Quantum Circuit Based on Grover Algorithm to Solve Hamiltonian Cycle Problem

Jehn-Ruey Jiang
Department of Computer Science and Information Engineering
Naitonal Central University
Taoyuan City, Taiwan
jrjiang@csie.ncu.edu.tw

2022 IEEE 4th Eurasia Conference on IOT, Communication and Engineering (ECICE) | 978-1-6654-8208-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ECICE55674.2022.10042919

***Abstract*—We propose the concepts of the explicit oracle and the implicit oracle for realizing quantum algorithms. Then, the quantum circuit of the well-known Grover algorithm is constructed with the explicit oracle to solve the Hamiltonian cycle problem for the complete graph. The quantum circuit has a quadratic speedup over the classical unstructured search algorithm for solving the same problem. The IBM quantum computer simulator is used to run the quantum circuit to validate that it can indeed derive the Hamiltonian cycle of the complete graph.**

Keywords—Grover algorithm, Hamiltonian cycle, noisy intermediate-scale quantum, oracle, quantum computer, quantum circuit

I. Introduction

Smart manufacturing has been attracting much research attention recently. Combinatorial optimization problems need to be solved in some smart manufacturing applications. For example, smart factory applications may need to solve the job shop scheduling problem, logistics applications may need to solve the vehicle routing problem, and operations research applications may need to solve the Hamiltonian cycle problem. Nevertheless, since combinatorial optimization problems usually require huge computing power for obtaining instant solutions, studies advocate using quantum computers to solve the problems.

Traditional (or classical) computers perform computation based on bits, whereas quantum computers perform computation based on qubits. Because qubits have superposition and entanglement states, quantum computers may solve problems fast that classical computers can never solve in a feasible time, achieving quantum supremacy [1]. We are now in the noisy intermediate-scale quantum (NISQ) era [2], in which quantum computers have only a moderate number of error-prone qubits. Hence, quantum computers may or may not solve a problem faster than classical computers. However, quantum computers are likely to show their advantages in computational speed over classical computers soon. It is then worthwhile to devote ourselves to investigating and designing quantum algorithms solving complex combinatorial optimization problems with quantum computers.

The concepts of the explicit oracle and the implicit oracle are proposed for realizing quantum algorithms. It then shows how to construct the quantum circuit of the well-known Grover algorithm [3] with the explicit oracle to solve the Hamiltonian cycle problem (HCP) for the complete graph (or clique) with n edges. The classical algorithm may take $O(2^n)$ time complexity to solve the problem, whereas the quantum circuit based on the Grover algorithm takes $O(\sqrt{2^n})$ time complexity to solve the same problem, achieving a quadratic speedup. The quantum circuit is realized and run by an IBM quantum computer simulator to verify that it can solve the HCP with the time complexity of $O(\sqrt{2^n})$.

The rest of this paper is organized as follows. Section II introduces some preliminaries. The concepts of the explicit oracle and the implicit oracle are described in Section III. Section IV shows the quantum circuit based on the Grover algorithm using the explicit oracle to solve the HCP for the complete graph. Finally, concluding remarks are drawn in Section V.

II. Preliminaries

A. Grover Algorithm

The Grover algorithm (or Grover's algorithm) [3] is a quantum algorithm proposed by Grover in 1996 to solve the unstructured search problem with a high probability. Specifically, it finds the unique input with a high probability to an oracle or a black box function that produces a specific output. It is also known as the quantum unstructured search algorithm, as the associated overall input has no predefined pattern, nor is it arranged or organized in a predefined way.

Consider an oracle or a black box function f: $\{0,1\}^n \rightarrow \{0,1\}$. The oracle has $N=2^n$ input instances and there exists a unique solution input instance x^* such that $f(x^*)=1$. The oracle f is defined precisely as follows.

$$f(x) = \begin{cases} 1 & \text{if } x = x^* \\ 0 & \text{if } x \neq x^* \end{cases} \quad (1)$$

The unstructured search problem is to find the solution input instance x^*.

