

Grover's Algorithm: Oracle Development

THESIS

**Submitted in Partial Fulfillment of
the Requirements for
the Degree of**

MASTER OF SCIENCE (Applied Physics)

at the

**NEW YORK UNIVERSITY
TANDON SCHOOL OF ENGINEERING**

by

Amro Saidelahel

May 2023

Grover's Algorithm: Oracle Development

THESIS

Submitted in Partial Fulfillment of
the Requirements for
the Degree of

MASTER OF SCIENCE (Applied Physics)

at the

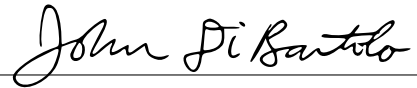
NEW YORK UNIVERSITY
TANDON SCHOOL OF ENGINEERING

by

Amro Saidelahel

May 2023

Approved:



Department Chair Signature

5/6/2023

Date

Approved by the Guidance Committee:

Major: Applied Physics



Javad Shabani

Center for Quantum Information Physics Director; Associate Professor
New York University - Graduate School of Arts & Science

05/08/2023

Date



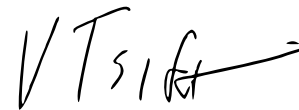
Dries Sels

Assistant Professor

New York University - Graduate School of Arts & Science

05/08/2023

Date



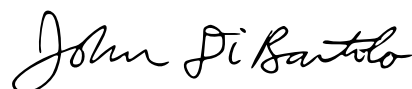
Vladimir Tsifrinovich

Industry Professor

New York University - Tandon School of Engineering

5/6/2023

Date



John Di Bartolo

Department Chair; Industry Professor

New York University - Tandon School of Engineering

5/6/2023

Date

Microfilm or other copies of this thesis are obtainable from

UMI Dissertation Publishing

ProQuest CSA

789 E. Eisenhower Parkway

P.O. Box 1346

Ann Arbor, MI 48106-1346

Vita

Amro Elsayed Ahmed Imam Saidelahel was born on June 24th, 1997 in the Sharqia Governorate, Egypt. His schooling was at Al Najah Private School in the United Arab Emirates, studying for IGCSEs and the A-Level Diploma until graduation in May 2015. He received his Bachelor of Science in physics with honors from the United Arab Emirates University, where he co-founded The Physics Club and The Debate Club, graduating in 2020. He is also a recipient of The Sheikh Mohamed bin Zayed Scholars Program Scholarship at New York University Abu Dhabi (NYUAD), graduating in May 2020 with honors. He then worked as a Research Assistant in Extragalactic Astronomy at NYUAD before matriculating at New York University (NYU) in September 2021 for his Master of Science in Applied Physics at The Tandon School of Engineering. He conducted his research on quantum algorithms at The Center for Quantum Information Physics starting March 2022. He founded The Egyptian Union at NYU before graduating in May 2023.

Acknowledgements

All thanks first and foremost goes to Allah SWT. I extend endless gratitude to my parents, Amani and Elsayed, that have never failed to choose to invest in me emotionally, physically and financially. It is only because of them that I have arrived at this stage, and only because of them, that I shall go elsewhere. I thank my brothers Ahmed and Mohammed for supporting me throughout this journey with their words, presence and the remarkable model they provided me with to emulate. More thanks go to my dear friends, Basil, Redha, Ibrahim, Noor, Awadhi, Omar, Ayman and Ali that have equally pushed and supported me through hurdles leading up to this stage. Moreover, I thank the Sheikh Mohamed bin Zayed Scholars Program for providing me with a full scholarship to pursue my Master's degree at NYU. I would not be graduating without their generous support. Finally, I thank Professor Javad Shabani, my advisor and director of the Center for Quantum Information Physics, for providing me with the opportunity to freely explore and innovate in the field of quantum algorithms as well as being a bridge to remarkable learning opportunities both within and outside the lab. I also extend my gratitude to Dr. Mohammad Farzaneh for endless discussions that helped me further comprehend and develop this work.

Amro Saidelahel

May 2023

To Allah SWT, my family, friends and Egypt.

ABSTRACT**Grover's Algorithm: Oracle Development****by****Amro Saidelabel****Advisor: Prof. Javad Shabani, Ph.D.****Submitted in Partial Fulfillment of the Requirements for
the Degree of Master of Science (Applied Physics)****May 2023**

Quantum supremacy is achieved when a quantum computing process outperforms any known classical computing counterpart. One of the very promising quantum algorithms to exhibit this supremacy, is the quantum search algorithm for unsorted databases, Grover's algorithm. In this report we present the fundamental background for the inner-workings of the algorithm mathematically and graphically. We also replicate a hypothesized application for the algorithm, gravitational wave matched filtering using IBM's Qiskit. Furthermore, we propose a model for an n-qubit global oracle that can be used to retrieve any desired computational basis state using a multi-controlled single target Z-gate $C^{n-1}Z$. We also explore entangled registers and formulate an oracle operator for database search problems.

Table of Contents

Vita	iv
Acknowledgements	v
Abstract	vii
List of Figures	xiii
1 Introduction	1
2 Background	4
2.1 The Quantum Advantage	4
2.2 Mathematical & Graphical Representation	6
2.3 The Oracle	17
3 Gravitational wave matched-filtering	23
4 The n-qubit Oracle	30
4.1 An n-qubit boolean oracle	30
4.2 A viable oracle	32
4.3 A dynamic oracle	33
5 Entangled Registers and qRAM	38

	ix
6 Conclusions and Future Work	42
Appendix A	44

List of Figures

2.1	The Full Grover Process	9
2.2	Boolean Oracle [12]	11
2.3	Phase Oracle [12]	14
2.4	An example of information being encoded into computational basis states. The first column is of a random sample of numerical data, the second is of the corresponding binary representation. The third column is of the corresponding computational basis states and the fourth is the decimal representation of the basis.	18
2.5	Application of Haddamard gates to create a uniform superposition of computational basis states in a 3-qubit circuit.	18
2.6	Evolution of the probability distribution of measured states across different Grover iterations for a 6-qubit system with $M = 4$. The possible basis states are on the x-axis and their corresponding probabilities on the y-axis. The solution states in this example are $ 001001\rangle$, $ 101001\rangle$, $ 011001\rangle$, $ 111001\rangle$	21

- 3.1 Left: Noisy signals with different SNR values. Right: Corresponding output signals from the matched filter. It can be seen that the output signals, are of a higher SNR compared to the input signals. They also appear at the point in time, where the corresponding embedded waveform in the input signals start [4]. 24
- 3.2 Recreation of the Quantum Oracle Circuit from [13]. The circuit is split into two sections, separated by the dashed barrier in the middle. The first section is the phase kickback operation (Oracle). Qubits q_i encode the signal, and qubits t_i store a superposition of all possible templates. The ancillary qubit (anc) is used to store phase kickback feedback. The measurement classical bit (meas) is used to store measurements of t_i 's performed after the diffusion operator. . . 28
- 4.1 An extension of the boolean oracle proposed in [12], to a 5-qubit search for the state $|00101\rangle$. The ancilla qubit (q_5) is initialized to $|1\rangle$, and then the entire qubit register is put into a maximally entangled superposition using Hadamard gates. The oracle embedded between the two barriers, consists of X gates placed before and after the $C^5X_{12345,6}$. The diffuser after the second barrier is identical to what has been previously described in 2.2.3 but with a $C^4Z_{1234,5}$ instead. 31
- 4.2 The probability distribution of measured states for the circuit in Figure 4.1 after 10,000 runs. The state $|00101\rangle$ was most likely to be observed with a probability $\approx 33.7\%$ 31

4.3	An extension of the phase oracle proposed in [12], to a 5-qubit search for the state $ 00101\rangle$. The entire qubit register is put into a maximally entangled superposition using Hadamard gates. The oracle embedded between the two barriers, consists of X gates placed before and after the $C^4Z_{1234,5}$. The diffuser is identical to the aforementioned boolean oracle in 4.1.	32
4.4	The probability distribution of measured states for the circuit in Figure 4.3 after 10,000 runs. The state $ 00101\rangle$ was most likely to be observed with a probability $\approx 99.9\%$	33
4.5	The Eye of Horus primitive for a 3-qubit system Grover search for state $ 100\rangle$ and the corresponding probability distribution of states after circuit simulation.	36
4.6	Probability distribution after running a Grover circuit using the n-qubit generalizable Grover's algorithm for 3-qubits on a noisy quantum processor. JobId: 640964c55fbdedb851894a2d	36
4.7	The increase in circuit depth on a logarithmic scale for the n-qubit generalizable Grover's algorithm with an increasing number of qubits in the circuit. An exponential plot is plotted for reference.	37
A.1	Quantum Counting circuit for estimating the number of solutions M in a Grover search. t are the counting qubits, and n are the searching qubits. After initialization using Hadamard gates on both registers, a series of controlled Grover iterations are appended onto the circuit before the final inverse QFT is applied to the counting register.[3]. .	44

A.2 A catalogue of possible oracle circuits corresponding to single 3-qubit states. The purple coded 3-qubit gate is a $C^2Z_{12,3}$ gate, and the blue 2-qubit gate is a CZ gate. 45

Chapter 1

Introduction

Quantum computing is a method of computation that borrows from various concepts and methods used in classical computation, while making use of non-classical quantum mechanical phenomena to perform specific tasks at a faster and more efficient rate [22]. State of the art quantum processors, however, are still nowhere near achieving their promised potential. This is due to both the environmental interference with the qubits making up the processor and the qubit-qubit interaction, leading to very short decoherence times [23]. The qubits are cooled to temperatures close to absolute zero for better control over the states they are required to be in [2], but it has proven difficult to prevent decoherence for more than $8.2 \mu s$ as of the writing of this report [8]. Another problem lies in the difficulty of error-correction in the quantum mechanical regime where no cloning of the information encoded in the quantum states can be performed [25]. That being said, the reason work is still being done to improve the accuracy and reliability of quantum computing, lies in its potential. Apart from the increase in computational power and efficiency, quantum computing holds potential to solve

classically intractable problems. Some of the applications of quantum computing include the factorization of prime numbers [20], the simulation of quantum systems [11] [14], data encryption [5] and the search through an unsorted database [16]. The last of the list is attended to by the quantum search algorithm; Grover's algorithm. In a database containing N entries, a classical search algorithm would require an average of $\frac{N}{2}$ steps before finding the desired answer, meaning that such algorithms scale as $\mathcal{O}(N)$. Grover's algorithm, however, only scales as $\mathcal{O}(\sqrt{N})$ [17], a quadratic speed-up compared to the classical algorithm. Although this is not as impressive as the exponential speed-up promised by Shor's algorithm, it still is an advantage of utility. The algorithm comprises mainly of two components:

1. The search oracle
2. The diffuser/amplifier

A lot of the literature has focused on potential applications for Grover's algorithm. Apart from the unsorted database search, this includes the solving of the collision problem [7], gravitational wave detection [13] and combinatorial optimization problems [21]. There has, however been a lack of focus on the search oracle component of the algorithm. It is convention to refer to the oracle in the algorithm, as a black box, a function $f(x)$ that marks entries of the database, encoded as computational basis states, when x is a desired/solution state. This marking normally takes the form of a phase shift applied to the solution state. From there the second part of the algorithm is put to work, the amplifier.

