Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 16, Number 5 (2020), pp. 711-721

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

Quantum Algorithm for Traveling Salesman Problem by Quarter Method

Toru Fujimura

Third area security office, University of Tsukuba, Ibaraki-branch, Rising Sun Security Service Co., Ltd., 1-1-1, Tennodai, Tsukuba, Ibaraki 305-8577, Japan

Abstract

A quantum algorithm for the traveling salesman problem by a quarter 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 $(n - 1)!/2$. The computational complexity becomes about $3(\log_2 (n - 1))^2(n - 1)^2$ by this quantum algorithm. Therefore, a decreased process becomes possible.

Keywords: Quantum algorithm, traveling salesman problem, quarter method, computational complexity, decreased process.

AMS subject classification: Primary 81-08; Secondary 68R10, 68W40.

1. INTRODUCTION

Ono, Mori, and Moriyama reported a high temperature silicon qubit [1]. The algorithms of the quantum computer by Deutsch-Jozsa, Shor, Grover, and so on are known [2-7]. Ambainis's quantum walk algorithms was the example to decrease the computational complexity [8]. When the feature of the problem isn't used, it is difficult to decrease the computational complexity. Bennett, Bernstein, Brassard, and Vazirani addressed the class NP cannot be solved on a quantum Turing machine in time $O(2^{n/2})$ [9]. However, they didn't eliminate the unnecessary data on the machine's way to the end.

For this reason, Fujimura suggested that the probability amplitudes of the maximum integer multiple-choice generalized knapsack problem are converged quickly by a hybrid method of Grover's database search and Shor's data decrease [3, 5-7, 10]. Its computational complexity is decreased. The traveling salesman problem [3, 4, 11] is examined by a quarter method 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, 4, 11].

3. QUANTUM ALGORITHM

It is assumed that n points of $P_0 (x_0, y_0)$, $P_1 (x_1, y_1)$, $P_2 (x_2, y_2)$, $\cdots$, $P_{n-2} (x_{n-2}, y_{n-2})$, and $P_{n-1} (x_{n-1}, y_{n-1})$ are set [x_i and y_i are the two dimensional coordinates. $0 \le i \le n - 1$. i is an integer.], P_0 is fixed, and a distance between P_i and P_j is $L(i, j)[= L(j, i)]$. Therefore, routes of $P_1, P_2, \cdots , P_{n-2}$, and 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 permutation of $n - 1$ points is $(n - 1)!$.

When $n - 1$ points are $P_{1+a(1)}, P_{1+a(2)}, \cdots , P_{1+a(n-2)}$, and $P_{1+a(n-1)}$ [where, $a(q) = a_q$; $0 \le a_q \le n - 2$. a_q is an integer.], it is assumed that $U[X] = a_1 (n - 1)^{n-2} + a_2 (n - 1)^{n-3} + \cdots + a_{n-2} (n - 1)^1 + a_{n-1} (n - 1)^0$ [X is the number of datum.] is the numbering datum from 0 to $(n - 1)^{n-1} - 1$ [For example, $U[X = 0]$ is $a_1 = 0$, $a_2 = 0$, $\cdots$, $a_{n-2} = 0$, and $a_{n-1} = 0$, and $U[X = (n - 1)^{n-1} - 1]$ is $a_1 = n - 2$, $a_2 = n - 2$, $\cdots$, $a_{n-2} = n - 2$, and $a_{n-1} = n - 2$.] in (1).

In (2), it is assumed that the first datum $V(Y = 1)$ is $a_1 = 0$, $a_2 = 1$, $\cdots$, $a_{n-2} = n - 3$, and $a_{n-1} = n - 2$, and the $(n - 1)!$-th datum $V(Y = (n - 1)!)$ is $a_1 = n - 2$, $a_2 = n - 3$, $\cdots$, $a_{n-2} = 1$, and $a_{n-1} = 0$, Y [where, $1 \le Y \le (n - 1)! - 1$. Y is an integer.] is obtained from $v_1 (n - 2)! + v_2 (n - 3)! + \cdots + v_{n-2} 1!$. [Where, $0 \le v_i \le n - (1 + i)$. v_i is an integer.] Each of t_i [$1 \le i \le n - 1$. i is an integer.] is 1 piece of permutation from 0 to $n - 2$.

(I) When v_i is 0 from $i = 1$ to $i = n - 3$ sequentially, t_i is the smallest number in remained numbers.