Since the N input instances of the oracle f are unstructured, a classical algorithm needs to call the oracle with every input instance to find the solution input instance x^*. A classical algorithm needs to call the oracle N (resp., $\frac{N+1}{2}$) times to spot x^* in the worst case (resp, the average case). Therefore, the classical algorithm to solve the unstructured search problem has a time complexity of $O(N)$. The Grover algorithm solves the unstructured search problem with high a probability in the time complexity of $O(\sqrt{N})$. Therefore, the Grover algorithm has a quadratic speedup over its classical counterpart in time complexity.

Charles et al. showed the Grover algorithm is asymptotically optimal by proving that any quantum algorithm to solve the unstructured search problem needs to call the oracle $\Omega(\sqrt{N})$ times [4]. The Grover algorithm has many applications. It can be extended and applied for finding the minimum [5], maximum [6], mean, median [7,8], number of solution input instances [9], and collision pairs [10] of a given set of unstructured data.

Figure 1 shows the quantum circuit of the Grover algorithm using the quantum phase oracle U_f [11]. Let $|x^*\rangle$

978-1-6654-8208-0/22/$31.00 ©2022 IEEE

be the unique solution input instance such that the phase of $|x^*\rangle$ is inversed. Specifically, U_f is defined as follows.

$$U_f|x\rangle = \begin{cases} -|x\rangle & \text{if } |x\rangle = |x^*\rangle \\ |x\rangle & \text{if } |x\rangle \neq |x^*\rangle \end{cases} \quad (2)$$

The following steps describe the details of the quantum circuit of the Grover algorithm:

Step 1 prepares n working qubits in the state $|0\rangle$, i.e., the state $|0\rangle^{\otimes n}$.

Step 2 lets all qubits pass through the Hadamard (H) gate or operation so that the qubits are in a uniform superposition state, as shown in the following equation.

$$H^{\otimes n}|0\rangle^{\otimes n} = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x\rangle, \quad (3)$$

where N=2^n.

Step 3 lets the qubits in the superposition state pass through the phase oracle to perform phase inversion for the solution input instance $|x^*\rangle$.

Step 4 lets all qubits go through the diffusion operation U_s, the inversion-around-mean operation is defined.

$$U_s = H^{\otimes n}(2|0^n\rangle\langle 0^n| - I)H^{\otimes n}, \quad (4)$$

where I is the identity matrix, and $|0^n\rangle\langle 0^n|$ stands for the outer product of an n×1 column vector $(1,0\ldots,0)^T$ and a 1×n row vector $(1,0,\ldots 0)$.

As shown in Fig. 2, the diffusion operation causes the probability amplitudes of all qubits to invert around the mean μ of all amplitudes [11]. Thus, the original positive amplitude just decreases slightly. However, the original negative amplitude becomes a very large positive amplitude.

Note that $2|0^n\rangle\langle 0^n| - I = -1X^{\otimes n}[MCZ]X^{\otimes n}$, where X stands for the X gate or operation and [MCZ] stands for the multi-controlled Z gate or operation.

Step 5 repeats steps 3 and 4 for $\left\lfloor\frac{\pi}{4}\sqrt{N}\right\rfloor$times.

Note that step 3 performs phase inversion for the solution input instance $|x^*\rangle$ and step 4 performs the inversion-around-mean operation for all input instances. Thus, the probability amplitude of the solution input instance $|x^*\rangle$ become larger, whereas other input instances' amplitudes remain relatively small. Also note that Chen et al. [12] showed that when the number of solution input instances is M, $M \geq 1$, then steps 3 and 4 are repeated for $\left\lfloor\frac{\pi}{4}\sqrt{\frac{N}{M}}\right\rfloor$ times to find all the M solution input instances with high probability.

Step 6 measures all qubits. The qubit state with the highest probability corresponds to the solution input instance.

Based on the quantum circuit of the Grover algorithm, it easily derives that the time complexity of the Grover algorithm is $O(\sqrt{N})$. This is because steps 3 and 4 are repeated for $O(\sqrt{N})$ times.

