Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 9, Number 6 (2013), pp. 545–551

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

Quantum Algorithm for Traveling Salesman Problem by Numbering Method

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 by a numbering method 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 $(n-1)!/2$. In the quantum algorithm by using quantum phase inversion gates, quantum inversion about mean gates and the numbering method, its computational complexity is about $3(log_2(n-1))^2(n-1)^2$. Therefore, a polynomial time process becomes possible.

AMS subject classification: Primary 81-08; Secondary 68R10, 68W40.
Keywords: Quantum algorithm, traveling salesman problem, numbering method, computational complexity, polynomial time.

1. Introduction

The methods for the very first steps towards building a quantum computer were developed by Haroche and Wineland [1]. The algorithms of the quantum computer by Deutsch-Jozsa [2–4], Shor [3–5], Grover [3, 6, 7] and so on are known. A quantum algorithm for the vertex coloring problem by the central limit theorem has recently been reported by Fujimura [8]. Its computational complexity becomes a polynomial time. The traveling salesman problem [3,4] is examined this time. 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. The computational complexity of a classical computation is $(n-1)!/2$ because a starting point is fixed and counter routes are excluded.

3. Quantum Algorithm

It is assumed that n points of $P_0(x_0, y_0), P_1(x_1, y_1), \cdots, P_{n-2}(x_{n-2}, y_{n-2})$ and $P_{n-1}(x_{n-1}, y_{y-1})$ are set, P_0 is fixed, and a distance between P_i and P_j is $L_{i,j}[=L_{j,i}]$. Therefore, routes of $P_1, P_2, \cdots, P_{n-2}, P_{n-1}$ are considered.

(1) The number of the repeated permutation of $n-1$ points is $(n-1)^{n-1}$.

(2) The number of the permutation of $n-1$ points is $(n-1)!$.

When $(n-1)$ points are $P_{a_1}, P_{a_2}, \cdots, P_{a_{n-2}}$ and $P_{a_{n-1}}$, $a_1(n-1)^{n-2}+a_2(n-1)^{n-3}+\cdots+a_{n-2}(n-1)^1+a_{n-1}(n-1)^0 = U$ is the numbering datum from 0 to $(n-1)^{n-1}-1$ [The 0-th datum is $0, 0, \cdots, 0$ and 0. The $((n-1)^{n-1}-1)$-th datum is $(n-2), (n-2), \cdots, (n-2)$ and $(n-2)$.] in (1). In (2), it is assumed that the first datum is $0, 1, \cdots, n-2$, and the $(n-1)!$-th datum is $(n-2), (n-3), \cdots, 0$, the V-th datum is obtained from $v_1(n-2)! + v_2(n-3)! + \cdots + v_{n-2}1!$. Each of φ_i $[1 \le i \le n-1$. i is an integer.] is 1 piece of permutation from 0 to $n-2$. When v_i is 0 from $i=1$ to $i=n-3$ sequentially, φ_i is the smallest number in remained numbers. When v_i isn't 0 from $i=1$ to $i=n-3$ sequentially, and $v_{i+1}, v_{i+2}, \cdots, v_{n-3}$ and v_{n-2} are 0, φ_i is the v_i-th small number in remained numbers, and $\varphi_{i+1} > \varphi_{i+2} > \cdots > \varphi_{n-2} > \varphi_{n-1}$ is selected in remained numbers. When v_i isn't 0 from $i=1$ to $i=n-3$ sequentially, and there are $v_{i+1} \ne 0$ or $v_{i+2} \ne 0$ or $\cdots$ or $v_{n-3} \ne 0$ or $v_{n-2} \ne 0$, φ_i is the (v_i+1)-th small number in remained numbers. When v_{n-2} is 1, $\varphi_{n-2} < \varphi_{n-1}$ is selected in remained numbers. Therefore, $\varphi_1(n-1)^{n-2} + \varphi_2(n-1)^{n-3} + \cdots + \varphi_{n-2}(n-1)^1 + \varphi_{n-1}(n-1)^0$ is $U(V)$. This method is named a numbering method for this problem. g that is the minimum integer follows $(n-1)!/2 \le 2^{2g} = 4^g$. $U(V=1), U(V=((n-1)!/4)-2), U(V=((n-1)!/16)-2), \cdots, U(V=((n-1)!/4^{g-1})-2)$ and $U(V=(n-1)!/4^g)$ are computated. M_1 that is a starting distance value is decided at random.