The amplifier receives the superposition that now includes the marked state(s) and increases its probability amplitude, thereby increasing the state's probability of

being observed upon measurement whilst diminishing and suppressing the probability of the non-marked states. The process of marking and then amplifying, known as a Grover iteration, is then repeated $\approx \frac{\pi}{4} \sqrt{\frac{N}{M}}$ times [17], where M is the number of solutions in our search problem i.e. the number of states we wish to mark in the superposition. However, many problems within the scope of Grover's algorithm have more than one solution, and the number of solutions M is often unknown. For this purpose we introduce and discuss in further detail the quantum counting algorithm which makes use of fundamental quantum algorithms such as the quantum Fourier transform (QFT) and phase estimation [6]. Assuming that the correct number of Grover iterations has been performed, the resulting probability distribution of the states is now highly skewed to our solution state(s). Therefore, upon measurement, and consequently, the collapse of the superposition, the desired solution(s) are likely to be observed, while all non-solutions are suppressed.

Chapter 2

Background

2.1 The Quantum Advantage

Imagine that you are a cryptocurrency coin miner. Your task is to find a specific 256-bit long string, which we call x , to mine a coin successfully. Since you have 256 bits in the string, and each bit can take up values of 0 or 1, your search space includes $N = 2^{256} \approx 10^{77}$ possibilities. Depending on its position, you can find the string x at the beginning of the database storing all possible strings, or closer to the end. If this brute search method is repeated enough times, you will on average require $\frac{2^{256}}{2} = 2^{255}$ search instances i.e. $\mathcal{O}(N)$. Grover's algorithm, provides better prospects. For a given, unsorted database of size N , the algorithm's time complexity is $\mathcal{O}(\sqrt{N})$. This is due to the number of search instances, or equivalently, Grover iterations that the algorithm requires for successful execution. This number is $\approx \frac{\pi}{4} \sqrt{\frac{N}{M}}$, where M is the number of solutions for the search problem. This translates to a **quadratic speedup** in run-time relative to its classical counterpart. Another way of thinking of the difference between both methods, is the fact that

in practice, Grover's algorithm does not traverse the search space in search for the solution. Instead, it uses superposition, an oracle and a diffuser to retrieve the solution i.e. the difference in run-time complexity is not due to a search over fewer terms on Grover's part, but due to inherently different techniques used in the search. We outline the inner-workings of the oracle in section [2.3](#) for better distinction. We also reference the Quantum Counting circuit [A.1](#) that can be used to estimate M when such information is unknown [\[6\]](#).

2.2 Mathematical & Graphical Representation

In this section, we first graphically describe the entire Grover iteration for better intuition, and then we mathematically describe the behaviour of qubits, in two different oracle circuits, under the application of various gates. The first type, the boolean oracle, makes use of an ancilla qubit initialized to $|1\rangle$ to store the phase kickback operation, and a combination of a multi-control single-target $CCCX$ gate, Toffoli gates CCX , CX gates, and X gates. The second type, the phase oracle, negates the need for an ancilla qubit and uses a combination of a multi-control single-target CCZ gate, CZ gates, X gates, and Z gates.

2.2.1 The Grover Iteration

The Grover iteration is performed in 2 operations: 1) The oracle U_w which marks the desired state, and 2) the diffusion operator $U_s = 2|s\rangle\langle s| - I$ which amplifies the marked state relative to the non-solution states. To be able to mathematically observe the effect of the diffusion operator, we define a superposition of computational basis states as our initial state, $|s\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} x_i |i\rangle$. How this state is created is further discussed in [2.3](#).

For example, for a 2-qubit system

$$|s\rangle = \frac{1}{\sqrt{4}} (|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

Assume that your desired state, $|w\rangle = |10\rangle$. The effect the oracle operator U_w has on the initialized state $|s\rangle$ is

$$U_w |s\rangle = \frac{1}{\sqrt{4}} (|00\rangle + |01\rangle - |\mathbf{10}\rangle + |11\rangle) = |s^*\rangle$$

If we were to arbitrarily pile the non-solution states of this superposition into a separate superposition $|s_{\perp}\rangle = \frac{1}{\sqrt{3}}(|00\rangle + |01\rangle + |11\rangle)$, we can then re-write the **marked** superposition $|s^*\rangle$

$$|s^*\rangle = -\sqrt{\frac{1}{N}}|w\rangle + \sqrt{\frac{N-1}{N}}|s_{\perp}\rangle$$

Note that $\left\|-\sqrt{\frac{1}{N}}\right\|^2 + \left\|\sqrt{\frac{N-1}{N}}\right\|^2 = 1$.

The effect of the diffusion operator, U_s on $|s^*\rangle$ is as follows:

$$\begin{aligned} & U_s \left[-\sqrt{\frac{1}{N}}|w\rangle + \sqrt{\frac{N-1}{N}}|s_{\perp}\rangle \right] \\ &= (2|s\rangle\langle s| - I) \left[-\sqrt{\frac{1}{N}}|w\rangle + \sqrt{\frac{N-1}{N}}|s_{\perp}\rangle \right] \\ &= \underbrace{2|s\rangle\left(-\frac{1}{N} + \frac{N-1}{N}\right)}_{\text{Effect of } 2|s\rangle\langle s|} + \underbrace{\sqrt{\frac{1}{N}}|w\rangle - \sqrt{\frac{N-1}{N}}|s_{\perp}\rangle}_{\text{Effect of } -I} \\ &= \left(2 - \frac{4}{N}\right)|s\rangle + \sqrt{\frac{1}{N}}|w\rangle - \sqrt{\frac{N-1}{N}}|s_{\perp}\rangle \\ &= \left(2 - \frac{4}{N}\right)\left(\sqrt{\frac{1}{N}}|w\rangle + \sqrt{\frac{N-1}{N}}|s_{\perp}\rangle\right) + \sqrt{\frac{1}{N}}|w\rangle - \sqrt{\frac{N-1}{N}}|s_{\perp}\rangle \end{aligned}$$

$$|s_G\rangle = \underbrace{\left(3 - \frac{4}{N}\right)} \cdot \sqrt{\frac{1}{N}} |w\rangle + \underbrace{\left(1 - \frac{4}{N}\right)} \cdot \sqrt{\frac{N-1}{N}} |s_\perp\rangle$$

Ignoring the $\frac{4}{N}$ term which is $\ll 1$ for large N , we can see that the amplitude of the solution/desired state $|w\rangle$ has tripled in the final superposition $|s_G\rangle$ after a full Grover iteration. Once both these operations are repeated for the optimal number of Grover iterations, we end up with larger amplitudes for state $|w\rangle$, and hence a superposition that is much more likely to collapse to a solution state upon measurement.

We can also represent the aforementioned evolution of our initial state $|s\rangle$ graphically. Figure-2.1 describes the changes $|s\rangle$ undergoes when represented in a 2-D plane spanned by the solution superposition $|w\rangle$ and the non-solution superposition $|s_\perp\rangle$, that $|s\rangle$ is made up of. In Figure-2.1a, $|s\rangle$ can be seen as a vector between orthogonal vectors skewed more towards the non-solution vector $|s_\perp\rangle$. This is due to the fact that $|s\rangle$ is made up of more non-solution states than solution states. Figure-2.1b demonstrates the effect of the oracle U_w on $|s\rangle$, which takes form as a reflection about $|s_\perp\rangle$ yielding $|s^*\rangle$. Finally, in Figure-2.1c, the diffusion operator U_s acts on $|s^*\rangle$, which takes form as a reflection about $|s\rangle$, moving the system's state closer to $|w\rangle$ after a full Grover iteration yielding $|s_G\rangle$.

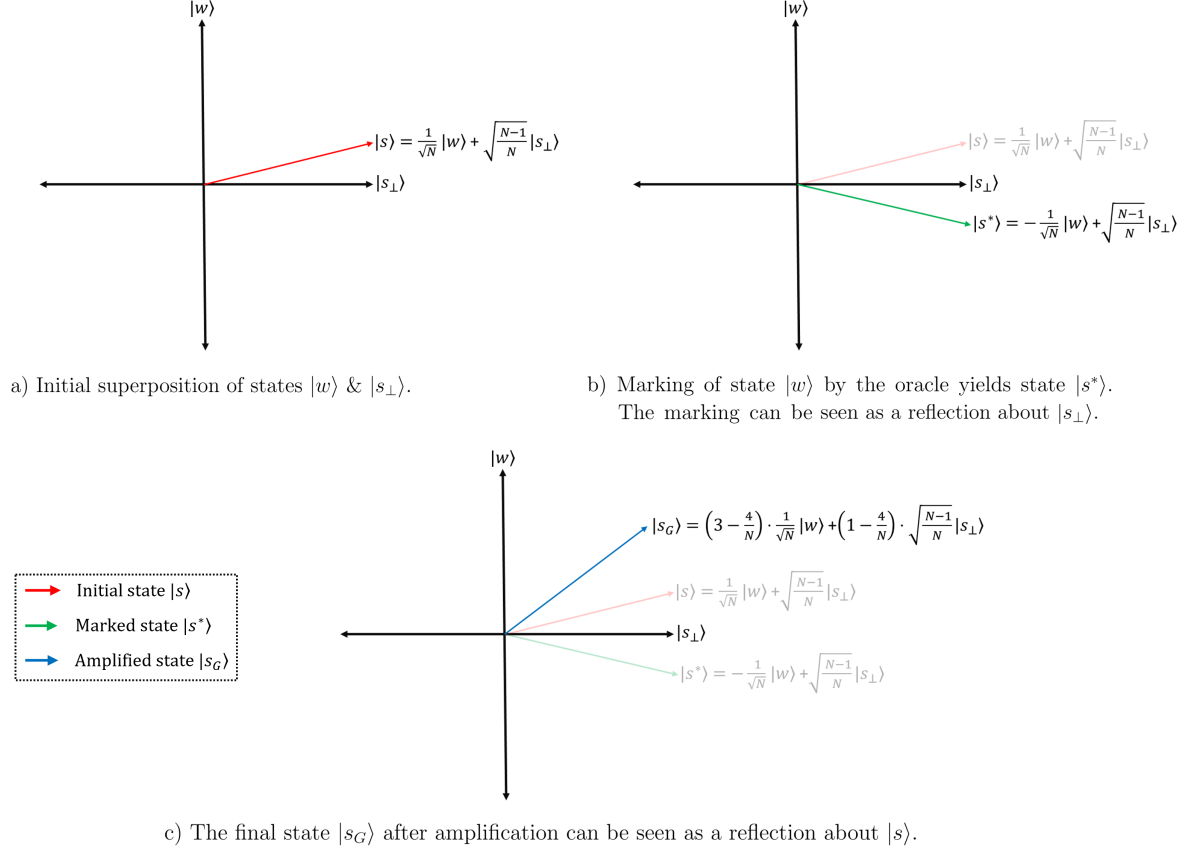


Figure 2.1: The Full Grover Process

2.2.2 The Circuit

To implement the desired changes on the computational basis states that encode information, we build circuits, made up of gates that manipulate qubits in a quantum processor into a desired state. In this section, we take a look at some of these circuits and observe their effects on the systems' states explicitly. Specifically, we discuss 2 circuits implementing a single full Grover iteration, once with a boolean oracle, and another with a phase oracle [12].

A few notes on notation are necessary for this discussion. A $CX_{1,4}$ is a CX -gate

where the control is qubit 1 in the circuit and the target is the 4th qubit where the X -gate is applied. A $CCZ_{12,3}$ gate is a multi-control single-target gate where the controls are qubit 1 and 2, and the target is qubit 3 where the Z -gate is applied.