Figure 3 shows the quantum circuit of the Grover algorithm of 2 input qubits with the solution input instance being $|10\rangle$. The part between the first barrier and the second barrier in Fig. 3 is the circuit of the oracle, which is realized by adding the X gate on qubit 0, adding the controlled-Z (CZ) gate on qubits 0 and 1, and adding the X gate on qubit 0. The phase kickback effect of the CZ gate inverts the phase of the input instance $|10\rangle$. The part between the second barrier and the third barrier in Fig. 3 is the circuit of the diffusion operation, which is realized by adding the H gate on qubits 0 and 1, adding the X gate on qubits 0 and 1, adding the CZ gate on qubits 0 and 1 (if there are three qubits or more, then the MCZ gate is added on all the qubits instead), adding the X gate on qubits 0 and 1, and finally adding the H gate on qubits 0 and 1. There are 4 input instances in total and only one solution input instance, so the oracle and the diffusion operation repeat only once ($\left\lfloor\frac{\pi}{4}\sqrt{N}\right\rfloor = 1$).

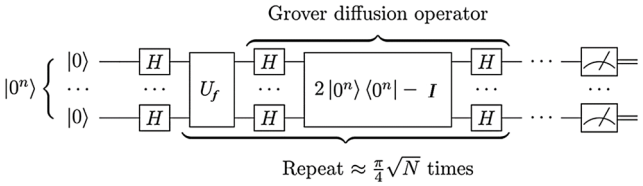

Fig. 1. Quantum circuit of the Grover algorithm [11].

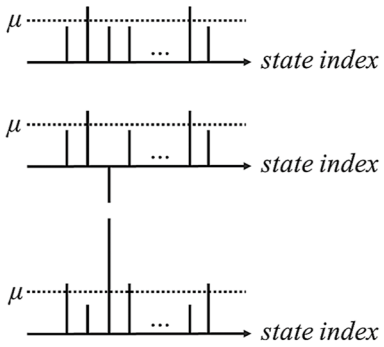

Fig. 2. Illustration of the diffusion operation [11].

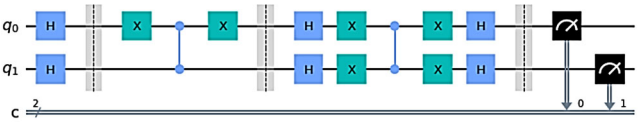

Fig. 3. Qantum circuit of the Grover algorithm of 2 input qubits with the solution input instance being $|10\rangle$.

B. Hamiltonian Cycle Problem

Given an undirected graph G=(V, E) with vertex set V of n vertices and edge set E, the Hamiltonian cycle (HC) of graph G is a cycle containing n edges and passing through every vertex exactly once. The Hamiltonian cycle problem (HCP) is to find the HC of a given graph.

Figure 4 (a) shows a complete graph or clique of 4 vertices, which is also called a 4-clique. It is known that an n-clique has (n-1)!/2 HCs. Therefore, a 4-clique has (4-1)!/2=3 HCs. The 3 HCs are v_0-e_0-v_1-e_1-v_2-e_2-v_3-e_3, v_0-e_0-v_1-e_5-v_3-e_2-v_2-e_4, and v_0-e_4-v_2-e_1-v_1-e_5-v_3-e_3. Figure 4 (b) shows a 5-clique. It has (5-1)!/2=12 HCs. The 12 HCs are v_0-e_0-v_1-e_1-v_2-e_2-v_3-e_3-v_4-e_4, v_0-e_0-v_1-e_1-v_2-e_9-v_4-e_3-v_3-e_6, v_0-e_0-v_1-e_8-v_3-e_2-v_2-e_9-v_4-e_4, v_0-e_0-v_1-e_8-v_3-e_3-v_4-e_9-v_2-e_5, v_0-e_0-v_1-e_7-v_4-e_9-v_2-e_2-v_3-e_6, v_0-e_0-v_1-e_7-v_4-e_3-v_3-e_2-v_2-e_5, v_0-e_5-v_2-e_1-v_1-e_8-v_3-e_3-v_4-e_4, v_0-e_5-v_2-e_1-v_1-e_7-v_4-e_3-v_3-e_6, v_0-e_5-v_2-e_2-v_3-e_8-v_1-e_7-v_4-e_0, v_0-e_6-v_3-e_8-v_1-e_1-v_2-e_9-v_4-e_5, v_0-e_6-v_3-e_8-v_1-e_7-v_4-e_9-v_2-e_4, and v_0-e_6-v_3-e_3-v_4-e_8-v_1-e_1-v_2-e_5.