Next, a quantum algorithm is shown as the following.

First of all, quantum registers $|a_1>, |a_2>, \cdots, |a_{n-1}>, |b_1>, |b_2>, \cdots, |b_{n-2}>, |c_1>, |c_2>, |d>$ and $|e>$ are prepared. When α is the minimum integer that is $log_2(n-1)$ or more, each of $|a_f>$ that f is an integer from 1 to $n-1$ is consisted of α quantum bits [= qubits]. States of $|a_1>, |a_2>, \cdots, |a_{n-1}>, |b_1>, |b_2>, \cdots, |b_{n-2}>, |c_1>, |c_2>, |d>$ and $|e>$ are $a_1, a_2, \cdots, a_{n-1}, b_1, b_2, \cdots, b_{n-2}, c_1, c_2, d$ and e, respectively.

Step 1: Each qubit of $|a_1>, |a_2>, \cdots, |a_{n-1}>, |b_1>, |b_2>, \cdots, |b_{n-2}>, |c_1>, |c_2>, |d>$ and $|e>$ is set $|0>$.

Step 2: The Hadamard gate $\boxed{\text{H}}$ [3, 4] acts on each qubit of $|a_1>, |a_2>, \cdots, |a_{n-2}>$ and $|a_{n-1}>$. It changes them for entangled states. The total states are $(2^\alpha)^{n-1}$.

Step 3: It is assumed that a quantum gate (A) doesn't change $|b_1>$ in $a_f < n-1$, or it changes $|b_1>$ for $|b_1+1>$ in the others of a_f. As a target state for $|b_1>$ is 0, quantum phase inversion gates (PI) and quantum inversion about mean gates (IM) [3, 6, 7] act on $|b_1>$. When β is the minimum even integer that is $(2^\alpha/(n-1))^{1/2}$ or more, the total number that (PI) and (IM) act on $|b_1>$ is β, because they are a couple. Next, an observation gate (OB) observes $|b_1>$. These actions are repeated sequentially from $|a_1>$ to $|a_{n-1}>$. Therefore, each state of $|a_f>$ is $0, 1, \cdots, n-3$ and $n-2$, and the total states become $(n-1)^{n-1}$.

Step 4: It is assumed that a quantum gate (B) changes $|b_1>, |b_2>, \cdots, |b_{n-3}>$ and $|b_{n-2}>$ for $|b_1+1>, |b_2+1>, \cdots, |b_{n-3}+1>$ and $|b_{n-2}+1>$ in $a_f = 0, 1, \cdots, n-4$ and $n-3$, respectively. This action repeats from $|a_1>$ to $|a_{n-1}>$. As the target state for $|b_1>$ is 1, (PI) and (IM) act on $|b_1>$. When γ_1 is the minimum even integer that is $((n-1)/(n-2))^{(n-2)/2}$ or more, the total number that (PI) and (IM) act on $|b_1>$ is γ_1. Next, (OB) observes $|b_1>$. Therefore, only the routes that contain 1 piece of 0 remain. The number of data is $(n-1)(n-2)^{n-2}$. As the target state for $|b_2>$ is 1, (PI) and (IM) act on $|b_2>$. When γ_2 is the minimum even integer that is $((n-2)/(n-3))^{(n-3)/2}$ or more, the total number that (PI) and (IM) act on $|b_2>$ is γ_2. Next, (OB) observes $|b_2>$. Therefore, only the routes that contain 1 piece of 1 remain. The number of data is $(n-1)(n-2)(n-3)^{n-3}$. Similarly, these actions are repeated sequentially from $|b_3>$ to $|b_{n-2}>$. Only the routes that contain 1 piece of number from 0 to $n-2$, respectively, remain. The number of data is $(n-1)![=W_0]$.