(II) When v_i isn't 0 from $i = 1$ to $i = n - 3$ sequentially, and v_{i+1}, v_{i+2}, ⋯, v_{n-3}, and v_{n-2} are 0, t_i is the v_i-th small number in remained numbers, and $t_{i+1} > t_{i+2} > \cdots > t_{n-2} > t_{n-1}$ is selected in remained numbers.

(III) When v_i isn't 0 from $i = 1$ to $i = n - 3$ sequentially, and there are $v_{i+1} \neq 0$ or $v_{i+2} \neq 0$ or ⋯ or $v_{n-3} \neq 0$ or $v_{n-2} \neq 0$, t_i is the $(v_i + 1)$-th small number in remained numbers.

(IV) When v_{n-2} is 1, $t_{n-2} < t_{n-1}$ is selected in remained numbers. Therefore, $t_1(n-1)^{n-2} + t_2(n-1)^{n-3} + \cdots + t_{n-2}(n-1)^1 + t_{n-1}(n-1)^0$ is $U(V(Y))$. [Where, t_i is equal to a_i in (2).]

g is the minimum integer that follows $(n-1)!/2 \leq 2^{2g} = 4^g$, because a number of combinations of answers is at least 2. $U(V(Y = 1))$, $U(V(Y = ((n-1)!/4) - 2))$, $U(V(Y = ((n-1)!/16) - 2))$, ⋯ , $U(V(Y = ((n-1)!/4^{g-1}) - 2))$, and $U(V(Y = (n-1)!/4^g))$ are computed. [→ See Appendix-1] 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 $|a_1\rangle$, $|a_2\rangle$, ⋯ , $|a_{n-1}\rangle$, $|b_1\rangle$, $|b_2\rangle$, ⋯ , $|b_{n-1}\rangle$, $|c_1\rangle$, $|c_2\rangle$, $|d\rangle$, and $|e\rangle$ are prepared. When F is the minimum integer that is $\log_2(4(n-1))$ or more, each of $|a_h\rangle$ that h is an integer from 1 to $n - 1$ is consisted of F qubits. [→ See Appendix-2] States of $|a_1\rangle$, $|a_2\rangle$, ⋯ , $|a_{n-1}\rangle$, $|b_1\rangle$, $|b_2\rangle$, ⋯ , $|b_{n-1}\rangle$, $|c_1\rangle$, $|c_2\rangle$, $|d\rangle$, and $|e\rangle$ are a_1, a_2, ⋯ , a_{n-1}, b_1, b_2, ⋯ , b_{n-1}, c_1, c_2, d, and e, respectively.

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

Step 2: The Hadamard gate $\boxed{H}$ acts on each qubit of $|a_1\rangle$, $|a_2\rangle$, ⋯ , $|a_{n-2}\rangle$, and $|a_{n-1}\rangle$ [3, 4]. It changes them for entangled states. The total states are $(2^F)^{n-1}$ [$2^F \approx 4(n-1)$]. [$|a_h\rangle$ is consisted of F qubits. Each qubit is acted on by $\boxed{H}$. Therefore, $F(n-1)$ of $\boxed{H}$ are necessary.]

Step 3: It is assumed that a quantum gate (B) changes $|b_1\rangle$, $|b_2\rangle$, ⋯ , $|b_{n-3}\rangle$, $|b_{n-2}\rangle$, and $|b_{n-1}\rangle$ for $|1 + b_1\rangle$, $|1 + b_2\rangle$, ⋯ , $|1 + b_{n-3}\rangle$, $|1 + b_{n-2}\rangle$, and $|1 + b_{n-1}\rangle$ in $a_h = 0, 1, \cdots, n - 4, n - 3$, and $n - 2$, respectively. This action repeats from $|a_1\rangle$ to $|a_{n-1}\rangle$.

