Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 8, Number 1 (2012), pp. 65–69

http://www.ripublication.com/gjpam.htm

Quantum Algorithm for Traveling Salesman Problem

Toru Fujimura
Chemical Department, Industrial Property Cooperation Center,
1-2-15, Kiba, Koto-ku, Tokyo 135-0042, Japan
E-mail: tfujimura8@gmail.com

Abstract

A quantum algorithm for the traveling salesman problem and its example are reported. A route of the shortest distance is decided on turning round n points with fixing a starting point. When the counter routes are excluded, a computational complexity of a classical computation is $(n - 1)!/2[= N]$ times. In the quantum algorithm by using quantum phase inversion gates and quantum inversion about mean gates, its computational complexity is about $N^{1/2} log_2 N$ times. Therefore, a high-speed process becomes possible.

AMS Subject Classification: Primary 81-08, Secondary 68R10, 68W40.
Keywords: Quantum algorithm, traveling salesman problem, computational complexity.

1. Introduction

A quantum computer can compute speedily by a parallel computation that uses quantum entangled states. Deutsch-Jozsa's algorithm for the rapid solution [1–3], Shor's algorithm for the factorization [2–4], Grover's algorithms for the database search [2,5–9], Durr-Hoyer's algorithm for finding a minimum [10], Brassard-Hoyer-Tapp's algorithm for counting [11], and so on are known. A quantum algorithm for the knapsack problem has recently been reported by Fujimura [12]. The traveling salesman problem [2,3] is examined this time to expand the application range of the quantum computation. Therefore, its result is reported.

2. Traveling salesman problem

It is the traveling salesman problem to decide a route that turns round n points in the shortest distance. A computational complexity [3] of a classical computation is $(n-1)!/2$ times because a starting point is fixed and counter routes are excluded.

3. Quantum algorithm

First of all, some assumptions are shown. It is assumed that n points of $P_0(x_0, y_0), \ldots, P_{n-2}(x_{n-2}, y_{n-2})$, and $P_{n-1}(x_{n-1}, y_{n-1})$ are set, and P_0 is fixed. Therefore, routes of $P_1, \ldots, P_{n-2}$, and P_{n-1} are considered. Quantum registers of $|a>, |b>$, and $|c>$ are set to $|0>$. $|a>$ is composed of second quantum registers of $|a_1>, \ldots, |a_{n-2}>$, and $|a_{n-1}>$. Here, it is assumed that N is $(n-1)!/2$, W_0 is $(2^\beta)^{n-1}[\approx n^{n-1}]$, β is a minimum integer that becomes

$$log_2 n \le \beta, \tag{3.1}$$

α is $\gamma(n-2)!$, and γ is a minimum integer that becomes

$$(n-1)/2 \le \gamma. \tag{3.2}$$

$|a>$ is used to generate numbers of the $n-1$ points of $P_1, \ldots, P_{n-2}$, and P_{n-1}. When the Hadamard gate is $\boxed{H}$ [2,3], each quantum bit (=qubit) [2,3] of $|a_1>, \ldots, |a_\gamma>, \ldots, |a_{n-2}>$, and $|a_{n-1}>$ is operated by $\boxed{H}$. A number of pieces of $\boxed{H}$ is β. This corresponds to the number of qubits of individual register. As a result, $|a_i>$ shows the state of 2^β pieces of $|0\ldots0>, \ldots, |1\ldots0>$, and $|1\ldots1>$. Here, i is an integer from 1 to $n-1$. It is assumed that a distance M_0 of route of $P_0 \to P_1 \to P_2 \to \ldots \to P_{n-2} \to P_{n-1} \to P_0$ is $((x_1-x_0)^2+(y_1-y_0)^2)^{1/2}+\ldots+((x_0-x_{n-1})^2+(y_0-y_{n-1})^2)^{1/2}$.

Moreover, following gates are set.

(A_m): m is an integer from 1 to $n-1$. When a state of $|a_i>$ is m, 1 is added to $|c>$, and when it of $|a_i>$ isn't m, $|c>$ is made the state as it is.