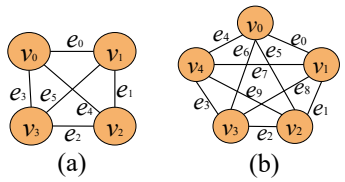

Fig. 4. (a) 4-clique complete graph of 4 vertices and 6 edges, and (b) 5-clique complete graph of 5 vertices and 10 edges.

III. Explicit Oracle and Implicit Oracle

The explicit oracle and the implicit oracle borrow the names of the explicit function and the implicit function in mathematics, but they have different meanings from their mathematical counterparts. They are defined precisely below.

An explicit oracle has solution input instances known in advance and given explicitly. For example, the oracle in the quantum circuit of the Grover algorithm of two input qubits in Fig. 3 is explicit. This is because the unique solution input instance $|10\rangle$ is known and given explicitly, and the oracle can then be easily built accordingly. For another example, to build an oracle to check whether a given edge set constitutes an HC for the 4-clique or 5-clique in Fig. 4, we easily build an explicit oracle according to the three or twelve known HCs given in advance.

On the other hand, an implicit oracle has solution input instances that are unknown in advance but given implicitly by constraints or conditions. For example, to build an oracle to check whether a given edge set constitutes an HC for the cliques in Fig. 4 without the HCs known in advance, we build an implicit oracle according to the given constraints (i.e., the edge set must contain n edges to pass through every vertex exactly once). Since the solution input instance is usually not known in advance, the implicit oracle is more practical but harder to build than the explicit oracle.

IV. Proposed Quantum Circuits

This section shows the quantum circuit construction of the explicit oracle for the Grover algorithm to solve the Hamiltonian cycle problem for the cases of the 4-clique and the 5-clique. It also shows the experiment results of executing the whole circuit to verify that the Grover algorithm along with the explicit oracle indeed can find the HCs of the 4-clique and the 5-clique.

A. 4-clique case

Figure 5 shows the quantum circuit of the Grover algorithm with the explicit oracle to solve the HCP for the 4 clique. Six input qubits $q_0,\ldots,q_5$ are present in the circuit, each of which corresponds to an edge. On the one hand, if a qubit q_i, $0 \le i \le 5$, is of the state $|1\rangle$, then edge e_i is regarded to be included in the HC. On the other hand, if a qubit q_i, $0 \le i \le 5$, is of the state $|0\rangle$, then edge e_i is regarded to be excluded from the HC. There are $N=2^6=64$ possible input instances and 3 solution input instances. Hence, the oracle and the diffusion operation should repeat $\left\lfloor \frac{\pi}{4}\sqrt{\frac{64}{3}} \right\rfloor = 3$ times. The repeating number is in the order of $\sqrt{N}$, so the whole circuit has the time complexity of $O(\sqrt{N})$.

Figure 6 shows the measurement result of running (or evolving) the quantum circuit in Fig. 5 for 1000 shots using a quantum computer simulator provided by the IBM quantum service [13]. The three input instances 001111, 110101, and 111010 have the highest occurrence probabilities, about 1/3, at 0.329, 0.341, and 0.329, respectively. They are the three solution input instances. However, the input instance 101001 has a very low occurrence of 0.001. Such an input instance is not a solution input instance, since its appearing probability is too low. Following the '1' bit of the three solution input instances from right to left (i.e., from the least significant bit to the most significant bit) leads to edge sequences, each of which corresponds to an HC.