As the target state for $|b_1\rangle$ is 1, quantum phase inversion gates (PI) and quantum inversion about mean gates (IM) act on $|b_1\rangle$. [Grover's database search. The same gates action is shown in the following.] [3, 6, 7] When G_1 is 2 that is $((2^F)^{n-1}/((n -$

$1)(n-2)^{n-2}4^{n-2}))^{1/2} \approx (4^{n-1}(n-1)^{n-1}/((n-1)(n-2)^{n-2}4^{n-2}))^{1/2} = (4(n-1)^{n-2}/(n-2)^{n-2})^{1/2} \approx 2$, the total number that (*PI*) and (*IM*) act on $|b_1\rangle$ is $G_1 = 2$, because they are a couple. Next, (*OB*) observes $|b_1\rangle$. Therefore, only the routes that contain 1 piece of 0 remain. The number of data is $(n-1)(n-2)^{n-2}4^{n-2}$. [Shor's data decrease. The same gate action is shown in the following.] [3, 5] [→ See Appendix-2]

As the target state for $|b_2\rangle$ is 1, (*PI*) and (*IM*) act on $|b_2\rangle$. When G_2 is 2 that is $(((n-1)(n-2)^{n-2}4^{n-2})/((n-1)(n-2)(n-3)^{n-3}4^{n-3}))^{1/2} = (4(n-2)^{n-3}/(n-3)^{n-3})^{1/2} \approx 2$, the total number that (*PI*) and (*IM*) act on $|b_2\rangle$ is $G_2 = 2$. Next, (*OB*) observes $|b_2\rangle$. Therefore, only the routes that contain 1 piece of 1 remain. The number of data is $(n-1)(n-2)(n-3)^{n-3}4^{n-3}$.

Similarly, these actions are repeated sequentially from $|b_3\rangle$ to $|b_{n-1}\rangle$ with G_i [$3 \le i \le n-1$. i is an integer.]. 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 4: It is assumed that a quantum gate (C_1) changes $|c_1\rangle$ and $|c_2\rangle$ for $|c_1 + L(0, 1 + a_1) + L(1 + a_1, 1 + a_2)\rangle$ and $|c_2 + (n-1)^{n-2}a_1 + (n-1)^{n-3}a_2\rangle$, respectively, from $|a_1\rangle$ and $|a_2\rangle$.

Similarly, (C_i) [$2 \le i \le n-3$. i is the integer.] changes $|c_1\rangle$ and $|c_2\rangle$ for $|c_1 + L(1 + a_i, 1 + a_{i+1})\rangle$ and $|c_2 + (n-1)^{n-(i+2)}a_{i+1}\rangle$, respectively, from $|a_i\rangle$ and $|a_{i+1}\rangle$. This action is repeated sequentially from $|a_2\rangle$ to $|a_{n-3}\rangle$.

(C_{n-2}) changes $|c_1\rangle$ and $|c_2\rangle$ for $|c_1 + L(1 + a_{n-2}, 1 + a_{n-1}) + L(1 + a_{n-1}, 0)\rangle$ and $|c_2 + (n-1)^0 a_{n-1}\rangle$, respectively, from $|a_{n-2}\rangle$ and $|a_{n-1}\rangle$.

Therefore, $|c_1\rangle$ and $|c_2\rangle$ become $|L_{\text{total}} = L(0, 1 + a_1) + L(1 + a_1, 1 + a_2) + \cdots + L(1 + a_{n-2}, 1 + a_{n-1}) + L(1 + a_{n-1}, 0)\rangle$ and $|U(V)\rangle$, respectively.

Step 5: It is assumed that a quantum gate (D) changes $|d\rangle$ for $|d + c_1\rangle$ in $c_1 \le M_1$, or it changes $|d\rangle$ for $|d + M_1 + c_2\rangle$ in the others of c_1.

Step 6: It is assumed that a quantum gate (E_1) doesn't changes $|e\rangle$ in $d \le M_1$ or $M_1 + U(V(Y = 1)) \le d \le M_1 + U(V(Y = ((n-1)!/4) - 2))$, or it changes $|e\rangle$ for $|e + 1\rangle$ in the others of d. As the target state for $|e\rangle$ is 0, (*PI*) and (*IM*) act on $|e\rangle$. The number of the data that is included in $d \le M_1$ or $M_1 + U(V(Y = 1)) \le d \le M_1 + U(V(Y = ((n-1)!/4) - 2))$ is $W_1 \approx (n-1)!/4$. [→ See Appendix-3] When K_1 is 2 that is $(W_0/W_1)^{1/2} \approx 2$, the total number that (*PI*) and (*IM*) act on $|e\rangle$ is $K_1 = 2$. Next, (*OB*) observes $|e\rangle$, and the data of W_1 remain.

