Ahmedabad University

Project

Topic: Quantum Counting

Course: MAT 485 Introduction to Quantum Computing

Submitted to Dr Alok Shukla

Date of Submission: 29th April 2022

Name	Roll No
Hirmay Sandesara	AU1940265
Freya Shah	AU2120184
Aum Trivedi	AU2020128
Kirtan Kalaria	AUL2020005

Table of Contents

Introduction

Quantum Counting is an algorithm which brings together the Grover's Algorithm and the Quantum Phase estimation algorithm. It estimates the phase while giving us an approximate estimate of the count of solutions (essentially the number of marked items) for a specified search problem. It is a practical algorithm since it gives the count of solutions, but a particular case also tells us whether any solution exists (problem of quantum existence).

We use Grover's Algorithm to estimate the number of marked bits. We get the eigenvalues of Grover's iterator as, $e^{\pm i\theta}$ and it rotates by an angle every time Grover's iterator is applied. After k iterations, we can obtain the phase by measuring the superposition state to find the value of rotation which gives us the phase. Then we estimate M by using the Quantum Phase Estimation and then using the formula: $M = N\, sin^2(\theta/2)$. This algorithm to measure both counts M and phase is known as Quantum Counting. Here, we will have a detailed look at the algorithm.

Problem Description

The problem is that for a given problem, it might be necessary to efficiently find and count the number of solution that exists for that particular problem. At the same time, we might come across a situation where there might not exist any solution. We might need to find (and count) the number of solutions (which could be thought of as marked items) present and stored in a phase and then extracted to compute the number of solutions present in the problem.

Mathematics of the Algorithm

- Handwritten notes of the mathematical structure (and calculation) of the algorithm

Quantum Counting.

Problem: $N = 2^n$, $f: \{0,1\}^n \longrightarrow \{0,1\}$. $M \subset \{0,1\}^n$

Here M represents the number of marked elements.

such that $f(x) = \begin{cases} 1 & x \in M \\ 0 & x \notin M \end{cases}$

Task: Count $f^{-1}(1) = M \Rightarrow$ get $|M|$

Classical: $\Omega(N)$ (search everything)

Recall: Grover's algorithm.

$$|\psi\rangle = \left(\frac{1}{\sqrt{M}} \sum_{x \in M} |x\rangle\right) \sqrt{\frac{M}{N}} + \sqrt{\frac{N-M}{N}} \left(\frac{1}{\sqrt{N-M}} \sum_{x \notin M} |x\rangle\right)$$

$$= |\beta\rangle \sqrt{\frac{M}{N}} + |\alpha\rangle \sqrt{\frac{N-M}{N}}$$

$$= \cos\frac{\theta}{2}|\alpha\rangle + \sin\frac{\theta}{2}|\beta\rangle$$

$$\sin\frac{\theta}{2} \approx \sqrt{\frac{M}{N}} \rightarrow \frac{\theta}{2} \approx \sqrt{\frac{M}{N}} \rightarrow \theta \approx 2\sqrt{\frac{M}{N}}$$

Grover's operator:

$$G = 2|\psi\rangle\langle\psi| - I$$

$$U|\alpha\rangle = 2|\psi\rangle\langle\psi|\alpha\rangle - |\alpha\rangle$$

$$= 2|\psi\rangle \cos\frac{\theta}{2} - |\alpha\rangle$$

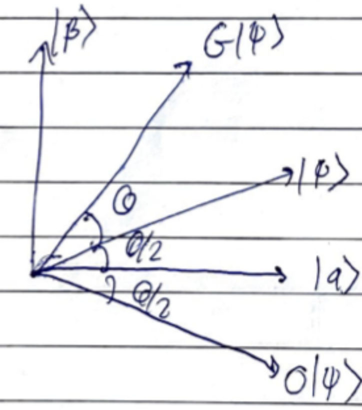
$$= \left(2\cos^2\frac{\theta}{2} - 1\right)|\alpha\rangle + 2\sin\frac{\theta}{2}\cos\frac{\theta}{2}|\beta\rangle$$

$$U|\beta\rangle = 2|\psi\rangle\langle\psi|\beta\rangle - |\beta\rangle = 2|\psi\rangle\sin\frac{\theta}{2} - |\beta\rangle$$