(PI): A quantum phase inversion gate [2,5–9] reverses a phase of a probability amplitude for a target state.

(IM) : A quantum inversion about mean gate [2,5–9] reverses to a mean of the probability amplitude.

(OB): An observation gate.

(DC): This gate calculates a distance of each route of $|a_1>, \ldots, |a_{n-2}>$, and $|a_{n-1}>$.

(ST): When a state of one register is smaller than a set state, 1 is substituted for other register.

An algorithm is shown in following sentences.

Step 1: $|a>, |b>$, and $|c>$ are set to $|0>$, and a state of M_0 are set. Next, $|a_1>, \ldots, |a_{n-2}>$, and $|a_{n-1}>$ are acted by $\boxed{H}$.

Step 2: At a gate (A_1), when a state of $|a_1>$ is 1, 1 is added to $|c>$, and when it of $|a_1>$ isn't 1, $|c>$ is made a state as it is. This is repeated to $|a_\gamma>$. $|c>$ is entangled to routes of $|a_1>, \ldots, |a_{n-2}>$, and $|a_{n-1}>$. By the gate action, each state of $|c>$ takes a state from 0 to γ. Then, when the gates of both (PI) and (IM) act on $|c>$, a target state of the gate (PI) is 1, and $|c>$ is observed at the gate (OB), only the routes that contain 1 piece of 1 leave. The total action times of both (PI) and (IM) are about $(W_0/W_1)^{1/2}$, where

$$W_m = (2^\beta - m)^{n-(m+1)}\gamma(n-2)!/(n-(m+1))!, \tag{3.3}$$

m is an integer from 1 to $n-1$, and W_{n-1} is $\gamma(n-2)![=\alpha]$.

Step 3: $|c>$ is set to $|0>$. At the gate (A_2), when a state of $|a_1>$ is 2, 1 is added to $|c>$, and when it of $|a_1>$ isn't 2, $|c>$ is made a state as it is. This is repeated to $|a_{n-1}>$. $|c>$ is entangled to the routes of $|a_1>, \ldots, |a_{n-2}>$, and $|a_{n-1}>$. By the gate action, each state of $|c>$ takes a state from 0 to $n-1$. When the gates of both (PI) and (IM) act on $|c>$, the target state of the gate (PI) is 1, and $|c>$ is observed at the gate (OB), only the routes that contain 1 piece of 2 leave. The total action times of both (PI) and (IM) are about $(W_1/W_2)^{1/2}$ in eq. (3). When the operations to the gate (A_{n-1}) that contain the gates of (PI), (IM), and (OB) are similarly done, and the target state of the gate (PI) is 1, and $|c>$ is reset to $|0>$ after the gate (OB), only the routes that contain 1 piece of number from 1 to $n-1$, respectively leave. After all, counter routes are excluded, and a total number of the routes is $W_{n-1} = \alpha \approx (n-1)!/2 = N$. However, when $n-1$ is an odd number, counter routes remain slightly.

Step 4: A distance of each route is calculated at the gate (DC). It is substituted for the register $|b>$. And then, $|c>$ is reset to $|0>$. At the gate (ST), when the state of $|b>$ is smaller than M_0, 1 is substituted for $|c>$. And, the gates of both (PI) and (IM) act on $|c>$. Here, the target state of the gate (PI) is 1. The total action times of both (PI) and (IM) are about $\alpha^{1/2}$.

Step 5: At the gate (OB), $|a>$, $|b>$, and $|c>$ are observed. The route of $|a>$, the distance M_1 of $|b>$, and the state of $|c>$ are gotten. When the state of $|c>$ is 0, this computation ends. When it of $|c>$ is 1, M_0 is assumed to be M_1 and this calculation is repeated. It is assumed that a minimum distance M_{min} obtains by repeating about $log_2\,\alpha$ times [10].