Similarly, (E_i) [$2 \leq i \leq g - 1$. i is the integer.] doesn't change $|e\rangle$ in $d \leq M_1$ or $M_1 + U(V(Y = 1)) \leq d \leq M_1 + U(V(Y = ((n - 1)!/4^i) - 2))$, or it changes $|e\rangle$ for $|e + 1\rangle$ in the others of d. As the target state for $|e\rangle$ is 0, (PI) and (IM) act on $|e\rangle$. The number of the data that is included in $d \leq M_1$ or $M_1 + U(V(Y = 1)) \leq d \leq M_1 + U(V(Y = ((n - 1)!/4^i) - 2))$ is $W_i \approx (n - 1)!/4^i$. When K_i is 2 that is $(W_{i-1}/W_i)^{1/2} \approx 2$, the total number that (PI) and (IM) act on $|e\rangle$ is $K_i = 2$. Next, (OB) observes $|e\rangle$, and the data of W_i remain. These actions are repeated sequentially from 2 to $g - 1$ at i.

(E_g) doesn't change $|e\rangle$ in $d \leq M_1$, or it changes $|e\rangle$ for $|e + 1\rangle$ in the others of d. As the target state for $|e\rangle$ is 0, (PI) and (IM) act on $|e\rangle$. The number of the data that is included in $d \leq M_1$ is $W_g \approx 2$. When K_g is 2 that is $(W_{g-1}/W_g)^{1/2} \approx 2$, the total number that (PI) and (IM) act on $|e\rangle$ is $K_g = 2$. Next, (OB) observes $|a_1\rangle, |a_2\rangle, \cdots, |a_{n-1}\rangle, |b_1\rangle, |b_2\rangle, \cdots, |b_{n-1}\rangle, |c_1\rangle, |c_2\rangle, |d\rangle$, and $|e\rangle$, and one of the data of W_g remains.

Therefore, one example of routes that are $L_{\text{total}} \leq M_1$ is obtained.

Step 7: When the state of $|e\rangle$ 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 7 are repeated. It is assumed that the minimum distance $M_{\min}$ obtains by repeating about $\log_2 (n - 1)!$ [12].

An example is shown as the next section. However, this algorithm is applied as far as the effect of Grover's database search and Shor's data decrease.

4. NUMERICAL COMPUTATION

It is assumed that there are $n = 10$, $P_0(0, 0)$, $P_1(1, -2)$, $P_2(3, -1)$, $P_3(4, 1)$, $P_4(2, 3)$, $P_5(1, -1)$, $P_6(3, -2)$, $P_7(4, 0)$, $P_8(0, 1)$, $P_9(2, 2)$, $L(0, 2) \approx 3.2$, $L(0, 5) \approx 1.4$, $L(0, 8) = 1$, $L(0, 3) \approx 4.1$, $L(0, 7) = 4$, $L(1, 5) = 1$, $L(1, 4) \approx 5.1$, $L(2, 6) = 1$, $L(3, 7) = 1$, $L(3, 1) \approx 4.2$, $L(3, 9) \approx 2.2$, $L(4, 9) = 1$, $L(4, 6) \approx 5.1$, $L(4, 2) \approx 4.1$, $L(5, 9) \approx 3.2$, $L(6, 1) = 2$, $L(6, 8) \approx 4.2$, $L(7, 2) \approx 1.4$, $L(7, 5) \approx 3.2$, $L(7, 8) \approx 4.1$, $L(8, 4) \approx 2.8$, $L(8, 7) \approx 4.1$, $L(8, 3) = 4$, $L(9, 3) \approx 2.2$, $L(9, 2) \approx 3.2$, $L(9, 4) = 1$ [The value of the others of $L(i, j)$ is $10M_1$.], $g = 9$ [$9!/2 = 181440 \leq 4^9 = 262144$], $U(V(Y = 1)) = 6053444$, $U(V(Y = (9!/4) - 2 = 90718)) = 95584572$ [for example, $Y = 90718 = 2{\cdot}8! + 1{\cdot}7! + 6{\cdot}6! + 5{\cdot}5! + 4{\cdot}4! + 3{\cdot}3! + 2{\cdot}2! + 0{\cdot}1!$, $U(V(Y = 90718)) = 95584572 = 2{\cdot}9^8 + 1{\cdot}9^7 + 8{\cdot}9^6 + 7{\cdot}9^5 + 6{\cdot}9^4 + 5{\cdot}9^3 + 3{\cdot}9^2 + 4{\cdot}9^1 + 0{\cdot}9^0$], [→ See Appendix-1] $U(V(Y = (9!/16) - 2 = 22678)) = 26275564$, $U(V(Y = (9!/64) - 2 = 5668)) = 10598756$, $U(V(Y = (9!/256) - 2 \approx 1416)) = 6894596$, $U(V(Y = (9!/1024) - 2 \approx 352)) = 6198348$, $U(V(Y = (9!/4096) - 2 \approx 87)) =$