Other important things to take into consideration are the following:

$$\begin{aligned} |++\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ &= \frac{1}{\sqrt{8}}[|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle] \end{aligned}$$

$$X|-\rangle = -|-\rangle$$

$$Z|+\rangle = Z\left[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right] = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle$$

$$Z|-\rangle = |+\rangle$$

$$CCZ_{12,3}|++\rangle = CCZ_{12,3}\left[\frac{1}{\sqrt{8}}[|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle]\right]$$

This gate only effects the $|111\rangle$ state, resulting in the following superposition:

$$\begin{aligned} &[\frac{1}{\sqrt{8}}[|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle - |\mathbf{111}\rangle]] \\ &\equiv [\frac{1}{\sqrt{8}}[|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |\mathbf{111}\rangle - \mathbf{2}|\mathbf{111}\rangle]] \\ CCZ_{12,3}|++\rangle &= |++\rangle - \frac{2}{\sqrt{8}}|111\rangle \end{aligned}$$

2.2.3 The Boolean Oracle

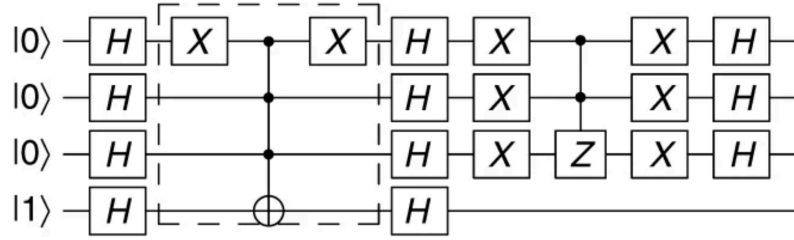


Figure 2.2: Boolean Oracle [12]

Initialization

$$|0\rangle |0\rangle |0\rangle |1\rangle$$

H-Gate

$$|+\rangle |+\rangle |+\rangle |-\rangle$$

$$\frac{1}{\sqrt{8}}[|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle] \otimes |-\rangle$$

$$\mathbf{X}_1$$

$$\frac{1}{\sqrt{8}}[|100\rangle + |101\rangle + |110\rangle + |111\rangle + |000\rangle + |001\rangle + |010\rangle + |011\rangle] \otimes |-\rangle$$

$$\mathbf{CCCX}_{123,4}$$

$$\frac{1}{\sqrt{8}}[|100\rangle + |101\rangle + |110\rangle - |\mathbf{111}\rangle + |000\rangle + |001\rangle + |010\rangle + |011\rangle] \otimes |-\rangle$$

X₁

$$\frac{1}{\sqrt{8}}[|000\rangle + |001\rangle + |010\rangle - |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle] \otimes |-\rangle$$

H-Gate

$$\begin{aligned} \frac{1}{\sqrt{8}}[& |+++\rangle + |++-\rangle + |+-+\rangle - |+--\rangle + |-++\rangle + |--+\rangle \\ & + |--+\rangle + |---\rangle] \otimes |1\rangle \end{aligned}$$

X-Gate(First 3 qubits)

$$\begin{aligned} \frac{1}{\sqrt{8}}[& |+++\rangle - |++-\rangle - |+-+\rangle - |+--\rangle - |-++\rangle + |--+\rangle \\ & + |--+\rangle - |---\rangle] \otimes |1\rangle \end{aligned}$$

CCZ_{12,3}

$$\begin{aligned} \frac{1}{\sqrt{8}}[& (|+++\rangle - \frac{2}{\sqrt{8}}|111\rangle) - (|++-\rangle + \frac{2}{\sqrt{8}}|111\rangle) - (|+-+\rangle + \frac{2}{\sqrt{8}}|111\rangle) \\ & - (|+--\rangle - \frac{2}{\sqrt{8}}|111\rangle) - (|-++\rangle + \frac{2}{\sqrt{8}}|111\rangle) + (|--+\rangle - \frac{2}{\sqrt{8}}|111\rangle) \\ & + (|--+\rangle - \frac{2}{\sqrt{8}}|111\rangle) - (|---\rangle + \frac{2}{\sqrt{8}}|111\rangle)] \otimes |1\rangle \end{aligned}$$

This simplifies to:

$$\begin{aligned} \frac{1}{\sqrt{8}}[& |+++\rangle - |++-\rangle - |+-+\rangle - |+--\rangle - |-++\rangle + |--+\rangle \\ & + |--+\rangle - |---\rangle - \frac{12}{\sqrt{8}}|111\rangle] \otimes |1\rangle \end{aligned}$$

X-Gate(First 3 qubits)

$$\begin{aligned} & \frac{1}{\sqrt{8}}[|+++\rangle + |++-\rangle + |+-+\rangle - |+--\rangle + |-++\rangle + |+-+\rangle \\ & + |--+\rangle + |---\rangle - \frac{12}{\sqrt{8}}|000\rangle] \otimes |1\rangle \end{aligned}$$

H-Gate(First 3 qubits)

$$\frac{1}{\sqrt{8}}[|000\rangle + |001\rangle + |010\rangle - |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle - \frac{12}{\sqrt{8}}|+++\rangle] \otimes |1\rangle$$

Note that the expansion of the $|+++\rangle$ term results in another $\frac{1}{\sqrt{8}}$ term.

$$\frac{12}{\sqrt{8}} \cdot \frac{1}{\sqrt{8}} = \frac{12}{8}$$

Subtracting that from the coefficient of the nonsolution states $1 = \frac{8}{8}$ we get

$$\frac{8}{8} - \frac{12}{8} = -\frac{4}{8}$$

$$\frac{1}{\sqrt{8}} \left[-\frac{4}{8}[|000\rangle + |001\rangle + |010\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle] - \frac{20}{8}|011\rangle \right] \otimes |1\rangle$$

Probability for the solution state $P_{|011\rangle} = \left\| \frac{-20}{8} \cdot \frac{1}{\sqrt{8}} \right\|^2 = \frac{25}{32} \approx 0.78$

while the non-solutions $P_{\neq|011\rangle} = 7 \cdot \left\| \frac{-4}{8} \cdot \frac{1}{\sqrt{8}} \right\|^2 = \frac{7}{32} \approx 0.22$

These are the probabilities of this circuit implementation as suggested in [12].

H-Gate

$$\frac{1}{\sqrt{8}}[|+++ \rangle + |++- \rangle + |+ - + \rangle - |+ - - \rangle + |- + + \rangle - |- + - \rangle + |- - + \rangle + |- - - \rangle]$$

X-Gate

$$\frac{1}{\sqrt{8}}[|+++ \rangle - |++- \rangle - |+ - + \rangle - |+ - - \rangle - |- + + \rangle - |- + - \rangle + |- - + \rangle - |- - - \rangle]$$

CCZ_{12,3}

$$\begin{aligned} & \frac{1}{\sqrt{8}}[(|+++ \rangle - \frac{2}{\sqrt{8}}|111 \rangle) - (|++- \rangle + \frac{2}{\sqrt{8}}|111 \rangle) - (|+ - + \rangle + \frac{2}{\sqrt{8}}|111 \rangle) \\ & - (|+ - - \rangle - \frac{2}{\sqrt{8}}|111 \rangle) - (|- + + \rangle + \frac{2}{\sqrt{8}}|111 \rangle) - (|- + - \rangle - \frac{2}{\sqrt{8}}|111 \rangle) \\ & + (|- - + \rangle - \frac{2}{\sqrt{8}}|111 \rangle) - (|- - - \rangle + \frac{2}{\sqrt{8}}|111 \rangle)] \end{aligned}$$

This simplifies to:

$$\begin{aligned} & \frac{1}{\sqrt{8}}[|+++ \rangle - |++- \rangle - |+ - + \rangle - |+ - - \rangle - |- + + \rangle - |- + - \rangle + |- - + \rangle \\ & - |- - - \rangle - \frac{8}{\sqrt{8}}|111 \rangle] \end{aligned}$$

X-Gate

$$\begin{aligned} & \frac{1}{\sqrt{8}}[|+++ \rangle + |++- \rangle + |+ - + \rangle - |+ - - \rangle + |- + + \rangle - |- + - \rangle + |- - + \rangle \\ & + |- - - \rangle - \frac{8}{\sqrt{8}}|000 \rangle] \end{aligned}$$

H-Gate

$$\frac{1}{\sqrt{8}}[|000\rangle + |001\rangle + |010\rangle - |011\rangle + |100\rangle - |101\rangle + |110\rangle + |111\rangle - \frac{8}{\sqrt{8}}|+++\rangle]$$

Note that the expansion of the $|+++\rangle$ term results in another $\frac{1}{\sqrt{8}}$ term.

$$\frac{8}{\sqrt{8}} \cdot \frac{1}{\sqrt{8}} = 1$$

Subtracting that from the coefficient of the nonsolution states

$$\frac{1}{\sqrt{8}}[-\mathbf{2}|011\rangle - \mathbf{2}|101\rangle]$$

Probability for the solution states $P_{|011\rangle} = P_{|101\rangle} = \left\| \frac{-2}{\sqrt{8}} \right\|^2 = \frac{4}{8} = 0.5$

while the non-solutions $P_{\neq\{|011\rangle, |101\rangle\}} = 0$

2.3 The Oracle

In section 2.1, we discussed how the oracle in Grover's algorithm does not traverse the search space. How then is the oracle able to retrieve the answer to our search problem? Simply, it is because the oracle does not retrieve the state, but only marks it. To the reader being exposed to this concept for the first time, it should not at all be intuitive how this can offer an advantage over the classical counterpart of brute search. After all, it seems that in both primitives one needs to traverse the entire database to find the solution. The key to understanding the advantage of the quantum algorithm lies in two facts:

1. The oracle $f(x)$ is **applied once** to a superposition of the computational basis states storing the data.
2. Marking, and retrieving the desired state(s) are distinct concepts.

Under the primitive of the Grover search, information from the unsorted database is encoded into the computational basis states of the qubits making up the circuit used for the search. The number of computational basis states grows as 2^n where n is the number of qubits used in the circuit. A simple example of basis encoding of a random sample of data is shown in Figure 2.4. Since data encoding in quantum circuits is still a work under progress, we will discuss the simplest situation where an equal superposition of computational basis states is created using Haddamard (H) gates. The H -gate behaves in the following manner when applied to qubits:

$$\begin{aligned} H|0\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle \\ H|1\rangle &= \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle \end{aligned} \tag{2.1}$$

$$\begin{array}{lclcl}
2 & \rightarrow & 0010 & \rightarrow & |0010\rangle \rightarrow |2\rangle \\
15 & \rightarrow & 1111 & \rightarrow & |1111\rangle \rightarrow |15\rangle \\
4 & \rightarrow & 0100 & \rightarrow & |0100\rangle \rightarrow |4\rangle \\
8 & \rightarrow & 1000 & \rightarrow & |1000\rangle \rightarrow |8\rangle \\
13 & \rightarrow & 1101 & \rightarrow & |1101\rangle \rightarrow |13\rangle
\end{array}$$

Figure 2.4: An example of information being encoded into computational basis states. The first column is of a random sample of numerical data, the second is of the corresponding binary representation. The third column is of the corresponding computational basis states and the fourth is the decimal representation of the basis.

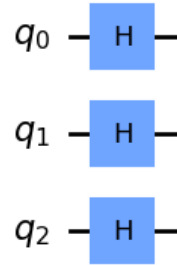


Figure 2.5: Application of Haddamard gates to create a uniform superposition of computational basis states in a 3-qubit circuit.