$$= 2\cos\frac{\theta}{2}\sin\frac{\theta}{2}|\alpha\rangle + \left(2\sin^2\frac{\theta}{2} - 1\right)|\beta\rangle$$

$$U = \begin{pmatrix} \cos\theta & \sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \quad O = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$G = UO = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$



$$k\theta \approx \frac{\pi}{2}$$

$$k \approx \frac{\pi}{2\theta}$$

$$k \approx \frac{\pi}{4}\sqrt{\frac{N}{M}}$$

G is a unitary operator

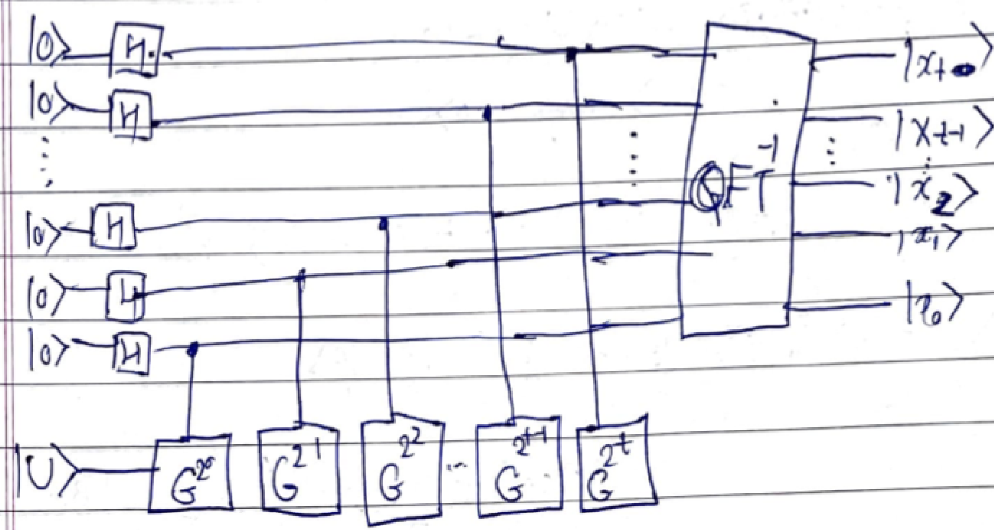
$\therefore G|\psi\rangle = e^{\pm i\theta}|\psi\rangle$ ($e^{\pm i\theta}$ are eigenvalues of G)

Using Quantum Phase estimation we get θ.

Once we get θ, we get M as

$$\frac{\theta}{2} \approx \sqrt{\frac{M}{N}}$$

In QPE, we use unitary operator as grover's operator.



Coding logic/thought structure

First, we import all the necessary packages like numpy, qiskit, pylatexenc, and matplotlib. Next, we proceed to make the individual components of the circuit.

We start by making the Controlled Grover Iterator. For this, we convert the circuit for a Grover iterator to a gate using .to_gate() and then apply the control part to it using .control().

Then we make the Inverse QFT part. For this, we first make a circuit for QFT and convert it to a gate, like we saw with Grover iterator. Then, using .inverse(), we invert the QFT gate, thereby getting our Inverse QFT gate.

Now that we have modularly designed the components, we can use them to make the circuit for Quantum Counting. We will measure the counting bits.

After this, we simulate the circuit using Aer and get counts of different measurements. We perform a little processing on the measurement to obtain the number of solutions.

We repeat this activity for different number of counting qubits. (t = 2, 4, 8)

Applications

This algorithm has many applications :

1. Finding the number of iterations for Grover’s algorithm.
2. Speeding up NP complete problems (one example is finding the existence of hamilton cycles).
3. Checking whether or not a function has a solution (Quantum existence problem).
4. Reducing search spaces.

References

IBM Quantum. (2022, April 28). *Quantum Counting*. Qiskit.

https://qiskit.org/textbook/ch-algorithms/quantum-counting.html

Wie, C. R. (2019). Simpler quantum counting. *Quantum Information and Computation*, *19*(11 &

12), 967–983. https://doi.org/10.26421/qic19.11-12-5