6073748, $U(V(Y = (9!/16384) - 2 \approx 20)) = 6055548$, $U(V(Y = (9!/65536) - 2 \approx 4)) = 6053532$, and $M_1 = 20$.

First of all, $|a_1\rangle, |a_2\rangle, \cdots, |a_9\rangle, |b_1\rangle, |b_2\rangle, \cdots, |b_9\rangle, |c_1\rangle, |c_2\rangle, |d\rangle$, and $|e\rangle$ are prepared. When F is the minimum integer that is $\log_2(4(n-1)) = \log_2(4\cdot 9) \approx 5.170 \leq 6 = F$, each of $|a_h\rangle$ that h is the integer from 1 to 9 is consisted of $F = 6$ qubits. States of $|a_1\rangle, |a_2\rangle, \cdots, |a_9\rangle, |b_1\rangle, |b_2\rangle, \cdots, |b_9\rangle, |c_1\rangle, |c_2\rangle, |d\rangle$, and $|e\rangle$ are $a_1, a_2, \cdots, a_9, b_1, b_2, \cdots, b_9, c_1, c_2, d$, and e, respectively.

Step 1: Each qubit of $|a_1\rangle, |a_2\rangle, \cdots, |a_9\rangle, |b_1\rangle, |b_2\rangle, \cdots, |b_9\rangle, |c_1\rangle, |c_2\rangle, |d\rangle$, and $|e\rangle$ is set $|0\rangle$.

Step 2: $\boxed{H}$ acts on each qubit of $|a_1\rangle, |a_2\rangle, \cdots, |a_8\rangle$, and $|a_9\rangle$. It changes them for entangled states. The total states are $(2^F)^{n-1} = (2^6)^9 = 64^9$.

Step 3: (B) changes $|b_1\rangle, |b_2\rangle, \cdots, |b_8\rangle$, and $|b_9\rangle$ for $|1 + b_1\rangle, |1 + b_2\rangle, \cdots, |1 + b_8\rangle$, and $|1 + b_9\rangle$ in $a_h = 0, 1, \cdots, 7$, and 8, respectively. This action repeats from $|a_1\rangle$ to $|a_9\rangle$.

As the target state for $|b_1\rangle$ is 1, (PI) and (IM) act on $|b_1\rangle$. When G_1 is 2 that is $(4(n-1)^{n-2}/(n-2)^{n-2}))^{1/2} = (4\cdot 9^8/8^8)^{1/2} = 2(9/8)^4 \approx 2$, the total number that (PI) and (IM) act on $|b_1\rangle$ is $G_1 = 2$. Next, (OB) observes $|b_1\rangle$. Therefore, only the routes that contain 1 piece of 0 remain. The number of data is $(n-1)(n-2)^{n-2}4^{n-2} = 9\cdot 8^8\cdot 4^8$.

As the target state for $|b_2\rangle$ is 1, (PI) and (IM) act on $|b_2\rangle$. When G_2 is 2 that is $(4(n-2)^{n-3}/(n-3)^{n-3})^{1/2} = 2(8/7)^{7/2} \approx 2$, the total number that (PI) and (IM) act on $|b_2\rangle$ is $G_2 = 2$. Next, (OB) observes $|b_2\rangle$. Therefore, only the routes that contain 1 piece of 1 remain. The number of data is $(n-1)(n-2)(n-3)^{n-3}4^{n-3} = 9\cdot 8\cdot 7^7\cdot 4^7$.

Similarly, these actions are repeated sequentially from $|b_3\rangle$ to $|b_9\rangle$ with G_i [$3 \leq i \leq n-1 = 9$. i is the integer.]. 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 4: (C_1) changes $|c_1\rangle$ and $|c_2\rangle$ for $|c_1 + L(0, 1 + a_1) + L(1 + a_1, 1 + a_2)\rangle$ and $|c_2 + 9^8 a_1 + 9^7 a_2\rangle$, respectively, from $|a_1\rangle$ and $|a_2\rangle$.