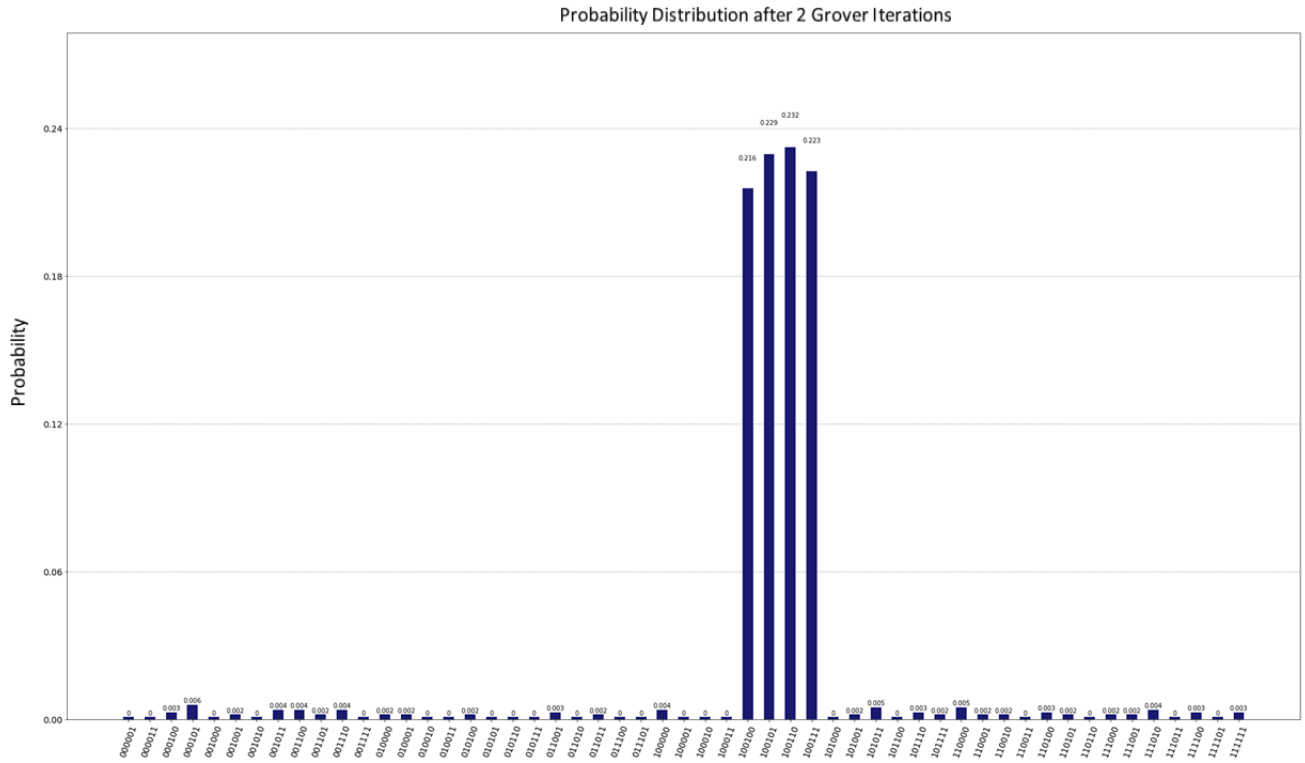
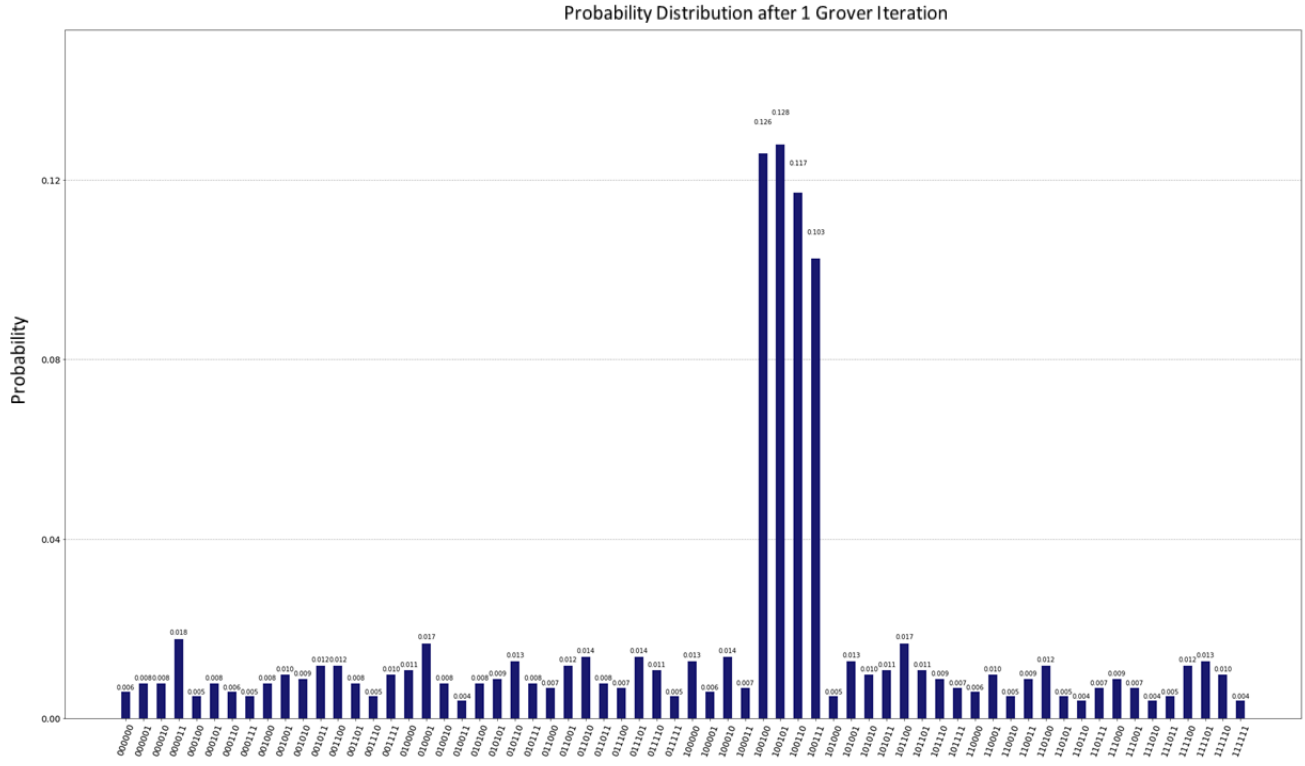
Application of the H -gates to the circuit qubits initialized to $|0\rangle$ creates a uniform superposition. Imagine we have 3 qubits in our circuit. The outcome of the application of an H -gate to each qubit as shown in Figure-2.5, is demonstrated in Equation-2.2. The ket subscripts $\{0, 1, 2\}$ are used to denote the qubit in the 3-qubit system. After the H -gate takes the qubit to an equal superposition of $|0\rangle$ and $|1\rangle$, the tensor product of the produced state $|+\rangle$ creates a uniform superposition of the computational basis states comprising each of the constituent states of the qubits, $|s\rangle$.

$$\begin{aligned}
& H|0\rangle_0 \otimes H|0\rangle_1 \otimes H|0\rangle_2 \\
&= \frac{1}{\sqrt{2}}(|0\rangle_0 + |1\rangle_0) \otimes \frac{1}{\sqrt{2}}(|0\rangle_1 + |1\rangle_1) \otimes \frac{1}{\sqrt{2}}(|0\rangle_2 + |1\rangle_2) \\
&= \frac{1}{\sqrt{8}}(|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle) \\
&= |s\rangle
\end{aligned} \tag{2.2}$$

The next task at hand is then to mark a particular state (or states if $M > 1$) such that the new superposition holding the marked state, $|s^*\rangle$, can go through the diffuser for amplification. That is the sole purpose of the oracle in Grover's algorithm. Assume that the desired state was $|010\rangle$, $|s^*\rangle$ would be:

$$\frac{1}{\sqrt{8}}(|000\rangle + |001\rangle - |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle)$$

It is worth expanding on the intrinsically different philosophies of the search in the quantum and the classical case. Classically, the searcher knows what they are looking for, so they check as many entries of the database as needed until they match with the desired state. In the quantum algorithm, the searcher also knows what they are looking for, but **instead of traversing the database, they create another circuit, catered only to the marking of the desired state in the superposition.** This circuit, along with the diffuser, acts $\approx \frac{\pi}{4}\sqrt{\frac{N}{M}}$ times on the qubits initially set to $|s\rangle$, with each Grover iteration skewing the probability distribution of measuring the states towards the solution state until the final iteration.



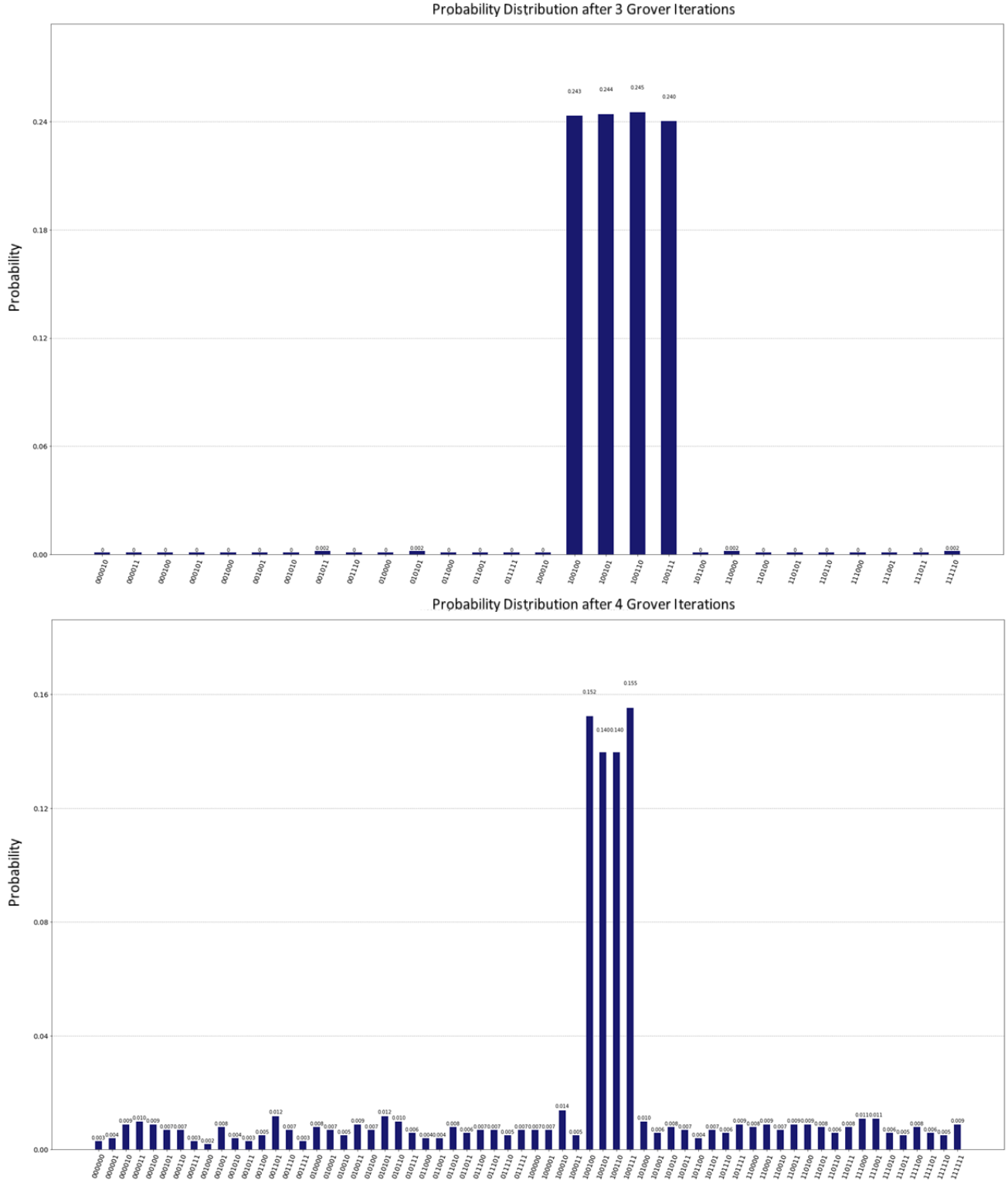


Figure 2.6: Evolution of the probability distribution of measured states across different Grover iterations for a 6-qubit system with $M = 4$. The possible basis states are on the x-axis and their corresponding probabilities on the y-axis. The solution states in this example are $|001001\rangle$, $|101001\rangle$, $|011001\rangle$, $|111001\rangle$.