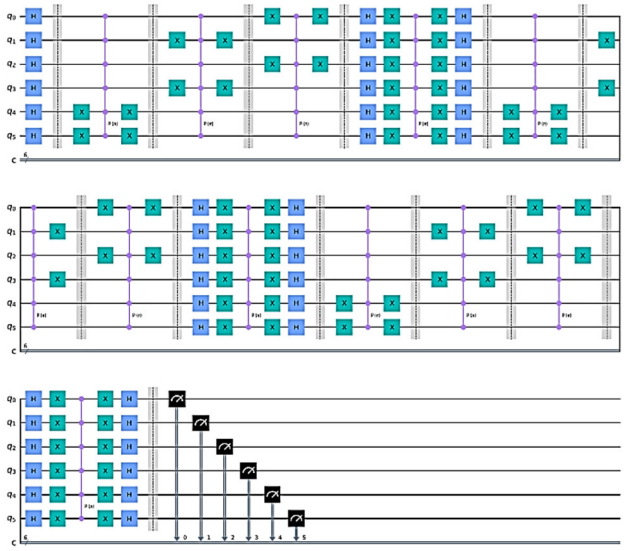

Fig. 5. Quantum circuit of the Grover algorithm with an explicit oracle to solve the HCP for 4-clique.

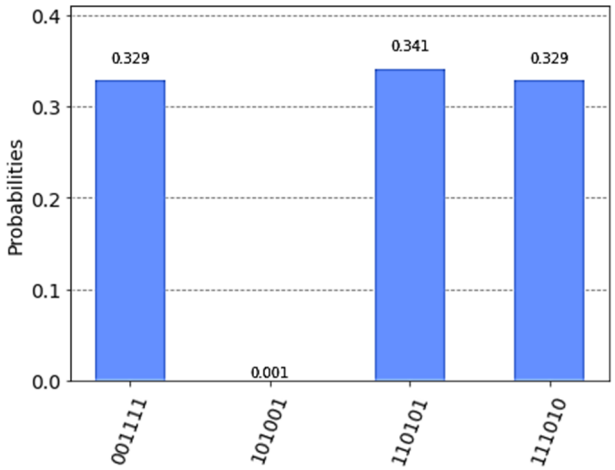

Fig. 6. Measurement results of running the quantum circuit of the Grover algorithm with an explicit oracle to solve the HCP for 4-clique.

B. 5-clique case

The quantum circuit of the Grover algorithm with the explicit oracle to solve the HCP for the 5-clique is not shown in this paper. This is because it is too complex and too lengthy to be shown. It has 10 input qubits $q_0,\ldots,q_9$, each of which corresponds to an edge. Similarly, if a qubit q_i, $0 \le i \le 9$, is of the state $|1\rangle$ (resp., $|0\rangle$), then edge e_i is regarded to be included in (resp., excluded from) the HC. There are $N=2^{10}=1024$ possible input instances and 12 solution input instances. Hence, the oracle and the diffusion operation should repeat $\left\lfloor \frac{\pi}{4}\sqrt{\frac{1024}{12}} \right\rfloor = 7$ times. The repeating number is

in the order of $\sqrt{N}$, so the whole circuit has the time complexity of O($\sqrt{N}$).

Figure 7 shows the measurement result of running (or evolving) the quantum circuit of the Grover algorithm with an explicit oracle to solve the 5-clique HCP for 1000 shots using a quantum computer simulator provided by the IBM quantum service [13]. Twelve input instances have high probabilities of occurrence. They are 0000011111, 0010101101, 0011101010, 0100111010, 0101101010, 0110100101, 1001001011, 1011000101, 1100010101, 1100101001, 1101100010, 1111010000. They are solution input instances and their associated probabilities are 0.074, 0.096, 0.072, 0.08, 0.08, 0.087, 0.082, 0.084, 0.082, 0.096, 0.079, 0.085, respectively. Three input instances 0001100110, 0001110010, and 0111001101 have low probabilities of occurrence. They are not solution input instances since the probabilities are too low.