Similarly, (C_i) [$2 \leq i \leq 7$. i is the integer.] changes $|c_1\rangle$ and $|c_2\rangle$ for $|c_1 + L(1 + a_i, 1 + a_{i+1})\rangle$ and $|c_2 + 9^{10-(i+2)} a_{i+1}\rangle$, respectively, from $|a_i\rangle$ and $|a_{i+1}\rangle$. This action is repeated sequentially from $|a_2\rangle$ to $|a_7\rangle$.

(C_8) changes $|c_1\rangle$ and $|c_2\rangle$ for $|c_1 + L(1 + a_8, 1 + a_9) + L(1 + a_9, 0)\rangle$ and $|c_2 + 9^0 a_9\rangle$, respectively, from $|a_8\rangle$ and $|a_9\rangle$.

Therefore, $|c_1\rangle$ and $|c_2\rangle$ become $|L_{\text{total}} = L(0, 1 + a_1) + L(1 + a_1, 1 + a_2) + \cdots + L(1 + a_8, 1 + a_9) + \text{L}(1 + a_9, 0)\rangle$ and $|U(V)\rangle$, respectively.

Step 5: (D) changes $|d\rangle$ for $|d + c_1\rangle$ in $c_1 \leq M_1 = 20$, or it changes $|d\rangle$ for $|d + 20 + c_2\rangle$ in the others of c_1.

Step 6: (E_1) doesn't change $|e\rangle$ in $d \leq M_1 = 20$ or $M_1 + U(V(Y = 1)) = 20 + 6053444 = 6053464 \leq d \leq M_1 + U(V(Y = ((n - 1)!/4) - 2 = (9!/4) - 2 = 90718)) = 20 + 95584572 = 95584592$, or it changes $|e\rangle$ for $|e + 1\rangle$ in the others of d. As the target state for $|e\rangle$ is 0, (PI) and (IM) act on $|e\rangle$. The number of the data that is included in $d \leq 20$ or $6053464 \leq d \leq 95584592$ is $W_1 \approx 9!/4$. When K_1 is 2 that is $(W_0/W_1)^{1/2} \approx (9!/(9!/4))^{1/2} = 2$, the total number that (PI) and (IM) act on $|e\rangle$ is $K_1 = 2$. Next, (OB) observes $|e\rangle$, and the data of W_1 remain.

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

(E_9) doesn't changes $|e\rangle$ in $d \leq 20$, or it changes $|e\rangle$ for $|e + 1\rangle$ in the others of d. As the target state for $|e\rangle$ is 0, (PI) and (IM) act on $|e\rangle$. The number of the data that is included in $d \leq 20$ is $W_9 \approx 2$. When K_9 is 2 that is $(W_8/W_9)^{1/2} \approx ((9!/4^8)/(9!/4^9))^{1/2} = 2$, the total number that (PI) and (IM) act on $|e\rangle$ is $K_9 = 2$. Next, (OB) observes $|a_1\rangle$, $|a_2\rangle$, $|a_3\rangle$, $|a_4\rangle$, $|a_5\rangle$, $|a_6\rangle$, $|a_7\rangle$, $|a_8\rangle$, $|a_9\rangle$, $|b_1\rangle$, $|b_2\rangle$, $\cdots$, $|b_9\rangle$, $|c_1\rangle$, $|c_2\rangle$, $|d\rangle$, and $|e\rangle$, 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_9, c_1, c_2, d, and e are 4, 0, 5, 1, 6, 2, 7, 3, 8, 1, 1, $\cdots$, 1, 18, $U(V(Y = 163491)) = 174944564$, 18, and 0, respectively.

Step 7: In the example, the state of $|e\rangle$ is 0. Therefore, M_1 is assumed to be $M_2 = 15$ [$< 18 < M_1 = 20$], and these calculations from step 1 to step 7 are repeated. It is assumed that the state of $|e\rangle$ is 0. When the states of $|e\rangle$ is 1 at $M_3 = 10$, $M_4 = 13$, and $M_5 = 14$, 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_9, c_1, c_2, d, and e are 4, 0, 5, 1, 6, 2, 8, 3, 7, 1, 1, $\cdots$, 1, 15, $U(V(Y = 163493)) = 174944644$, 15, and 0, respectively. As a result, the shortest route $P_0 \to P_5 \to P_1 \to P_6 \to P_2 \to P_7 \to P_3 \to P_9 \to P_4 \to P_8 \to$

P_0 is obtained. And then, $P_0 \to P_8 \to P_4 \to P_9 \to P_3 \to P_7 \to P_2 \to P_6 \to P_1 \to P_5 \to P_0$ is the another answer.

