marked entry using only O(Sq. rt N) queries instead of the O(N) queries required classically. Theorists have considered a hypothetical generalization of a standard quantum computer that could access the histories of the hidden variables in Bohmian mechanics.

Grover's algorithm performs a search over an unstructured and unsorted database of N entries for accessing a particular entry. Using a classical computation model a solution can be obtained by checking every item in the database to find the desired one.

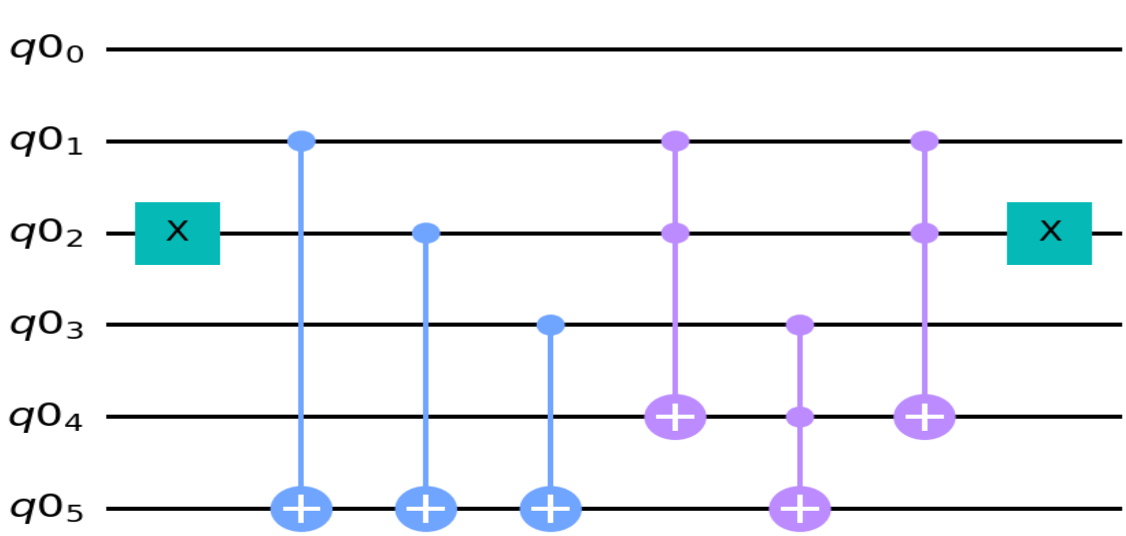
This search in the worst case will require a time-complexity O(N).

Grover developed an algorithm that finds a marked item, that is, the item of interest x* in a set of N elements (x , x , . . . , x); it completes the search in time complexity O(N) by utilizing the nature of quantum systems.



Quantum Circuit representation of Grover's Algorithm

Grover's Search Algorithm — Implementation using Qiskit



Observations and Conclusions :

• Brief Introduction of Quantum Algorithms has been proposed.

• Thorough mathematical analysis is conducted on Grover's Search Algorithm.

• Pseudocode of Grover's Search Algorithm has been written, in the process got introduced to the concept of Inversion about the average.

• Grover's Search Algorithm has been implemented using Qiskit library and results have been generated.
• Deutsch – Jozsa Algorithm is thoroughly studied and mathematical analysis of the same has been presented.

• Time Complexity analysis of Deutsch – Jozsa Algorithm & Grover's Search Algorithm has been done as well.

• Deutsch – Jozsa Algorithm is implemented using Qiskit library and the results are generated. Same implementation has been scaled to a real-world circuit.

• Learnt to read a research paper, understand it and implement the concepts proposed in the papers.

• Learnt to create job, schedule the job to run on a Live IBM Quantum Machine.

• Learnt about the concept of API calling ~~ when a unique token is generated in our personal IBM account and we must call the API using the token in our local environment for Running the circuit on real devices.

References :

Research papers referred :

• https://arxiv.org/abs/1708.03684

• https://arxiv.org/pdf/quant-ph/9708016.pdf

Textbooks referred :

• https://www.accessengineeringlibrary.com/content/book/9781260123111

Brief overview of the Contributions of the Project

Name of Team Member	Contribution %	Tasks Done
Sai Akhilesh Chunduri (sc2344)	50.0 %	• Conducted mathematical analysis of Grover's Search Algorithm. • Studied Research paper related to Grover's Search algorithm. • Implemented Grover's Search Algorithm and Deutsch – Jozsa Algorithm. • Scaled the same implementation onto a real world circuit.
Jasmin Patel (jp878)	50.0 %	• Conducted mathematical analysis of Deutsch – Jozsa Algorithm. • Studied research paper related to Deutsch – Jozsa Algorithm. • Implemented Grover's Search Algorithm and Deutsch -- Jozsa Algorithm. • Scaled the same implementation onto a real world circuit.

-- Thank You --
