مدينة زويل للعلوم والتكنولوجيا
Zewail City of Science and Technology

CIE 470 – Into to Quantum Comp. & Quantum Info.

Lecturer – Dr. Salem Hegazy

TA – Mr. Hesham Amer

Fall 2020 || Course Project

Grover's Algorithm for 3-Qubit System

Final Draft

Name: Muhammad Farouk El-Saeed　　ID: 201700188

Major: Physics of Earth and Universe　　Year: 4

Contents

1. Introduction

The problem of searching for a desired input among N inputs can be classically solved via doing O(N) operations, on average N/2, and it even can take very long time to be solved in some situations of large enough N. However, this problem can be solved via applying $O(\sqrt{N})$ operations by the means of a quantum mechanical search algorithm, known as "*Grover's Algorithm*". In this project, I will introduce a methodology to search for a single value in an N-Qubit system –taking a 3-Qubit system as an illustration—, via building a quantum circuit step-by-step. First, some basic definitions are stated in order to be used in building the circuit. Second, a comprehensive methodology to how exactly such circuit can be built and a full code will be used. Finally, there is a discussion upon the number of iterations and the efficiency of such a quantum mechanical circuit.

2. Definitions

2.1 Some Gates

2.1.1 Hadamard Gate: represents a π-rotation around the $\pi/8$-axis in the complex plane, and a reflection by $\pi/8$ in the real plane. It has the following matrix representation:

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

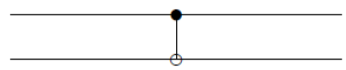
2.1.2 X Gate/ NOT Gate: flips the bit from $|0\rangle$ to $|1\rangle$ and vice versa. It has the following matrix representation:

$$NOT = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

2.1.3 Z Gate: flips the phase from $|+\rangle$ to $|-\rangle$ i.e., it is a NOT Gate upon actin in the plus-minus basis; $|+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}$, $|-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$. It has the following matrix representation:

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

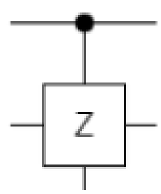
2.1.4 CNOT Gate: a 2-Qubit gate that takes the first bit a *control bit* and the second bit as a *target bit* and acts such that it flips the target if and only if the control is to |1⟩. It is drawn as follows:



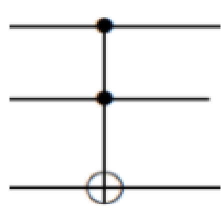
and has the following matrix representation:

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

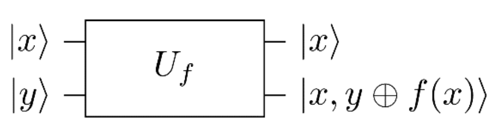
2.1.5 Control-Z Gate similar to the CNOT gate, takes a control bit and a target bit but it flips the sign the sign of the target bit if the control bit is |1⟩. It is drawn as follows:



2.1.6 CCNOT Gate/ Toffoli Gate: similar to the CNOT gate, but a 3-Qubit gate that take the first and the second bits as control bits and the third bit as a target. It is drawn as follows:



2.2 The Quantum Oracle is a unitary operator defined by its action on the computational basis such that it flips the sign of the desired state i.e., $f(x) = 1$ and do nothing if otherwise.