Step 5: It is assumed that a quantum gate (C_1) changes $|c_1>$ and $|c_2>$ for $|c_1 + L_{0,a_1} + L_{a_1,a_2}>$ and $|c_2 + (n-1)^{n-2}a_1 + (n-1)^{n-3}a_2>$, respectively, from $|a_1>$ and $|a_2>$. Similarly, (C_i) $[2 \leq i \leq n-3.$ i is the integer.$]$ changes $|c_1>$ and $|c_2>$ for $|c_1 + L_{a_i,a_{i+1}}>$ and $|c_2 + (n-1)^{n-(i+2)}a_{i+1}>$, respectively, from $|a_i>$ and $|a_{i+1}>$. This action is repeated sequentially from $|a_2>$ to $|a_{n-3}>$. (C_{n-2}) changes $|c_1>$ and $|c_2>$ for $|c_1 + L_{a_{n-2},a_{n-1}} + L_{a_{n-1},0}>$ and $|c_2 + (n-1)^0 a_{n-1}>$, respectively, from $|a_{n-2}>$ and $|a_{n-1}>$. Therefore, $|c_1>$ and $|c_2>$ become $|L_{total} = L_{0,a_1} + L_{a_1,a_2} + \cdots + L_{a_{n-2},a_{n-1}} + L_{a_{n-1},0}>$ and $|U>$, respectively.

Step 6: It is assumed that a quantum gate (D) changes $|d>$ for $|d + c_1>$ in $c_1 \leq M_1$, or it changes $|d>$ for $|d + M_1 + c_2>$ in the others of c_1.

Step 7: It is assumed that a quantum gate (E_1) doesn't change $|e>$ in $d \leq M_1$ and $M_1 + U(V=1) \leq d \leq M_1 + U(V = ((n-1)!/4) - 2)$, or it changes $|e>$ for $|e+1>$ in the others of d. As the target state for $|e>$ is 0, (PI) and (IM) act on $|e>$. The number of the data that is included in $d \leq M_1$ and $M_1 + U(V=1) \leq d \leq M_1 + U(V = ((n-1)!/4) - 2)$ is $W_1 \approx (n-1)!/4$. When δ_1 is the minimum even integer that is $(W_0/W_1)^{1/2}$ or more, the total number that (PI) and (IM) act on $|e>$ is $\delta_1 \approx 2$. Next, (OB) observes $|e>$, and the data of W_1 remain. Similarly, (E_i) $[2 \leq i \leq g-1.$ i is the integer.$]$ doesn't change $|e>$ in $d \leq M_1$ and $M_1 + U(V=1) \leq d \leq M_1 + U(V = ((n-1)!/4^i) - 2)$, or it changes $|e>$ for $|e+1>$ in the others of d. As the target state for $|e>$ is 0, (PI) and (IM) act on $|e>$. The number of the data that is included in $d \leq M_1$ and $M_1 + U(V=1) \leq d \leq M_1 + U(V = ((n-1)!/4^i) - 2)$ is

$W_i \approx (n-1)!/4^i$. When δ_i is the minimum even integer that is $(W_{i-1}/W_i)^{1/2}$ or more, the total number that (PI) and (IM) act on $|e>$ is $\gamma_i \approx 2$. Next, (OB) observes $|e>$, and the data of W_i remain. These actions are repeated sequentially from 2 to $g-1$ at i. (E_g) doesn't change $|e>$ in $d \leq M_1$, or it changes $|e>$ for $|e+1>$ in the others of d. As the target state for $|e>$ is 0, (PI) and (IM) act on $|e>$. The number of the data that is included in $d \leq M_1$ is $W_g \approx 2$. When δ_g is the minimum even integer that is $(W_{g-1}/W_g)^{1/2}$ or more, the total number that (PI) and (IM) act on $|e>$ is $\delta_g \approx 2$. Next, (OB) observes $|a_1>, |a_2>, \cdots, |a_{n-1}>, |b_1>, |b_2>, \cdots, |b_{n-2}>, |c_1>, |c_2>, |d>$ and $|e>$, and one of the data of W_g remains. Therefore, one example of routes that are $L_{total} \leq M_1$ is obtained.

