Investigation on Quantum Search Algorithm and its Possible Applications

Zhan Linying
Physics, Faculty of Science, the University of Hong Kong

Summer Research Fellowship (SRF) 2020 for Science Students
Poster No.: D14
Name: Zhan Linying
University No.: 3035533680
Student's Major: Physics

Introduction

Living in this Information Age, with a wealth of information stored in the database and the demand for better retrieval effect, developing an efficient search algorithm becomes increasingly vital in various fields. While with a properly sorted data structure, classical computation provides multiple search algorithms with quite satisfactory efficiency, it could do nothing more than a linear search for an unstructured database. Fortunately, the development of quantum computation gave us some novel insights into this problem. Here, we investigate the quantum search algorithm proposed by Lov K. Grover (1996), discuss its strengths and limitations, and analyze its possible applications.

Grover's Algorithm for N = 4

This case deals with the simplified problem of retrieving one marked item out of four. Since four items can be represented by two qubits, we prepare them in the state |x> = |00> supplemented by an ancillary qubit in the state |y> = |1>. The Grover's algorithm in this case is represented by the quantum circuit shown in Fig 1.

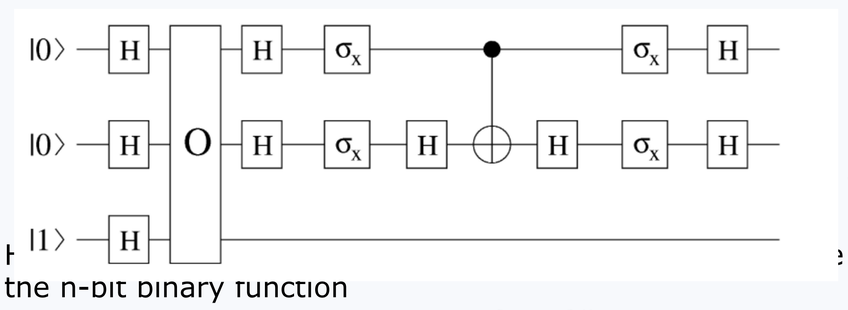

Fig 1: A quantum circuit implementing Grover's algorithm for N = 4 items.

H … e query, which computes the n-bit binary function

$$f:\{0,\ 1\}^n \rightarrow \{0,\ 1\},$$

defined as

$$f(x) = \begin{cases} 1 & if\ x = x_0, \\ 0 & otherwise. \end{cases}$$

This operates on the quantum-computer wave function by means of

$$|x> |y> \rightarrow |x> |y \oplus f(x)>.$$

Then after the first two transformations (the Hadamard gates and the oracle query), the wave function becomes

$$|00> |1> \rightarrow_{Hadamard\ gates} \frac{1}{2}(|00> + |01> + |10> + |11>) \frac{1}{\sqrt{2}}(|0> - |1>) \rightarrow_{oracle\ query} \frac{1}{2}((-)^{f(|00>)}|00> + (-)^{f(|01>)}|01> + (-)^{f(|10>)}|10> + (-)^{f(|11>)}|11>) \frac{1}{\sqrt{2}}(|0> - |1>).$$

Note that the minus sign on the second register due to the oracle query is transferred into the first register. Now, the coefficient in front of the marked state differs from the others by a minus sign. However, since the phase difference would not be recognized by measurement, it has to be changed into amplitude difference. One way of achieving it is to use the unitary transformation

$$D_{ij} = -\delta_{ij} + \frac{2}{2^n} \text{ with n = 2.}$$

Using the matrix representation of $D_{n=2} = \frac{1}{2}\begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}$, one can check that it operates on the first register to give the marked state $|x_0>$ only. Now, a standard measurement completes the search by giving the target item with certainty.

From the circuit, it can be seen that D can be decomposed into universal quantum gates

$$D_{n=2} = H^{\otimes 2} D'_{n=2} H^{\otimes 2} \text{ where } D'_{n=2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} = \sigma_x^{\otimes 2} (I \otimes H)\ CNOT\ (I \otimes H)\ \sigma_x^{\otimes 2}$$

This makes it possible to implement this algorithm practically.

Grover's Algorithm for N = 2^n

We now extrapolate the previous case to the search from a genetic size of $N = 2^n$ items. Similarly, the original state of the wave function is prepared as |x> |y> = |00…0> |1> where |x> consists of n qubits. Successive use of Hadamard gates and oracle query at the beginning transforms the wave function by

$$|00 \dots 0> |1> \rightarrow_{Hadamard\ gates} \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x> \frac{1}{\sqrt{2}}(|1> - |0>)$$

$$\rightarrow_{oracle\ query} \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-)^{f(x)}|x> \frac{1}{\sqrt{2}}(|1> - |0>).$$

The first register is very close to the equal superposition state with negative coefficient for $|x_0>$ only. This time, to transform the phase difference into amplitude difference, single application of D is not enough.

We need to accumulate the amplitude difference by iterating the transformation G = DO, known as Grover iteration, where O represents the oracle function. With a proper number of times of iteration, the resulting wave function becomes very close to $|x_0>$ and can then be measured to be x_0 with very high probability.