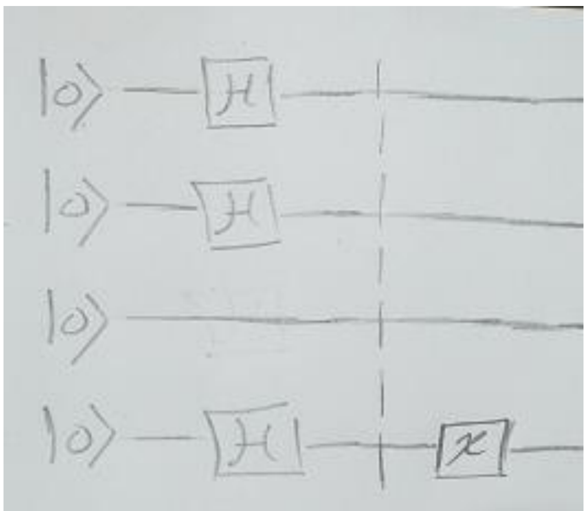


3. Methodology & Code Description

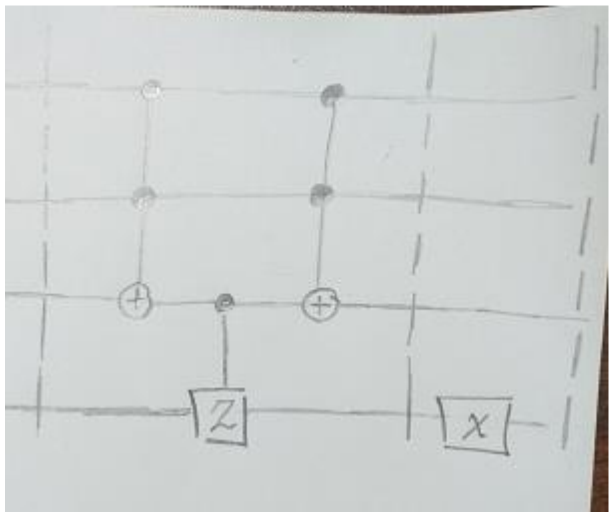
3.1 What to do?

The ultimate goal is to search for a single value in the database which contain 2^3 values because the circuit runs on 3-Qubit system.

1) First, we need a Quantum Oracle such that its function returns 1 in case of a solution, and 0 in case there is no solution. Therefore, if we let ω to be a solution then $f(x) = 1$ if and only if $x = \omega$ and 0 otherwise. Hence, after preparing $|0\rangle \otimes |0\rangle \otimes |0\rangle$ we start to construct the Oracle as follows:
    - We insert, in the N-1 position, one more $|0\rangle$, called a "*helping Qubit''*, because it will serve as a target bit for the CCNOT gate and as a control bit for the Control-Z gate. For N-Qubit system, we would insert $\mathrm{N}-2$ states to serve as target bits for the CCNOT gate.
    - We apply Hadamard gates on our prepared bits to convert them all into the plus states in order to generate the 8 equally-probable superposition states.
    - At this point, we should think of a desired single value to search for. For example, if we choose to search for $|110\rangle$. Then, this step would be applying X gate on the last state to convert 0 to 1 and the reason is to be known explicitly in the following steps.

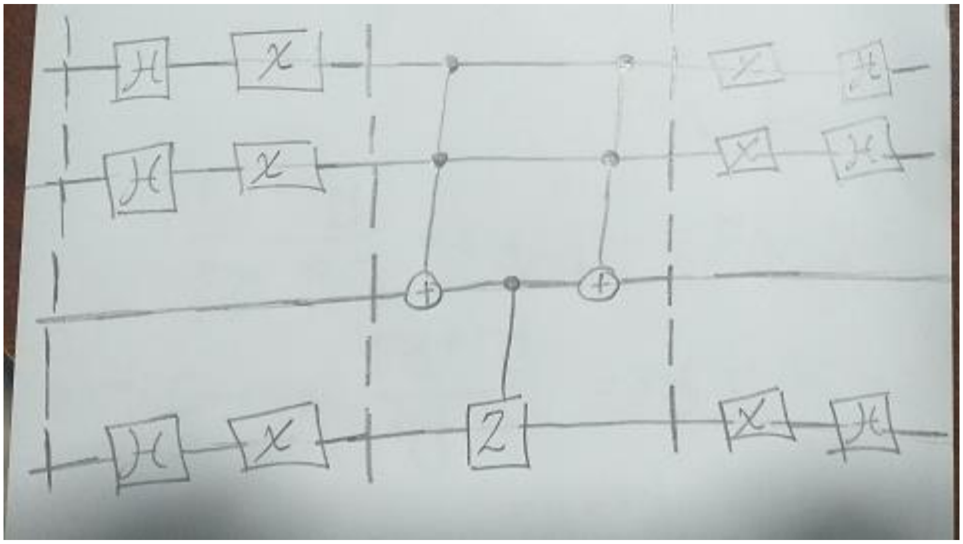


- This based on our choice, we apply the CCNOT gate such that it takes the 1^{st} and the 2^{nd} states as control states and the 3^{rd} state –the helping Qubit–, as its target state. This is because we need a gate that can control two states and the CNOT gate cannot handle such situation. Then, we apply the Control-Z gate such that it takes the 3^{rd} state as its control and the last state as its target. If the first two states are anything other than 1 and 1, the Control-Z gate will not operate. Then, we apply the CCNOT gate again, and we finish the Oracle via applying the X Gate again due to conserving the unitarity.