5. DISCUSSION AND SUMMARY

In the example of section 4, the computational complexity of this quantum algorithm [= S] is 680. The computational complexity of the classical computation [= Z] is $(n - 1)!/2 = 9!/2 = 181440$. After all, S/Z becomes about 1/267.

In general, S becomes the following. In the order of the actions by the gates, the number of them is $F(n - 1)$ at $\boxed{\mathrm{H}}$, $n - 1$ at (B), $\Sigma_{i=1 \to n-1} G_i = 2(n - 1)$ at (PI) and (IM), $(n - 1)$ at (OB), $n - 1$ at (C_i) [$1 \le i \le n - 1$. i is the integer.], 1 at (D), g at (E_i) [$1 \le i \le$ g. i is the integer.], $\Sigma_{i=1 \to g} K_i = 2g$ at (PI) and (IM), and g at (OB). These processes repeated about $\log_2 (n - 1)!$. Therefore, S becomes $(F(n - 1) + 7n - 6 + 4g)\log_2 (n - 1)!$.

When n is large enough, S becomes about $3(\log_2 (n - 1))^2(n - 1)^2$, where F is about $\log_2 (4(n - 1))$, 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 decreased process becomes possible.

I hope that this result will be confirmed by many experiments.

APPENDIX-1 [13]

$U[X]$ [X is the number of datum.] is the number of the repeated permutation of $(n - 1)^{n-1}$ type, and $V(Y)$ [Y is the number of datum.] is the number of permutation of $(n - 1)!$ type.

Therefore, in $(n - 1)!$ type, the numbers of permutation from first to $((n - 1)!/4) - 2$ are, in $(n - 1)^{n-1}$ type, the numbers of permutation from $U(V(Y = 1))$ to $U(V(Y = ((n - 1)!/4) - 2))$, because a number of combinations of answers is at least 2. The order of numbers of permutation from 1 to about $(n - 1)!/4$ converged the probability amplitudes by the Appendix-3. And then, this process is repeated.

Where, the examples of $U[X] = U(V(Y))$ are shown at the section 4.

Example-1:

$Y = 1 = (v_1 = 0)8! + (v_2 = 0)7! + (v_3 = 0)6! + (v_4 = 0)5! + (v_5 = 0)4! + (v_6 = 0)3! + (v_7 = 0)2! + (v_8 = 1)1!$.

Therefore, 0, 1, 2, 3, 4, 5, 6, 7, 8 ⇒ from (I), $v_1 = v_2 = v_3 = v_4 = v_5 = v_6 = v_7 = 0 \rightarrow t_1 = 0$, $t_2 = 1$, $t_3 = 2$, $t_4 = 3$, $t_5 = 4$, $t_6 = 5$, $t_7 = 6$; 7, 8 ⇒ from (IV), $v_8 = 1 \rightarrow t_8 = 7$, $t_9 = 8$.

And then, $U(V(Y = 1)) = (t_1 = 0)9^8 + (t_2 = 1)9^7 + (t_3 = 2)9^6 + (t_4 = 3)9^5 + (t_5 = 4)9^4 + (t_6 = 5)9^3 + (t_7 = 6)9^2 + (t_8 = 7)9^1 + (t_9 = 8)9^0 = 6053444$.

Example-2:

$Y = (9!/4) - 2 = 90718 = (v_1 = 2)8! + (v_2 = 1)7! + (v_3 = 6)6! + (v_4 = 5)5! + (v_5 = 4)4! + (v_6 = 3)3! + (v_7 = 2)2! + (v_8 = 0)1!$.