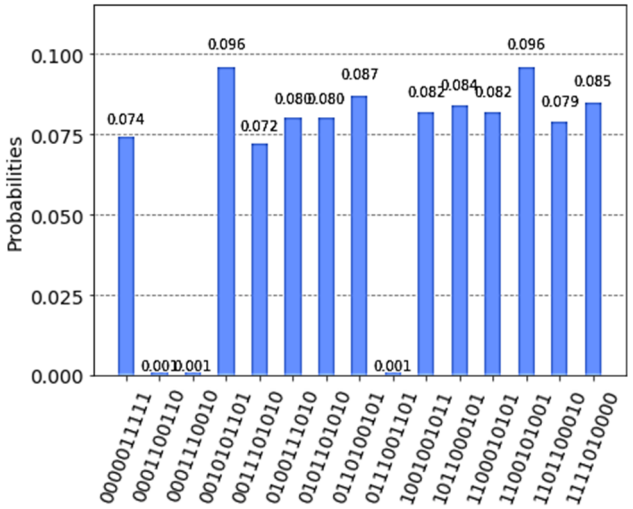

Fig. 7. Measurement results of running the quantum circuit of the Grover algorithm with an explicit oracle to solve the HCP for 5-clique.

V. Conclusion

We propose the concepts of the explicit oracle and the implicit oracle for realizing quantum algorithms. We also show how to construct the quantum circuit of the well-known Grover algorithm with the explicit oracle to solve the HCP for the 4-clique and the 5-clique complete graphs. The quantum circuit is shown to have quadratic speedup over the classical unstructured search algorithm for solving the same problem. It is implemented and run by the IBM quantum computer simulator to validate that it can derive HCs for the 4-clique and the 5-clique. Since the implicit oracle is more practical than the explicit oracle, we plan to build the quantum circuit of the Grover algorithm with the implicit oracle to solve the HCP for general graphs and other related problems.

References

[1] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, ... , and J. M. Martinis, "Quantum supremacy using a programmable superconducting processor," Nature, 574(7779), 505-510, 2019.

[2] J. Preskill, "Quantum computing in the NISQ era and beyond," Quantum, 2, 79, 2018.

[3] L. K. Grover, "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.

[4] C. H. Bennett, E. Bernstein, G. Brassard, U. Vazirani, "The strengths and weaknesses of quantum computation," SIAM Journal on Computing, 26(5), pp. 1510–1523, 1997.

[5] C. Durr, and P. Hoyer, "A quantum algorithm for finding the minimum," arXiv preprint quant-ph/9607014, 1996.

[6] A. Ahuja, and S. Kapoor, A quantum algorithm for finding the maximum. arXiv preprint quant-ph/9911082, 1999.

[7] L. K. Grover, "A framework for fast quantum mechanical algorithms," In Proceedings of the thirtieth annual ACM symposium on Theory of computing, pp. 53-62, 1998.

[8] G. Brassard, F. Dupuis, S. Gambs, and A. Tapp, "An optimal quantum algorithm to approximate the mean and its application for approximating the median of a set of points over an arbitrary distance," arXiv preprint arXiv:1106.4267.

[9] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, "Quantum amplitude amplification and estimation," Contemporary Mathematics, 305, pp. 53–74, 2002.

[10] G. Brassard, P. Hoyer, and A. Tapp, "Quantum algorithm for the collision problem," arXiv preprint quant-ph/9705002, 1997.

[11] J.-R. Jiang, "Easy to learn quantum programming (in Chinese)," Gotop, 2022.

[12] G. Chen, S. A. Fulling, H. Lee, and M. O. Scully, "Grover's algorithm for multiobject search in quantum computing," in Directions in Quantum Optics, pp. 165-175, 2001.

[13] IBM quantum service, utl: https://quantum-computing.ibm.com/, last accessed on December 12, 2022.