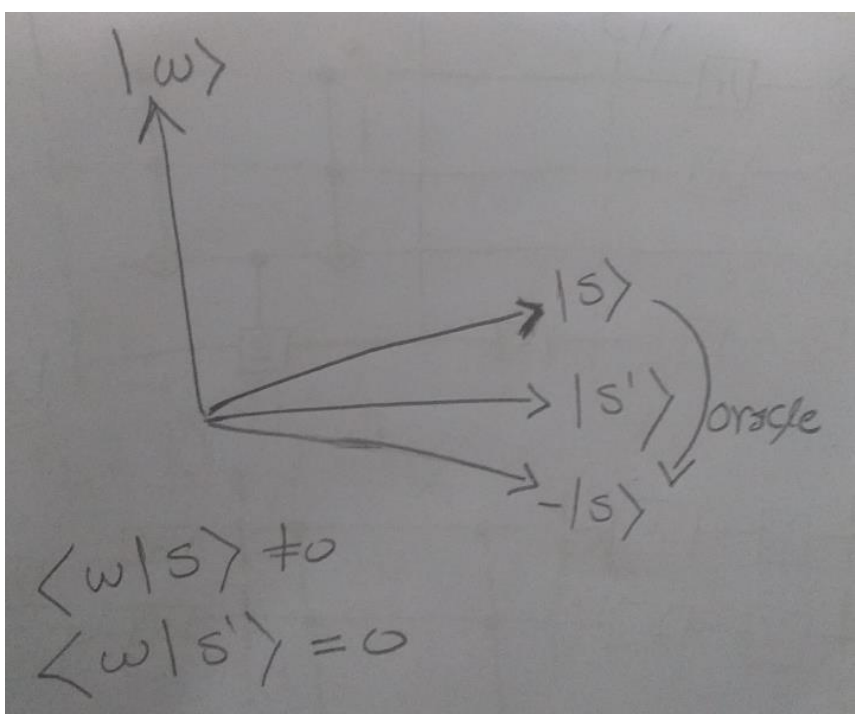
2) Second, we need a Reflection Operator putting a conditional phase shift on the state $|0\rangle$. This is because we want the previously flipped sign of $|0\rangle$ to be restored and the amplitude to be amplified.

- We apply the H gates on the 1^{st}, 2^{nd}, and the 4^{th} bits to restore basis of $|0\rangle$ and $|1\rangle$.
- We apply X gates on the 1^{st}, 2^{nd}, and the 4^{th} states in order to change all the resulting $|0\rangle$ states into $|1\rangle$ states and vice versa which return the desired state up to negative sign. Then, we apply the CCNOT, Control-Z, and CCNOT gates again, respectively, because we want to absorb back the negative sign the Oracle marked the desired state by. Eventually, we apply again X gates and H gates, respectively, on the 1^{st}, 2^{nd}, and the 4^{th} bits because this is a unitary operation after all.