It is important to note that running the circuit more times beyond the ideal number of iterations results in probability distributions that are less likely to yield solution states. In a 6-qubit system for example where there exists 4 solution states, the ideal number of iterations is $\approx \frac{\pi}{4} \sqrt{\frac{2^6}{4}} \approx 3.1$. This means that we should not run the circuit executing the Grover iteration for more than 3 times. The evolution of the probability distribution of the states of such a system can be seen in Figure-2.6 for 4 iterations. The distribution is optimal in its skewing towards the solution states by the third iteration, with probabilities ≥ 0.24 for each of the 4 states i.e. the probability of measuring a solution state is ≥ 0.96 . From there, as expected, the final iteration deviates from the ideal probability distribution seen with only 3 iterations, reducing the probability of measuring the solution states.

Shifting focus back to the circuit tasked with marking the desired solution state, we are presented with the question: For a certain state $|x\rangle$, how does one construct an oracle that exclusively marks $|x\rangle$ in the superposition $|s\rangle$? As of the writing of this paper, efforts to find oracles for n-qubit systems are underway [12] [19] [24].

Chapter 3

Gravitational wave matched-filtering

3.0.1 Matched Filtering

Signals do not normally arrive at detectors without a leeching guest; noise. This noise could distort the signal, or even completely mask it. Matched filtering is a process that can be used to extract a signal of interest from the noise it is embedded in [4]. The outcome of passing a noisy signal through a matched filter, is not actually the signal itself. Instead, a peak, or chirp at the beginning of the embedded signal with a relatively higher signal to noise ratio (SNR) as shown in Figure 3.1. There are two important points to be observed here. Firstly, the matched filter enables us to find where the embedded waveform starts in the noisy signal. Secondly, in the situation where a template bank of waveforms is being studied, the highest output SNR from the matched filter corresponds to the most likely waveform embedded in the noise.

The problem is set up in the following manner: Start with some noisy data (interesting waveform + noise) denoted as $x(t)$, as well as a model(s) of what the embedded waveform would look like without the noise. This will be used to calculate the convolution $y(t)$, using $x(t)$.

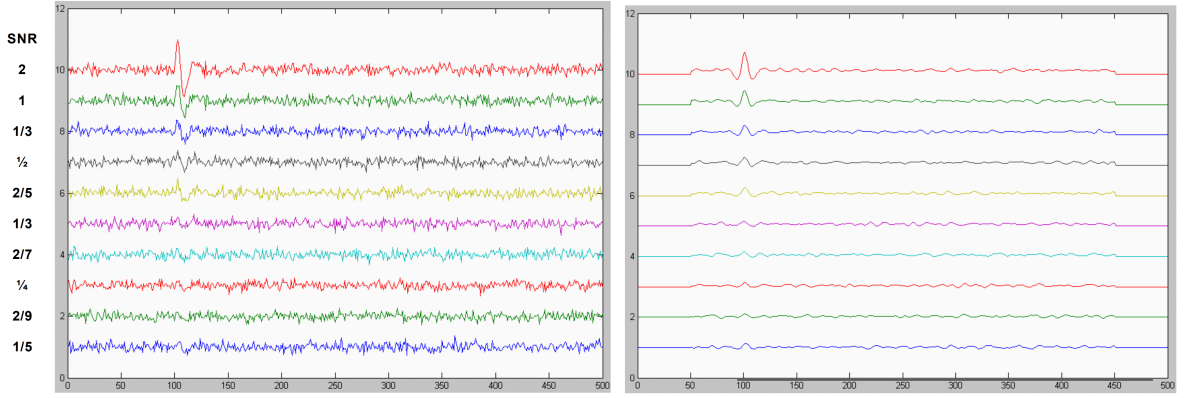


Figure 3.1: Left: Noisy signals with different SNR values. Right: Corresponding output signals from the matched filter. It can be seen that the output signals, are of a higher SNR compared to the input signals. They also appear at the point in time, where the corresponding embedded waveform in the input signals start [4].

The convolution $y(t)$ is defined as:

$$y(t) = \int_{-\infty}^{\infty} x(t)s(\tau - t)d\tau \quad (3.1)$$

where $s(\tau - t)$ is the time-reversed and shifted waveform, $\tau = t - t_o$, and t_o is the sampling time; the point in time in the data we're applying the match filter to [18].

The procedure is as follows:

1. Time reverse the model waveform.
2. Calculate the convolution $y(t)$, of the signal $x(t)$, with the time-reversed waveform $s(\tau - t)$.

It is important to note that the output of the matched filter, is not an improved version of the input signal, but rather, it is a signal that is higher in SNR (if the waveform chosen is correct) that appears at the point in time where the embedded waveform in the input signal starts.

3.0.2 Gravitational wave detection

Oscillations in the fabric of space-time, predicted by Einstein's theory of general relativity [9] [10] propagate through millions of light years at the speed of light, and if we are equal parts lucky and prepared, we just might be able to detect them. The first ever gravitational wave observed from event GW150914; a merger of two black holes, was detected by the Laser Interferometer Gravitational-Wave Observatory (LIGO) in 2015 [1]. Scientists at two independent locations managed to detect a perturbation in GW strain. However, given the immense distance this gravitational wave had to travel to reach, and consequently the attenuation it had suffered prior to registering at the detectors, scientists at LIGO had to use matched filtering to be able to extract the actual signal of the wave from the noise it is embedded in.

This happens by first looking at a bank of numerous templates of theoretically predicted GW waveforms, the number of which could reach 10^{12} . A Signal to Noise Ratio (SNR) is then computed by using information from the signal received at the detector and each template from the bank. That is 10^{12} SNR computations corresponding to 10^{12} templates from the bank. The template(s) with the highest SNR is chosen, and is used to identify and locate the embedded signal of interest in the detected, noisy signal. The search and computation associated with this large template bank for this problem scales as $\mathcal{O}(NM \log M)$. Fortunately, we currently

do have the computational prowess to perform these computations, but, we can do it much more efficiently.

Grover's Algorithm can help us achieve complexity proportional to

$$\mathcal{O}\left((M \log M + \log N) \cdot \sqrt{N}\right) [13]$$

The waveform templates from the bank in this procedure are prepared in superposition. The template(s) that satisfies a pre-set SNR value is marked. The following distinction needs to be made: The template is only marked, it is not retrieved.

The second important part of the Algorithm is amplification. Assume out of a 1,000 templates, 2 happen to be adequate solutions. This means that the algorithm would have so far marked these 2 templates. What happens next is an amplification of those templates so they are more apparent to the observer upon measurement. This process of marking and amplification is then repeated multiple times until virtually nothing but our solutions are observable, and all non-solutions are suppressed.

The final part is measurement. Having amplified our solution templates of interest, once a measurement is made, the superposition of all the templates collapses to only one, now with a high probability of that measured template being one of the solution templates.

3.0.3 The Circuit

We assume that the signal we detect can be represented by a string of 6 bits, 100101 for example. This string is encoded into the 6 qubits labelled q_i . The templates are also represented by 6-bit strings. To simulate the different templates available in our template bank, we use all the possible permutations of a 6-bit string where the bits take up a value of 0 or 1 exclusively. The 6 qubits that follow in the circuit, labelled t_i , are used to encode all the templates from the bank in the form of a superposition using Hadamard gates. We use an ancillary qubit register to store the phase kickback if it were to occur.

We would like to include the possibility of multiple templates matching with our signal. This is equivalent to setting an SNR threshold as a condition for accepting a template as a match. The way we achieve this is by excluding a select number of qubits during the phase kickback operation (oracle). The more qubits we choose to exclude, the more relaxed our SNR threshold equivalent is, and vice versa. We choose to neglect 2 qubits from the data register (q_i) in our circuit. The way this would work for our example with signal 100101, is instead of only matching with template 100101, the signal can match with templates: 1001**00**, 1001**10** and 1001**11**.

With the use of CNOT Gates, qubits t_i are converted to a string of 0's if they are a match to our input signal, given the degree of relaxation we allowed for. They are bit-flipped into a string of 1's and are then passed through a multi-control-NOT Gate ($C^4\text{NOT}$). It can be shown that the ancillary qubit, which is initialized to state $|-\rangle$, attains a phase of -1 if the template was indeed a match as discussed

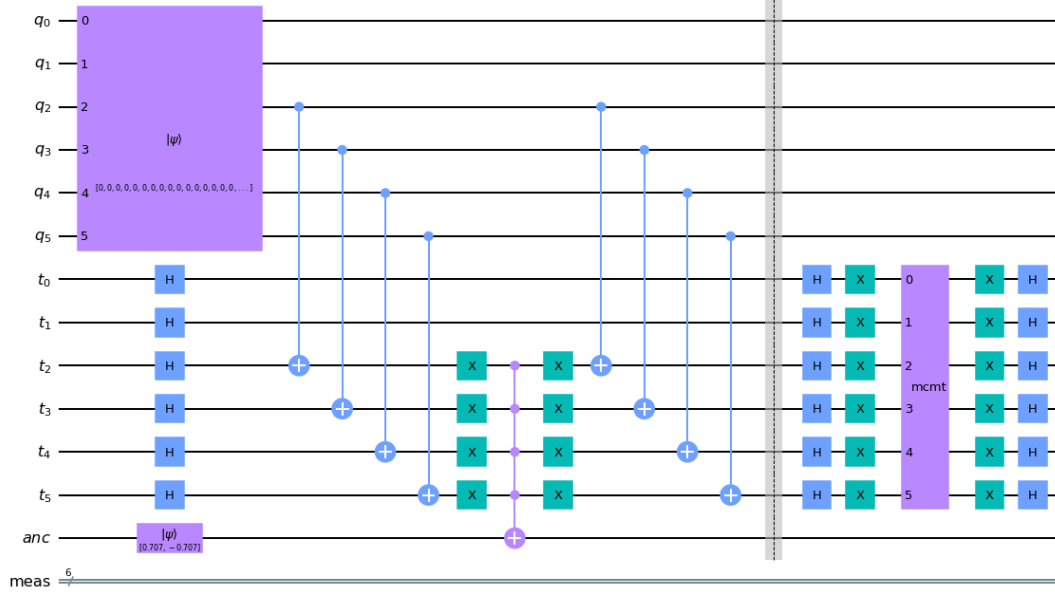


Figure 3.2: Recreation of the Quantum Oracle Circuit from [13]. The circuit is split into two sections, separated by the dashed barrier in the middle. The first section is the phase kickback operation (Oracle). Qubits q_i encode the signal, and qubits t_i store a superposition of all possible templates. The ancillary qubit (anc) is used to store phase kickback feedback. The measurement classical bit ($meas$) is used to store measurements of t_i 's performed after the diffusion operator.

in 2.2.3. The rest of the circuit in this first section, reverts the template back to its original encoded string value. The diffusion operator in the second part of the circuit consists of Hadamard Gates, X Gates and a multi-control-multi-target (MCMT) Z Gate. The MCMT Gate (C^5Z), uses the first 5 qubits of the template register as controls, and the 6th as the target.