Step 8: When the state of $|e>$ is 0 or 1, M_1 is assumed to be $M_2[<M_1]$ or $M_2[>M_1]$, respectively, these computations from step 1 to step 8 are repeated. It is assumed that the minimum distance M_{min} obtains by repeating about $log_2(n-1)!$ [9].

4. Numerical Computation

It is assumed that there are $n = 10, P_0(0,0), P_1(1,-2), P_2(3,-1), P_3(4,1), P_4(2,3), P_5(1,-1), P_6(3,-2), P_7(4,0), P_8(0,1), P_9(2,2), g = 9[9!/2 = 181440 \leq 4^9 = 262144]$, $U(V=1) = 6053444, U(V=(9!/4)-2=90718) = 95584572$ [for examle, $V = 90718 = 2\cdot 8! + 1\cdot 7! + 6\cdot 6! + 5\cdot 5! + 4\cdot 4! + 3\cdot 3! + 2\cdot 2! + 0\cdot 1!$, $U = 95584572 = 2\cdot 9^8 + 1\cdot 9^7 + 8\cdot 9^6 + 7\cdot 9^5 + 6\cdot 9^4 + 5\cdot 9^3 + 3\cdot 9^2 + 4\cdot 9^1 + 0\cdot 9^0$], $U(V=(9!/16)-2=22678) = 26275564, U(V=(9!/64)-2=5668) = 10598756, U(V=(9!/256)-2=1416) = 6894596, U(V=(9!/1024)-2=352) = 6198348, U(V=(9!/4096)-2=87) = 6073748, U(V=(9!/16384)-2=20) = 6055548, U(V=(9!/65536)-2=4) = 6053532$ and $M_1 = 20$.

First of all, $|a_1>, |a_2>, \cdots, |a_9>, |b_1>, |b_2>, \cdots, |b_8>, |c_1>, |c_2>, |d>$ and $|e>$ are prepared. When α is the minimum integer that is $log_2(n-1) = log_2 9 \approx 3.170 \leq 4 = \alpha$, each of $|a_f>$ that f is the integer from 1 to 9 is consisted of $\alpha = 4$ qubits. States of $|a_1>, |a_2>, \cdots, |a_9>, |b_1>, |b_2>, \cdots, |b_8>, |c_1>, |c_2>, |d>$ and $|e>$ are $a_1, a_2, \cdots, a_9, b_1, b_2, \cdots, b_8, c_1, c_2, d$ and e, respectively.

Step 1: Each qubit of $|a_1>, |a_2>, \cdots, |a_9>, |b_1>, |b_2>, \cdots, |b_8>, |c_1>, |c_2>$ $, |d>$ and $|e>$ is set $|0>$.

Step 2: $\boxed{\text{H}}$ acts on each qubit of $|a_1>, |a_2>, \cdots, |a_8>$ and $|a_9>$. It changes them for entangled states. The total states are $(2^\alpha)^{n-1} = (2^4)^9 = 16^9$.

Step 3: (A) doesn't change $|b_1>$ in $a_f < n-1 = 9$, or it changes $|b_1>$ for $|b_1+1>$ in the others of a_f. As the target state for $|b_1>$ is 0, (PI) and (IM) act on $|b_1>$. When β is the minimum even integer that is $(2^\alpha/(n-1))^{1/2} = (16/9)^{1/2} \approx 1.333 \leq 2 = \beta$, the total number that (PI) and (IM) act on $|b_1>$ is $\beta \approx 2$. Next, (OB) observes $|b_1>$. These actions are repeated sequentially from $|a_1>$

to $|a_9>$. Therefore, each state of $|a_f>$ is $0, 1, \cdots, 7$ and 8, and the total states become $(n-1)^{n-1} = 9^9$.