Complexity Analysis

In this part, we analyze the complexity of the general case of Grover's algorithm by interpreting it geometrically. We assess the complexity of search algorithm by the number of times the oracle is called, which directly corresponds to the number of iteration steps.

Firstly, let us visualize the transformations used in Grover's algorithm.

Oracle function:

$$O = I - 2|x_0 >< x_0| = R_{|x_0>}.$$

Unitary transformation D:

$$D = H^{\otimes n} D' H^{\otimes n} \text{ where } D' = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 0 & \cdots & 0 \\ 0 & 0 & -1 & & 0 \\ & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -1 \end{bmatrix} = -I + 2|0 >< 0| \rightarrow D$$

$$= H^{\otimes n}(-I + 2|0 >< 0|) H^{\otimes n} = -I + 2\left(H^{\otimes n}|0>\right)\left(<0|H^{\otimes n}\right)$$

$$= -I + |S >< S| = -R_{|S>} \ where\ |S> = H^{\otimes n}|0> (equal\ superposition\ state).$$

Hence the Grover iteration $G = DO = -R_{|S>} R_{|x_0>}$ corresponds to two reflections, one about the hyperplane perpendicular to $|x_0>$ and another about the plane of |S>.

In the 2D plane spanned by $|x_0>$ and |S>, the above two reflections rotate the original state by an angle 2θ if θ denotes the angle between $|x_0>$ and |S>, as shown in Fig 2.

Then we have

$$|\varphi_0> \equiv |S> = sin\theta|x_0> + cos\theta|x_0^{\perp}>$$

$$\rightarrow |\varphi_j> = \sin((2j+1)\theta)|x_0> + \cos((2j+1)\theta)|x_0^{\perp}>$$

$$\rightarrow (2k+1)\theta \approx \frac{\pi}{2} \rightarrow k = round\left(\frac{\pi}{4\theta} - \frac{1}{2}\right) \text{ such that } \sin((2k+1)\theta) \approx 1$$

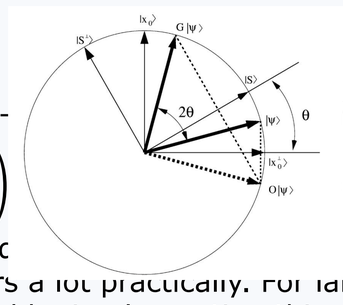

Fig 2: 2D plane spanned by $|x_0>$ and |S>

$= O(\sqrt{N})$.

It is better than the linear complexity obtained … Although the improvement is not exponential, it still matters a lot practically. For large number of N, considerable amount of time could be saved by implementing this quantum search algorithm.

Limitations

1. Does not give the correct result with certainty.

For Grover's algorithm with N ≠ 4, there is no guarantee that the final measurement gives the correct answer. However, the failure probability $p(x \neq x_0) = cos^2[(2k+1)\theta] = cos^2\left(\frac{\pi}{2} + O(\theta)\right) = cos^2\left(\frac{\pi}{2} + O\left(\frac{1}{\sqrt{N}}\right)\right) = O(\frac{1}{N})$ drops as the number of items increases. In real life, with a large enough database, the failure probability could be sufficiently small. By using the oracle to check its correctness, we can obtain the right result quite efficiently. Additionally, for databases with small dimensions, the original Grover's algorithm can be modified to provide zero failure rate. One method proposed by Long, Gui-Lu (2001) is to achieve transformation from phase difference to amplitude difference by repeated implementations of phase rotations with certain angles that depends on the size of the database.

2. Index representation could take long time.

Since Grover's algorithm makes use of the oracle function which checks the indices of the items, it is necessary to attach an index to each item in the database. Such representation could be time-consuming since going through the unstructured database requires O(N) times of operations. The good news is that we only need to do it once for an unchanged database and modify it later for any small changes.

3. Does not work well for databases distributed throughout the network (Viamontes et al. 2005, 65).

Such databases are generally too large to be stored in one computer, and multiple storage system allows them to be processed in parallel and then merged to form the final search result. However, while classical search algorithm is capable of being searched separately and then copied and added together, Grover's algorithm requires superposition of indices that only works for localized databases.

Possible Applications

There is no denying the fact that nowadays, the use of search algorithm is pervasive, from the searching of commodities in Taobao to the processing of financial transactions in banks all over the world. These are all the possible areas that quantum search algorithm could apply. However, due to the limitations stated above, it lacks the ability of searching for extremely large databases distributed throughout the network. So, one way of applying it is to use it as a coprocessor to work with a classical computer. For example, a quantum search algorithm may be used to search for a localized database and send the result to the classical computer to do the aggregation part. Additionally, for a moderate database that can be stored in a single computer, it works better than a linear search algorithm.

References

Grover, Lov K. "A fast quantum mechanical algorithm for database search." In Proceedings of the twenty-eighth annual ACM symposium on Theory of computing, pp. 212-219. 1996.

Long, Gui-Lu. "Grover algorithm with zero theoretical failure rate." Physical Review A 64, no. 2 (2001): 022307.

Viamontes, George F., Igor L. Markov, and John P. Hayes. "Is quantum search practical?" Computing in science & engineering 7, no. 3 (2005): 62-70.