4. Numerical calculation

An example is given here. It is assumed that n is 9, and coordinates of the various place point are $P_0(0,0)$, $P_1(3,2)$, $P_2(1,0)$, $P_3(2,2)$, $P_4(3,0)$, $P_5(0,1)$, $P_6(2,2)$, $P_7(2,0)$, and $P_8(1,2)$. The state of M_0 is 22.8. $|a>$ is composed of $|a_1>, \ldots, |a_7>$, and $|a_8>$, $|b>$ is composed of qubits enough for expression of the distance, and $|c>$ is composed of 4 qubits. Here, α is $20160[=N]$. β is 4 because eq.(1) becomes $log_2 9 \approx 3.2 \leq 4$. W_0 is $(2^\beta)^{n-1} = (2^4)^8 = 16^8 \approx 4.3 \times 10^9$. γ is 4 because eq.(2) is $(9-1)/2 = 4 \leq 4$. Moreover, in above-mentioned coordinates, the best routes are $P_0, P_5, P_8, P_3, P_1, P_6, P_4, P_7, P_2, P_0$, and the counter route. Its minimum distance is 9.4. However, the counter route is excluded by the gate (A_1) that operates $|a_1>$, $|a_2>$, $|a_3>$, and $|a_4>$, respectively. Therefore, the best route of $|a_1>, \ldots, |a_7>$, and $|a_8>$ is 5, 8, 3, 1, 6, 4, 7, and 2.

Now then, $|c>$ passes following gates, respectively and it is reset to $|0>$ after the gate (OB).

$$(A_1)[4\ times] \rightarrow [(PI),(IM)][total\ about\ (W_0/W_1)^{1/2}\ times] \rightarrow (OB) \rightarrow$$

$$|0> \rightarrow (A_2)[8\ times] \rightarrow [(PI),(IM)][total\ about\ (W_1/W_2)^{1/2}\ times] \rightarrow$$

$$(OB) \rightarrow |0> \rightarrow \ldots \rightarrow (A_8)[8\ times] \rightarrow$$

$$[(PI), (IM)][total\ about\ (W_7/W_8)^{1/2}\ times] \rightarrow (OB).$$

At the gate (PI), when a state of $|c>$ is 1, a phase of a probability amplitude of $|c>$ is inverted. At the gate (IM), it reverses to a mean of the probability amplitude of $|c>$. These processes are repeated. A frequency becomes $(W_0/W_1)^{1/2} \approx 2.5 \leq 4$, $(W_1/W_2)^{1/2} \approx 1.8 \leq 2$, $(W_2/W_3)^{1/2} \approx 1.8 \leq 2$, $(W_3/W_4)^{1/2} \approx 1.9 \leq 2$, $(W_4/W_5)^{1/2} \approx 2.0 \leq 2$, $(W_5/W_6)^{1/2} \approx 2.1 \leq 4$, $(W_6/W_7)^{1/2} \approx 2.4 \leq 4$, and $(W_7/W_8)^{1/2} \approx 3.0 \leq 4$ in eq.(3), respectively. However, because actions of the gates of both (PI) and (IM) are single-unit by two, it is limited to even number times. At the gate (OB), $|c>$ is observed after the gates of (A_m), (PI), and (IM), respectively, where m is an integer from 1 to 8, and $|c>$ is reset to $|0>$ after the gate (OB), respectively. Therefore, a number of routes of $|a_1>, \ldots, |a_7>$, and $|a_8>$ that are left is 20160 only.

Then, a distance of each route is calculated at the gate (DC). It is substituted for $|b>$. Next, $|c>$ is reset to $|0>$. At the gate (ST), when a state of $|b>$ is smaller than $M_0 = 22.8$, 1 is substituted for $|c>$. And, the gates of both (PI) and (IM) operated $|c>$. Here, the target state of the gate (PI) is 1. Total operation times of the gates of both (PI) and (IM) is $20160^{1/2}[\leq 142]$.