It is important to note that this circuit needs to be run $\leq \frac{\pi}{4} \sqrt{\frac{N}{M}}$ times for the most ideal result in which we are most likely to measure a desired solution state [17]. In our case, N is 64 because 6 binary bits can be arranged differently for a total of $2^6 = 64$ different combinations, and $M = 4$ because there are 4 possible template solutions to our matching problem given the exclusion of the 2 qubits

aforementioned. Hence, the number of Grover iterations is $\leq \frac{\pi}{4}\sqrt{\frac{64}{4}} = \pi \approx 3.14$. Therefore the ideal number of iterations for this example is 3. The probability distributions observed following the measurement of the qubits in the circuit, across different iterations, are shown in Figure [2.6](#).

Chapter 4

The n-qubit Oracle

An interesting question to ask at this point, is whether there can exist an oracle primitive that is generalizable to n-qubits, not just 3 or 4. To do this we first explore an extension of the boolean oracle discussed in [2.2.3](#).

4.1 An n-qubit boolean oracle

In this section, we take another look at the boolean oracle that made use of an extra ancillary qubit and multi-control single-target X gate to store the output of the phase kickback operation and explore whether the circuit can be extended to the case of 5 qubits for the arbitrary state $|01001\rangle$. The circuit in [Figure 4.1](#) was run for $\frac{\pi}{4}\sqrt{\frac{2^5}{1}} \approx 4$ iterations.

This was run for 10,000 counts on a simulator and the resulting probability distribution is shown in [Figure 4.2](#). The target state $|00101\rangle$ was indeed the most probable outcome of measurement, however, the probability was only $\approx 33.7\%$. This means that the circuit yields the desired state less often than a coin toss would, and is therefore unacceptable as a reliable implementation for Grover's algorithm.

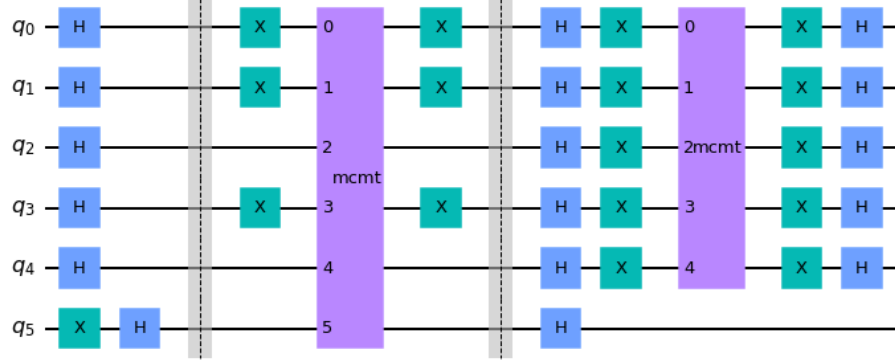


Figure 4.1: An extension of the boolean oracle proposed in [12], to a 5-qubit search for the state $|00101\rangle$. The ancilla qubit (q_5) is initialized to $|1\rangle$, and then the entire qubit register is put into a maximally entangled superposition using Hadamard gates. The oracle embedded between the two barriers, consists of X gates placed before and after the $C^5X_{12345,6}$. The diffuser after the second barrier is identical to what has been previously described in 2.2.3 but with a $C^4Z_{1234,5}$ instead.

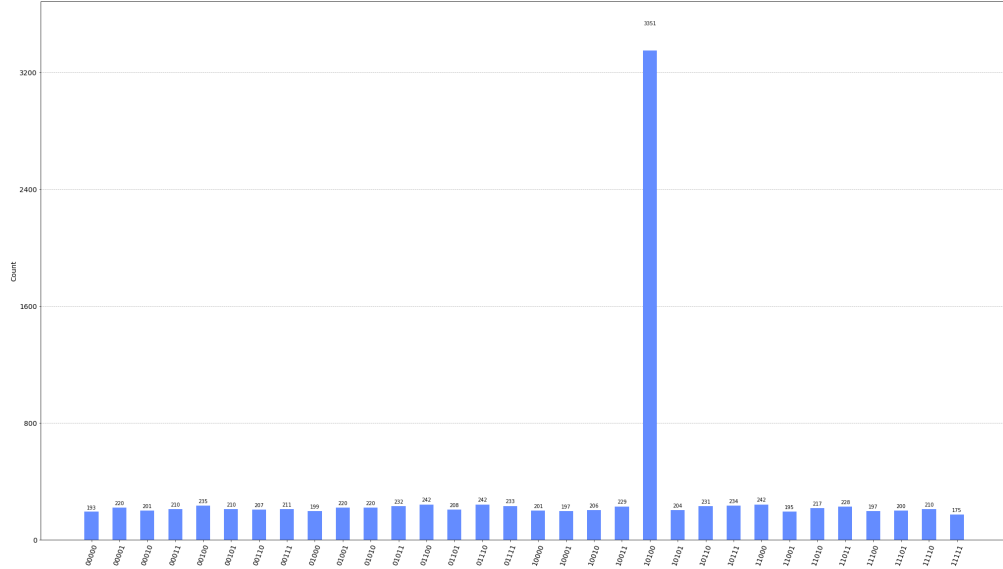


Figure 4.2: The probability distribution of measured states for the circuit in Figure 4.1 after 10,000 runs. The state $|00101\rangle$ was most likely to be observed with a probability $\approx 33.7\%$

Similar results were found for the $n \geq 6$ n-qubit boolean oracle circuits.

4.2 A viable oracle

Extending the same logic executed in 4.1 to the phase oracle which now negates the need of an ancilla qubit and uses a multi-control single-target Z gate ($C^{n-1}Z$), the 5-qubit phase oracle circuit in Figure 4.3 was also run 4 times yielding the probability distribution in Figure 4.4. The results when using the phase oracle are diametrically more promising compared to the n-qubit boolean oracle attempt.

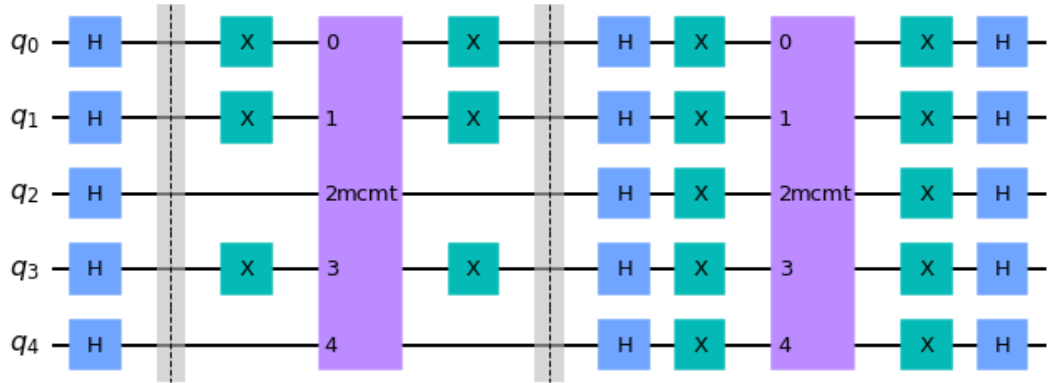


Figure 4.3: An extension of the phase oracle proposed in [12], to a 5-qubit search for the state $|00101\rangle$. The entire qubit register is put into a maximally entangled superposition using Hadamard gates. The oracle embedded between the two barriers, consists of X gates placed before and after the $C^4Z_{1234,5}$. The diffuser is identical to the aforementioned boolean oracle in 4.1.

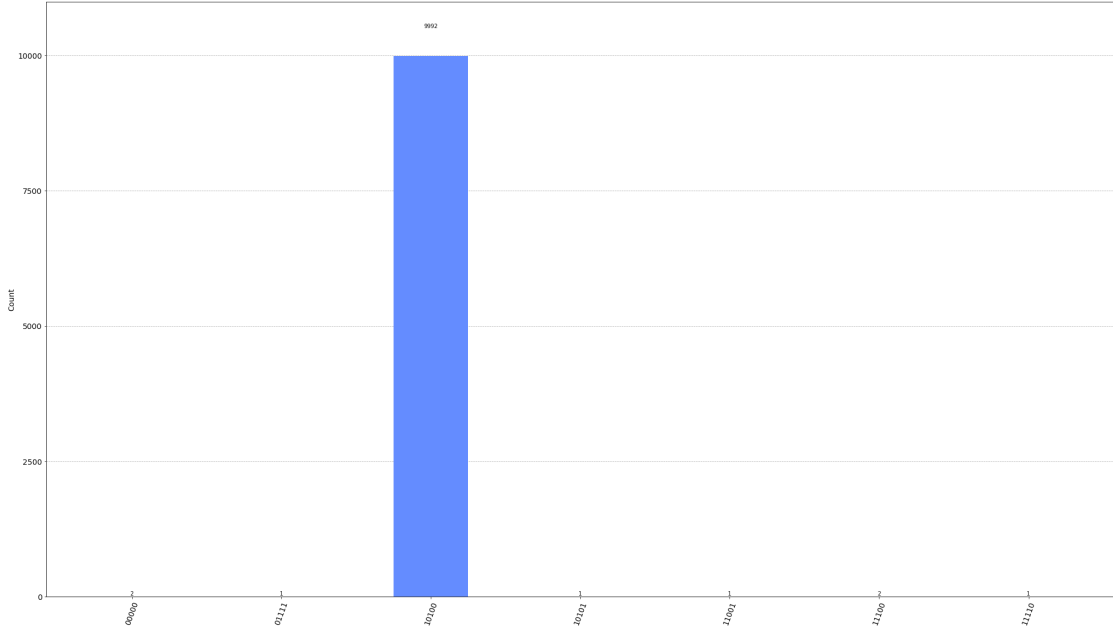


Figure 4.4: The probability distribution of measured states for the circuit in Figure 4.3 after 10,000 runs. The state $|00101\rangle$ was most likely to be observed with a probability $\approx 99.9\%$.

4.3 A dynamic oracle

The purpose of the oracle in Grover’s algorithm is to mark a desired state(s) without undergoing a classical search of the elements being searched over. The convention is that the quantum circuit implementing this oracle step, is static, unchanging throughout the Grover iterations. Now that we find promise in the generalized phase oracle discussed in 4.2, is it possible that we can alter its construction to decrease the circuit depth while maintaining its ability to retrieve all possible computational basis states?