Step 4: (B) changes $|b_1>, |b_2>, \cdots, |b_7>$ and $|b_8>$ for $|b_1+1>, |b_2+1>, \cdots, |b_7+1>$ and $|b_8+1>$ in $a_f = 0, 1, \cdots, 6$ and 7, respectively. This action is repeated from $|a_1>$ to $|a_9>$. As the target state for $|b_1>$ is 1, (PI) and (IM) act on $|b_1>$. When γ_1 is the minimum even integer that is $((n-1)/(n-2))^{(n-2)/2} = (9/8)^4 \approx 1.602 \leq 2 = \gamma_1$, the total number that (PI) and (IM) act on $|b_1>$ is γ_1. Next, (OB) observes $|b_1>$. Therefore, only the routes that contain 1 piece of 0 remain. The number of data is $(n-1)(n-2)^{n-2} = 9 \cdot 8^8$. As the target state for $|b_2>$ is 1, (PI) and (IM) act on $|b_2>$. When γ_2 is the minimum even integer that is $((n-2)/(n-3))^{(n-3)/2} = (8/7)^{3.5} \approx 1.596 \leq 2 = \gamma_2$, the total number that (PI) and (IM) act on $|b_2>$ is γ_2. Next, (OB) observes $|b_2>$. Therefore, only the routes that contain 1 piece of 1 remain. The number of data is $(n-1)(n-2)(n-3)^{n-3} = 9 \cdot 8 \cdot 7^7$. Similarly, these actions are repeated sequentially from $|b_3>$ to $|b_8>$. Only the routes that contain 1 piece of number from 0 to 8, respectively, remain. The number of data is $(n-1)! = 9![= W_0]$.

Step 5: (C_1) changes $|c_1>$ and $|c_2>$ for $|c_1+L_{0,a_1}+L_{a_1,a_2}>$ and $|c_2+9^8a_1+9^7a_2>$, respectively, from $|a_1>$ and $|a_2>$. Similarly, (C_i) $[2 \leq i \leq 7$. i is the integer.$]$ changes $|c_1>$ and $|c_2>$ for $|c_1+L_{a_i,a_{i+1}}>$ and $|c_2+9^{10-(i+2)}a_{i+1}>$, respectively, from $|a_i>$ and $|a_{i+1}>$. This action is repeated sequentially from $|a_2>$ to $|a_7>$. (C_8) changes $|c_1>$ and $|c_2>$ for $|c_1+L_{a_8,a_9}+L_{a_9,0}>$ and $|c_2+9^0a_9>$, respectively, from $|a_8>$ and $|a_9>$. Therefore, $|c_1>$ and $|c_2>$ become $|L_{total} = L_{0,a_1}+L_{a_1,a_2}+\cdots+L_{a_8,a_9}+L_{a_9,0}>$ and $|U>$, respectively.

Step 6: (D) changes $|d>$ for $|d+c_1>$ in $c_1 \leq M_1 = 20$, or it changes $|d>$ for $|d+20+c_2>$ in the others of c_1.

Step 7: (E_1) doesn't change $|e>$ in $d \leq M_1 = 20$ and $M_1+U(V=1) = 20+6053444 \leq d \leq M_1+U(V=((n-1)!/4)-2) = 20+95584572$, or it changes $|e>$ for $|e+1>$ in the others of d. As the target state for $|e>$ is 0, (PI) and (IM) act on $|e>$. The number of the data that is included in $d \leq 20$ and $6053464 \leq d \leq 95584592$ is $W_1 \approx 9!/4$. When δ_1 is the minimum even integer that is $(W_0/W_1)^{1/2}$ or more, the total number that (PI) and (IM) act on $|e>$ is $\delta_1 \approx 2$. Next, (OB) observes $|e>$, and the data of W_1 remain.

Similarly, (E_i) $[2 \leq i \leq g-1 = 8$. i is the integer.$]$ doesn't change $|e>$ in $d \leq 20$ and $6053464 \leq d \leq 20+U(V=(9!/4^i)-2)$, or it changes $|e>$ for $|e+1>$ in the others of d. As the target state for $|e>$ is 0, (PI) and (IM) act on $|e>$. The number of the data that is included in $d \leq 20$ and $6053464 \leq d \leq 20+U(V=(9!/4^i)-2)$ is $W_i \approx 9!/4^i$. When δ_i is the minimum even integer that is $(W_{i-1}/W_i)^{1/2}$ or more, the total number that (PI) and (IM) act on $|e>$ is $\gamma_i \approx 2$. Next, (OB) observes $|e>$, and the data of W_i remain. These actions are repeated sequentially from 2 to $g-1$ at i. (E_g) doesn't change $|e>$ in $d \leq 20$, or it changes $|e>$ for $|e+1>$