Finally, at the gate (OB), $|a>$, $|b>$, and $|c>$ are observed. A route of $|a>$, a distance M_1 of $|b>$, and a state of $|c>$ are gotten. Because it of $|c>$ is 1, M_0 is assumed to be M_1, and this computation is repeated. It is assumed that a minimum distance M_{min} obtains by repeating about $log_2 20160\, [\approx 14]$ times. When a state of $|c>$ is 0, this calculation ends. As a result , the best route 5, 8, 3, 1, 6, 4, 7, 2, and the minimum distance 9.4 are obtained.

5. Discussion and Summary

In general, a frequency of necessary gates and calculations is $\beta(n-1)$ times at $\boxed{\text{H}}$, $\gamma + (n-1)(n-2)$ times at the gates $(A_1), \ldots, (A_{n-2})$, and (A_{n-1}), about $\alpha^{1/2} + \sum_{f=0}^{n-2}(W_f/W_{f+1})^{1/2}$ times at the gates of both (PI) and (IM), $n-1$ times at the gate (DC), once at the gate (ST), and n times at the gate (OB). These processes are repeated about $log_2\, \alpha$ times. Therefore, total frequency is about $(\alpha^{1/2} + (n+\beta)(n-1) + \gamma + 2 + \sum_{f=0}^{n-2}(W_f/W_{f+1})^{1/2})log_2\, \alpha$ times. After all, if n is large enough, a computational complexity becomes $\alpha^{1/2}log_2\, \alpha = (\gamma(n-2)!)^{1/2}log_2\, (\gamma(n-2)!) \approx ((n-1)!/2)^{1/2}log_2\, ((n-1)!/2) = N^{1/2}log_2\, N$ times.

The computational complexity at $n = 9$ in section 4 is about 3900 times. Because the computational complexity of a classical computation is 20160 times, it of this quantum algorithm becomes about $1/5$.

Here, it thinks in case of $n = 50$. In the classical computation, the computational complexity of about 10^{62} times is necessary. In the quantum algorithm, it is about 10^{33} times. Therefore, a high-speed process becomes possible.

References

[1] Deutsch D., and Jozsa R., Rapid solution of problems by quantum computation, Proc. Roy. Soc. Lond. A, 439:553–558, 1992.

[2] Takeuchi S., Ryoshi Konpyuta (Quantum Computer), Kodansha, Tokyo, Japan, 2005 [in Japanese].

[3] Miyano K., and Furusawa A., Ryoshi Konpyuta Nyumon (An Introduction to Quantum Computation), Nihonhyoronsha, Tokyo, Japan, 2008 [in Japanese].

[4] Shor P.W., Algorithms for quantum computation: discrete logarithms and factoring, *Proc. 35th Annu. Symp. Foundations of Computer Science*, IEEE, pp. 124–134, 1994.

[5] Grover L.K., A fast quantum mechanical algorithm for database search, *Proc. 28th Annu. ACM Symp. Theory of Computing*, pp. 212–219, 1996.

[6] Grover L.K., Quantum mechanics helps in searching for a needle in a haystack, *Phys. Rev. Lett.*, 79:325–328, 1997.

[7] Grover L.K., A framework for fast quantum mechanical algorithms, *Proc. 30th Annu. ACM Symp. Theory of Computing*, pp. 53–62, 1998.

[8] Grover L.K., Quantum computers can search rapidly by using almost any transformation, *Phys. Rev. Lett.*, 80:4329–4332, 1998.

[9] Grover L.K., Superlinear amplitude amplification, [Online], 2008. Available: http://arXiv.org/quant-ph/arXiv:0806.0154v1 [quant-ph].

[10] Durr C., and Hoyer P., A quantum algorithm for finding the minimum, [Online], 1996. Available: http://arXiv.org/quant-ph/arXiv: quant-ph/9607014v2.

[11] Brassard G., Hoyer P., and Tapp A., Quantum counting, [Online], 1998. Available: http://arXiv.org/quant-ph/arXiv:quant-ph/9805082v1.

[12] Fujimura T., Quantum algorithm for knapsack problem, *Glob. J. Pure Appl. Math.*, 6:263–266, 2010.