In this work we find promising results for a dynamic, generalizable oracle. One that does not necessarily take the same form throughout the Grover iterations. **If the optimal number of Grover iterations is T , we find that it is possible to retrieve any desired state by first marking and amplifying the $|1\rangle^{\otimes n}$ state in our system $T - 1$ times using a multi-control single-target $C^{n-1}Z$ gate, and then altering the final step of the oracle implementation by, appending X gates in a sequence corresponding to the 0's in the state of interest, to the $C^{n-1}Z$ gate.** This can be extended to all possible computational basis states in our space, with the exception of the $|1\rangle^{\otimes n}$ state (all the qubits in state $|1\rangle$) such as, $|111\rangle$ for 3 qubits, $|1111\rangle$ for 4 qubits, etc. This type of state can be retrieved using the $C^{n-1}Z$ gate alone as can be seen in the catalogue in [Figure-A.2](#).

The following outlines a formal description of the generalizable n-qubit Grover search algorithm that marks and retrieves any desired state in a $2^n = N$ superposition of computational basis states with a minimum accuracy of 94%, and an accuracy $> 99\%$ for n-qubit systems where $n \geq 5$.

Algorithm 1 n-qubit generalizable Grover's algorithm**Input:** $|\text{Desired State}\rangle$ **Output:** $|\text{Desired State}\rangle$ with high probability*Initialisation:* Apply H gates to n-qubit register $Iterations = \text{floor}(\frac{\pi}{4}\sqrt{\frac{N}{M}})$ *LOOP process applied to qubit register*

- 1: **for** $d = 0$ to $Iterations - 2$ **do**
- 2: Apply $C^{n-1}Z$ gate
- 3: Apply X gates
- 4: Apply H gates
- 5: Apply $C^{n-1}Z$ gate
- 6: Apply H gates
- 7: Apply X gates
- 8: **end for**
- 9: Apply X gates to the 0 state qubits in $|\text{Desired State}\rangle$
- 10: **return** Probability distribution.

The **loop** in the algorithm is run for the range 0 to $iterations - 2$. For example, if iterations=3, the loop would run for the range [0,1] i.e. the loop would run twice.

For the 3-qubit circuit concerned with the retrieval of state $|100\rangle$ shown in Figure-4.5, the Grover iteration was performed $\frac{\pi}{4}\sqrt{\frac{N}{M}} = \frac{\pi}{4}\sqrt{\frac{2^3}{1}} \approx 2$ times. The first iteration of the oracle consisted of only the $C^{n-1}Z$ gate, and the second, of the $C^{n-1}Z$ gate appended by $2X$ gates corresponding to the 0's in the state $|100\rangle$. It can be seen that the desired probability expected from a Grover circuit is achieved, in line with the number of iterations from the theory.

The same circuit in Figure-4.5 is once again ran on a noisy quantum processor using an IBM backend. We do this as a proof of concept for the dynamic oracle. It can be seen in Figure-4.6 that the run yields a similar probability distribution as the simulator, with (as expected) less accuracy due to the noise in the backend.

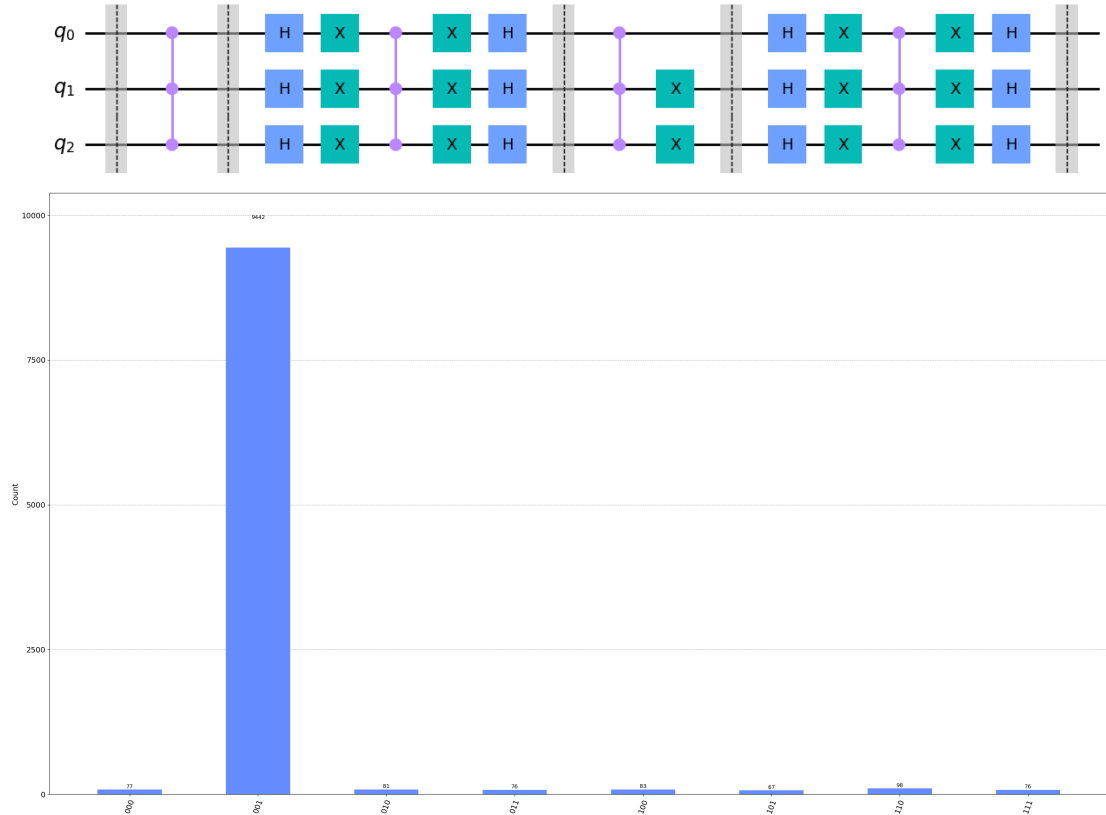


Figure 4.5: The Eye of Horus primitive for a 3-qubit system Grover search for state $|100\rangle$ and the corresponding probability distribution of states after circuit simulation.

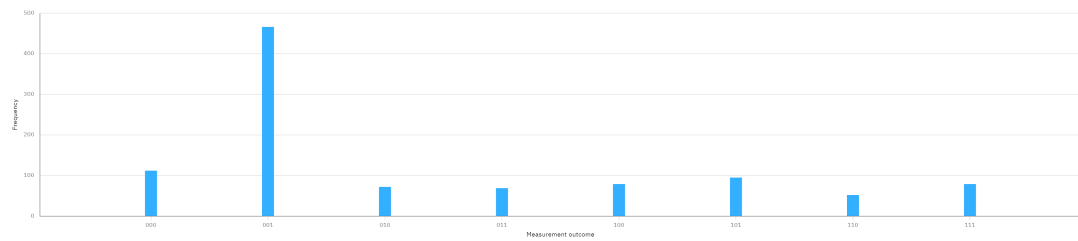


Figure 4.6: Probability distribution after running a Grover circuit using the n-qubit generalizable Grover's algorithm for 3-qubits on a noisy quantum processor. JobId: 640964c55fbdedb851894a2d

The only setback with using this algorithm is the rapid increase of the circuit depth with the number of qubits in the system. This arises from the $C^{n-1}Z$ gate used in the oracle section, as well as the diffuser. Figure-4.7 demonstrates this using an exponential plot as reference. Early steps in future work should focus on the marking of the $|1\rangle^{\otimes n}$ state in the $T - 1$ oracle steps using a more efficient method that does not involve an n -sized gate. Improvements can also be made independently on the diffuser primitive. This will drastically improve the circuit depth of the search problem.

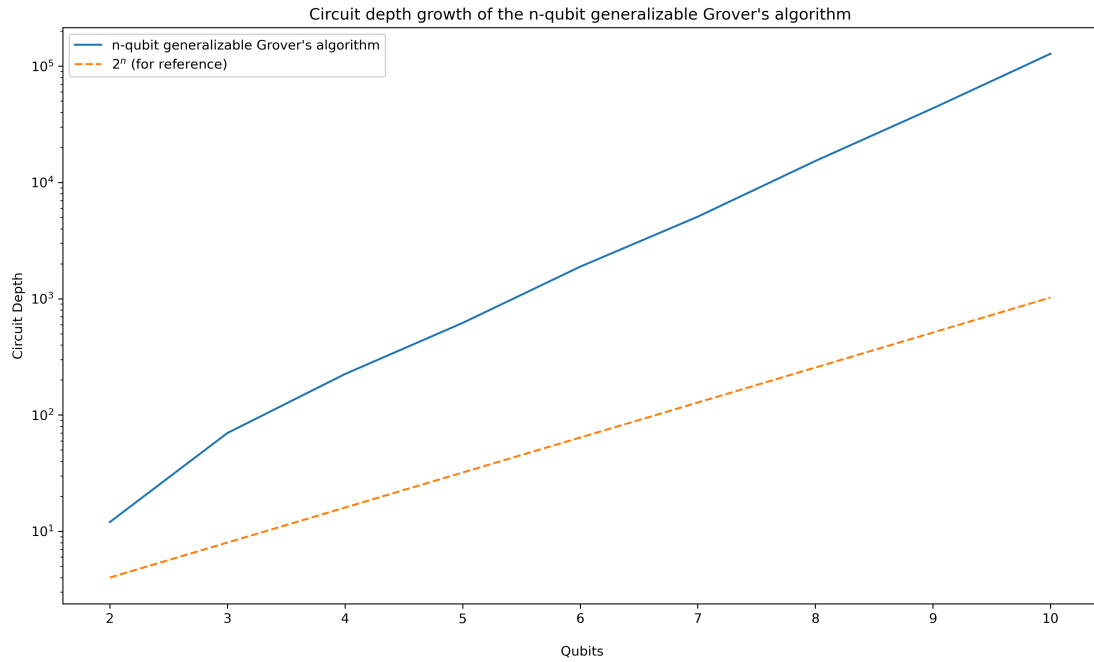


Figure 4.7: The increase in circuit depth on a logarithmic scale for the n -qubit generalizable Grover's algorithm with an increasing number of qubits in the circuit. An exponential plot is plotted for reference.

Chapter 5

Entangled Registers and qRAM

In this section, we discuss a database search problem and the potential of the use of entangled registers to retrieve a desired entry that holds a number of fields S (ID, name, phone number, etc.) in an N large database. We will use the ID in the entry of fields as the unique identifier of this search problem. We assume a function $E(x)$ exists that can encode the fields of the database entry into computational basis states, **uniquely and purely based on the content** of the data it is encoding:

$$987219876313 \rightarrow |3\rangle$$

$$821973982173 \rightarrow |21\rangle$$

$$123798612989 \rightarrow |5\rangle$$

$$000923821871 \rightarrow |9\rangle$$

$$329187120908 \rightarrow |43\rangle$$

The decoding of the computational basis states would **exclusively** return the data initially encoded. We call this the content encoding.

We also encode the database entries based on their position. This will necessitate a circuit that attaches a unique basis state from an indexing register to each of the field entries encoded earlier. We call this positional encoding:

$$987219876313 \rightarrow |3\rangle \otimes |0\rangle$$

$$821973982173 \rightarrow |21\rangle \otimes |1\rangle$$

$$123798612989 \rightarrow |5\rangle \otimes |2\rangle$$

$$000923821871 \rightarrow |9\rangle \otimes |3\rangle$$

$$329187120908 \rightarrow |43\rangle \otimes |4\rangle$$

In this type of search problem, the searcher knows what he is looking for (ID 123798612989 for example), therefore, also knows the corresponding content encoding; $|5\rangle$ in this case, but nothing about the positional encoding. We can use this information to construct an operator from the content encoding basis:

$$\begin{aligned} U_{Oracle} &= |s\rangle \langle s| - 2 |n\rangle \langle n| \\ &= |3\rangle \langle 3| + |21\rangle \langle 21| + |5\rangle \langle 5| + |9\rangle \langle 9| + |43\rangle \langle 43| - 2 |5\rangle \langle 5| \end{aligned}$$

where $|s\rangle$ is the superposition of all database entries, and $|n\rangle$ is the encoded content basis state of the entry of interest. This operator will be applied to the register that contains the content basis only, we can therefore ignore the positional basis for a moment.