in the others of d. As the target state for $|e>$ is 0, (PI) and (IM) act on $|e>$. The number of the data that is included in $d \leq 20$ is $W_9 \approx 2$. When δ_9 is the minimum even integer that is $(W_8/W_9)^{1/2}$ or more, the total number that (PI) and (IM) act on $|e>$ is $\delta_9 \approx 2$. Next, (OB) observes $|a_1>, |a_2>, \cdots, |a_9>, |b_1>, |b_2>, \cdots, |b_8>, |c_1>, |c_2>, |d>$ and $|e>$, and one of the data of W_9 remains. For example, when $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, b_1, b_2, \cdots, b_8, c_1, c_2, d$ and e are $4, 0, 5, 1, 6, 2, 7, 3, 8, 1, 1, \cdots, 1, 18, U(V = 163491) = 174944564, 18$ and 0, respectively.

Step 8: In the example, the state of $|e>$ is 0. Therefore, M_1 is assumed to be $M_2 = 15[< 18 < M_1 = 20]$, and these calculations from step 1 to step 8 are repeated. It is assumed that the state of $|e>$ is 0. When the state of $|e>$ is 1 at $M_3 = 10, M_4 = 13$ and $M_5 = 14$, respectively, the minimum distance M_{min} is $15[= M_2]$. Therefore, $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, b_1, b_2, \cdots, b_8, c_1, c_2, d$ and e are $4, 0, 5, 1, 6, 2, 8, 3, 7, 1, 1, \cdots, 1, 15, U(V = 163493) = 174944644, 15$ and 0, respectively. As a result, the shortest route $P_0 \rightarrow P_5 \rightarrow P_1 \rightarrow P_6 \rightarrow P_2 \rightarrow P_7 \rightarrow P_3 \rightarrow P_9 \rightarrow P_4 \rightarrow P_8 \rightarrow P_0$ is obtained.

5. Discussion and Summary

In the example of the section 4, the computational complexity of this quantum algorithm $[= S]$ is 795. The computational complexity of the classical computation $[= Z]$ is $(n-1)!/2 = 9!/2 = 181440$. After all, S/Z becomes about 1/200. In general, S becomes the following. In the order of the actions by the gates, the number of them is $\alpha(n-1)$ at $\boxed{\mathrm{H}}$, $n-1$ at (A), $\beta(n-1) \approx 2(n-1)$ at (PI) and (IM), $n-1$ at (OB), $n-1$ at (B), $\sum_{i=1}^{n-2} \gamma_i = 2(n-2)$ at (PI) and (IM), $n-2$ at (OB), $2(n-2)$ at (C_i) $[1 \leq i \leq n-2$. i is the integer.], 2 at (D), g at (E_i) $[1 \leq i \leq g.]$, $\sum_{i=1}^{g} \delta_i = 2g$ at (PI) and (IM), and g at (OB). These processes repeated about $log_2(n-1)!$. Therefore, S becomes $(\alpha(n-1) + 10n - 13 + 4g)log_2(n-1)!$. When n is large enough, S becomes about $3(log_2(n-1))^2(n-1)^2$, where α is about $log_2(n-1)$, g is about $(1/2)((log_2(n-1)!) - 1)$, and $n!$ is about $n^n e^{-n}(2\pi n)^{1/2}$ [Stirling's formula], and S/Z is about $3(log_2(n-1))^2(n-1)^2/((n-1)!/2)$. For example, as for $n = 50$, S/Z is about $1/10^{57}$. Therefore, a polynomial time process becomes possible.

References

[1] Kungl. Vetenskapsakademien (The Royal Swedish Academy of Sciences), The Nobel Prize in Physics 2012, [Online], Available: http: //www.kva.se/en/pressroom/ Press-releases-2012/The-Nobel-Prize-in-Physics-2012/, 2012.

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

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

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

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

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

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

[8] Fujimura T., Quantum algorithm for vertex coloring problem by central limit theorem, *Glob. J. Pure Appl. Math.*, 7:401–405, 2011.

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