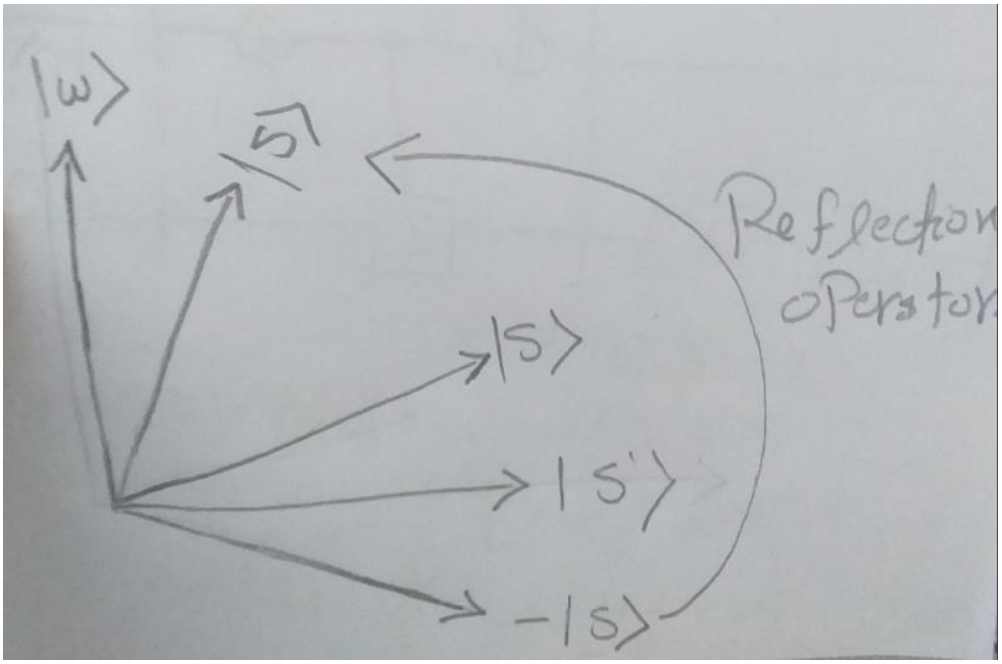


3) We apply our measurement.

Comment: Basically, we seek a desired state, $|\omega\rangle$, and applying the H returns superposition states, $|S\rangle$, which are not orthogonal to our desired state. Yet, within the span of our desired state and one of the superposition states, there always exist another state, $|S'\rangle$, that is orthogonal to the desired state i.e., $\langle S'|\omega\rangle = 0$. The Oracle job is to reflect that superposition state about the orthogonal complement. This can be visualized as the following sketch:

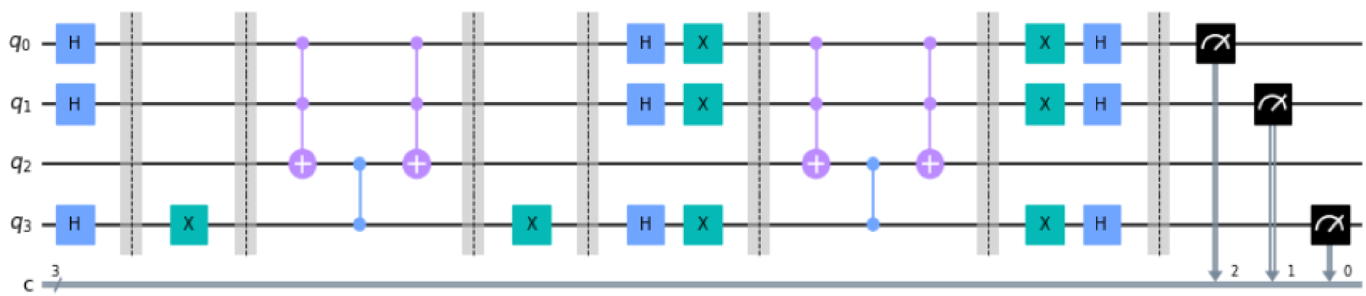


Now comes the Reflection Operator into the game; the reflection operator reflects the marked state, $-|S\rangle$, about the superposition state, $|S\rangle$, again which makes it closer to our desired state, and upon repeating the circuit many times, we get so close to the desired state. This is why it is called "*amplitude amplification*".



3.2 Code Description

We simply take the above detailed procedure and convert into a coding language and we would get a result like the following.



The above circuit using Qiskit is designed upon inserting 3-Qubit state i.e., $|110\rangle$ in the code. Yet, the code itself is designed to search for a given single value within N-Qubit system.

4. Discussion

This discussion is regarding the iterations and the efficiency of our circuit. Clearly, this quantum search algorithm is much more efficient than the classical method because, for example, in the case of 3-Qubits we get $|110\rangle$ with probability around 0.945 instead of a 0.125 classical probability. A more satisfying example is when we get $|1100110\rangle$, using the present algorithm with 10^5 shots, with probability around 0.996 instead of 0.0078125 classically. Evidently, it takes only very few seconds! Yet, using the Toffoli gate, when designing N-Qubit system, amounts to use $2N - 2$ Qubits in the circuit which seems to add more load to our code. However, the number of iterations for a single value within the database, which is $\frac{\pi}{4}\sqrt{2^N}$, does not change because there is no contribution from any helping Qubits regardless its number.

REFERENCES

- M. A. Nielsen, I. L. Chuang, “Quantum Search Algorithms, “in Quantum Computation and Quantum Information, 1th ed., New York: Cambridge, 2010, ch. 6, pp. 248-276.