Applying U_{Oracle} to the superposition $|s\rangle$:

$$U_{Oracle} |s\rangle = (|s\rangle \langle s| - 2 |n\rangle \langle n|) |s\rangle = |s\rangle - 2 |n\rangle = |s^*\rangle$$

Replacing, $|n\rangle$ with $|5\rangle$ in our example:

$$\begin{aligned}
 &= |3\rangle + |21\rangle + |5\rangle + |9\rangle + |43\rangle - 2|5\rangle \\
 |s^*\rangle &= |3\rangle + |21\rangle - |5\rangle + |9\rangle + |43\rangle
 \end{aligned}$$

We now apply the diffusion operator $U_D = 2|s\rangle\langle s| - I$

$$\begin{aligned}
 U_D |s^*\rangle &= (2|s\rangle\langle s| - I) |s^*\rangle \\
 &= |3\rangle + |21\rangle + 3|5\rangle + |9\rangle + |43\rangle
 \end{aligned}$$

$|5\rangle$ has now been amplified relative to the other states. We can of course repeat this process enough times until the amplification is significant enough for us to, with high probability, measure $|5\rangle$. The number of iterations is $\leq \frac{\pi}{4}\sqrt{N}$ where N is the number of entries in the superposition $|s\rangle$. We then re-introduce the positional basis/encoding:

$$|3\rangle \otimes |0\rangle + |21\rangle \otimes |1\rangle + 3|5\rangle \otimes |2\rangle + |9\rangle \otimes |3\rangle + |43\rangle \otimes |4\rangle$$

Performing a measurement on the content basis register, the superposition collapses to $|5\rangle \otimes |2\rangle$. A measurement on the positional basis register will then collapse to $|2\rangle$. We now know that our entry of interest is in index 2 of the database being searched over and can therefore easily retrieve the pertinent information for the rest of the entry.

The process described above borrows from the idea of quantum random access memory (qRAM) [15], a necessary component for Grover's algorithm to be of any utility. When searching over a classical database, one makes use of a piece of information to retrieve another. This would be the ID, for example, to retrieve a corresponding name. There would no purpose of searching over a databse of isolated IDs that do not connect to other information of interest. The purpose of qRAM is the same as RAM in the classical sense.

$$\sum_j \psi_j |j\rangle_a \xrightarrow{\text{qRAM}} \sum_j \psi_j |j\rangle_a |D_j\rangle_d$$

Having created a register of qubits that encodes addresses of information $|j\rangle_a$ (index in a database for example), qRAM, entangles another register of data $|D\rangle_d$ to $|j\rangle_a$ such that information corresponding to the addresses can be retrieved once the operations of interest are ran on the quantum processor.

Chapter 6

Conclusions and Future Work

We have discussed the inner-workings of Grover’s search algorithm, and provided a more explicit description of the evolution of the states in the Grover circuit for both the boolean and phase oracle. Moreover, we successfully replicated an implementation of Grover’s algorithm for gravitational wave matched filtering using IBM’s simulators. Furthermore, we particularly focused on the development of the oracle circuit. Pursuant to that, we proposed the n-qubit generalizable Grover’s algorithm, a global oracle able to retrieve any desired state from a superposition of computational basis states with an accuracy $\geq 94\%$. The advantage of this algorithm lies in its ability to retrieve any desired state, by implementing a dynamic oracle that makes use of X gates for the final iteration of the optimal number of Grover iterations. We performed simulations and also ran the proposed circuit on NISQ devices from IBM’s cloud backends for proof of concept. We also discussed the scaling of the circuit depth of this implementation with an increase in the number of qubits of the Grover circuit and have found it to be larger than an exponential growth due to the use of the multi-control single-target $C^{n-1}Z$ gate

in the oracle and diffuser. We also theoretically outlined a method to exploit entangled registers to retrieve the index of an entry in a database using qRAM. Future work will include a focus on finding a set of gates that are equivalent in function to the $C^{n-1}Z$ gate with a lower circuit depth , as well as the incorporation of error-correction/mitigation techniques to run the algorithm for $n > 3$ n-qubit circuits on NISQ devices.

Appendix A

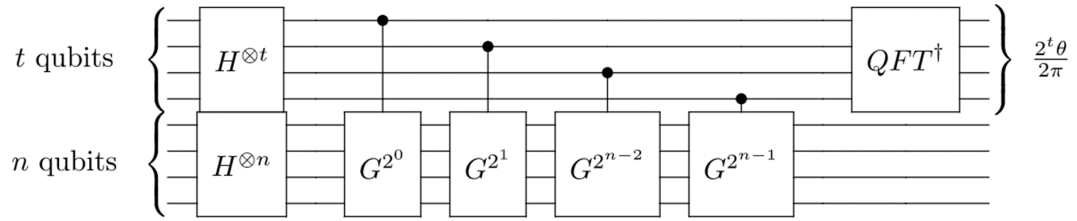


Figure A.1: Quantum Counting circuit for estimating the number of solutions M in a Grover search. t are the counting qubits, and n are the searching qubits. After initialization using Hadamard gates on both registers, a series of controlled Grover iterations are appended onto the circuit before the final inverse QFT is applied to the counting register.[\[3\]](#).

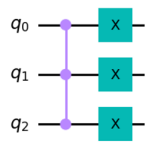
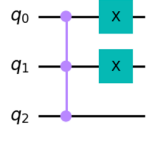
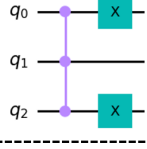
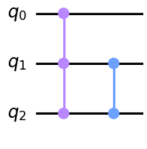
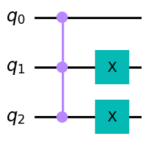
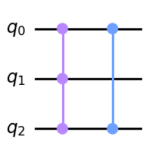
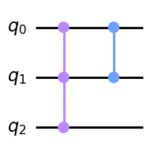
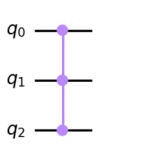
State	Oracle
$ 000\rangle$	
$ 001\rangle$	
$ 010\rangle$	
$ 011\rangle$	
$ 100\rangle$	
$ 101\rangle$	
$ 110\rangle$	
$ 111\rangle$	

Figure A.2: A catalogue of possible oracle circuits corresponding to single 3-qubit states. The purple coded 3-qubit gate is a $C^2Z_{12,3}$ gate, and the blue 2-qubit gate is a CZ gate.

Bibliography

- [1] B. P. Abbott, R. Abbott, T. D. Abbott, Abernathy et al., LIGO Scientific Collaboration, and Virgo Collaboration. Observation of Gravitational Waves from a Binary Black Hole Merger. , 116(6):061102, Feb. 2016.
- [2] D. Adams, C. Alduino, and F. Alessandria et al. CUORE opens the door to tonne-scale cryogenics experiments. *Progress in Particle and Nuclear Physics*, 122:103902, jan 2022.
- [3] M. S. ANIS, Abby-Mitchell, and H. Abraham. Qiskit: An open-source framework for quantum computing, 2021.
- [4] J. C. Bancroft. Introduction to matched filters - crewes, 2002.
- [5] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of the IEEE International Conference on Computers, Systems and Signal Processing*, pages 175–179. Bangalore, India, 1984.
- [6] G. Brassard, P. HØyer, and A. Tapp. Quantum counting. In *Automata, Languages and Programming*, pages 820–831. Springer Berlin Heidelberg, 1998.
- [7] G. Brassard, P. HØyer, and A. Tapp. Quantum cryptanalysis of hash and

- claw-free functions. In *LATIN'98: Theoretical Informatics*, pages 163–169. Springer Berlin Heidelberg, 1998.
- [8] J.-S. Chen, J. Trerayapiwat, L. Sun, M. Krzyaniak, M. Wasielewski, T. Rajh, S. Sharifzadeh, and X. Ma. Long-lived electronic spin qubits in single-walled carbon nanotubes. *Nature Communications*, 14, 02 2023.
 - [9] A. Einstein. Näherungsweise Integration der Feldgleichungen der Gravitation. *Sitzungsberichte der Königlich Preußischen Akademie der Wissenschaften (Berlin)*, pages 688–696, Jan. 1916.
 - [10] A. Einstein. Über Gravitationswellen. *Sitzungsberichte der Königlich Preußischen Akademie der Wissenschaften (Berlin)*, pages 154–167, Jan. 1918.
 - [11] R. P. Feynman. Simulating physics with computers. *International journal of theoretical physics*, 21(6/7):467–488, 1982.
 - [12] C. Figgatt, D. Maslov, K. A. Landsman, N. M. Linke, S. Debnath, and C. Monroe. Complete 3-qubit grover search on a programmable quantum computer. *Nature Communications*, 8(1), dec 2017.
 - [13] S. Gao, F. Hayes, S. Croke, C. Messenger, and J. Veitch. Quantum algorithm for gravitational-wave matched filtering. *Physical Review Research*, 4(2):023006, Apr. 2022.
 - [14] I. M. Georgescu, S. Ashhab, and F. Nori. Quantum simulation. *Rev. Mod. Phys.*, 86:153–185, Mar 2014.
 - [15] V. Giovannetti, S. Lloyd, and L. Maccone. Quantum random access memory. *Physical Review Letters*, 100(16), apr 2008.

- [16] L. K. Grover. A fast quantum mechanical algorithm for database search, 1996.
- [17] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.
- [18] N. R. Rypkema. A straightforward derivation of the matched filter, 2016.
- [19] S. Sadana. Grover’s search algorithm for n qubits with optimal number of iterations, 2020.
- [20] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, oct 1997.
- [21] K. Srinivasan, S. Satyajit, B. K. Behera, and P. K. Panigrahi. Efficient quantum algorithm for solving travelling salesman problem: An ibm quantum experience, 2018.
- [22] A. Steane. Quantum computing. *Reports on Progress in Physics*, 61(2):117–173, feb 1998.
- [23] W. G. Unruh. Maintaining coherence in quantum computers. *Physical Review A*, 51(2):992–997, feb 1995.
- [24] D. R. Vemula, D. Konar, S. Satheesan, S. M. Kalidasu, and A. Cangi. A scalable 5,6-qubit grover’s quantum search algorithm, 2022.
- [25] W. K. Wootters and W. H. Zurek. A single quantum cannot be cloned. , 299(5886):802–803, Oct. 1982.

ProQuest Number: 30490834

INFORMATION TO ALL USERS

The quality and completeness of this reproduction is dependent on the quality and completeness of the copy made available to ProQuest.



Distributed by ProQuest LLC (2023).

Copyright of the Dissertation is held by the Author unless otherwise noted.

This work may be used in accordance with the terms of the Creative Commons license or other rights statement, as indicated in the copyright statement or in the metadata associated with this work. Unless otherwise specified in the copyright statement or the metadata, all rights are reserved by the copyright holder.

This work is protected against unauthorized copying under Title 17,
United States Code and other applicable copyright laws.

Microform Edition where available © ProQuest LLC. No reproduction or digitization of the Microform Edition is authorized without permission of ProQuest LLC.

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346 USA