Therefore, 0, 1, 2, 3, 4, 5, 6, 7, 8 ⇒ from (III), $v_1 = 2 \rightarrow t_1 = 2$; 0, 1, 3, 4, 5, 6, 7, 8 ⇒ from (III), $v_2 = 1 \rightarrow t_2 = 1$; 0, 3, 4, 5, 6, 7, 8 ⇒ from (III), $v_3 = 6 \rightarrow t_3 = 8$; 0, 3, 4, 5, 6, 7 ⇒ from (III), $v_4 = 5 \rightarrow t_4 = 7$; 0, 3, 4, 5, 6 ⇒ from (III), $v_5 = 4 \rightarrow t_5 = 6$; 0, 3, 4, 5 ⇒ from (III), $v_6 = 3 \rightarrow t_6 = 5$; 0, 3, 4 ⇒ from (II), $v_7 = 2$, $v_8 = 0 \rightarrow t_7 = 3$, $t_8 = 4$, $t_9 = 0$.

And then, $U(V(Y = (9!/4) - 2 = 90718)) = (t_1 = 2)9^8 + (t_2 = 1)9^7 + (t_3 = 8)9^6 + (t_4 = 7)9^5 + (t_5 = 6)9^4 + (t_6 = 5)9^3 + (t_7 = 3)9^2 + (t_8 = 4)9^1 + (t_9 = 0)9^0 = 95584572$.

APPENDIX-2

It is assumed that the state of $|b_i\rangle$ is 1, and there is $\log_2 (4k) \leq F$. [$\rightarrow 4k \approx 2^F$] When the probability amplitudes of state of 1 are marked a minus, the mean of probability amplitudes becomes $((2^F)^{-1/2}(2^F - k) - (2^F)^{-1/2}k)/2^F = (1 - (2k/2^F))(2^F)^{-1/2} \approx (1/2)(4k)^{-1/2}$.

When the inversion about mean is practiced, the probability amplitudes of state of 1 are $-(-(2^F)^{-1/2}) + (1 - (2k/2^F))(2^F)^{-1/2}\times 2 = (3 - (4k/2^F))(2^F)^{-1/2} \approx 2(4k)^{-1/2}$, and the probability amplitude of state of 0 are

$$(2^F)^{-1/2} - ((2^F)^{-1/2} - (1 - (2k/2^F))(2^F)^{-1/2})\times 2 = (1 - (4k/2^F))(2^F)^{-1/2} \approx 0.$$

Therefore, the sum of square of probability amplitude is

$$((3 - (4k/2^F))(2^F)^{-1/2})^2 k + ((1 - (4k/2^F))(2^F)^{-1/2})^2(2^F - k) \approx 4(4k)^{-1}k + 0^2(4k - k) = 1.$$

After all, the data of state of 1 [$(4k)/4 \rightarrow k$] remain [3, 6, 7, 10]. [This is a quarter method-1.]

APPENDIX-3

It is assumed that the number of data is N, the value of data of $N/4$ is R, and values of data of $3N/4$ are the others. When the probability amplitudes of data of R are marked a minus, the mean of probability amplitudes becomes

$$(N^{-1/2}(3N/4) - N^{-1/2}(N/4))/N = (1/2)N^{-1/2}.$$

When the inversion about mean is practiced, the probability amplitudes of data of R are $-(-N^{-1/2}) + (1/2)N^{-1/2}\times 2 = 2N^{-1/2}$, and the probability amplitude of data of others are $N^{-1/2} - (N^{-1/2} - (1/2)N^{-1/2})\times 2 = 0$.

Therefore, the sum of square of probability amplitude is

$$(2N^{-1/2})^2(1/4)N + 0^2(3/4)\mathrm{N} = 1 + 0 = 1.$$

After all, the data of $N/4$ of R remain [3, 6, 7, 10, 11, 13]. [This is a quarter method-2.]

When this process is repeated, the number of data decreases and the probability amplitudes of necessary data increase.

REFERENCES

[1] Ono, K., Mori, T., and Moriyama, S., 2019, "High-temperature operation of a silicon qubit," Sci. Rep., **9**, 469, DOI: 10.1038/ s41598-018-36476-z.

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

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

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

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

[6] Grover, L. K., 1996, "A fast quantum mechanical algorithm for database

search," Proc. 28th Annu. ACM Symp. Theory of Computing, pp.212-219.

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

[8] Ambainis, A., 2008, "Quantum walks and their algorithmic applications," http:// arXiv. org/ quant-ph/ arXiv: quant-ph/ 0403120v3.

[9] Bennett, C. H., Bernstein, E., Brassard, G., and Vazirani, U., 1996, "Strengths and weaknesses of quantum computing," http:// arXiv. org/ quant-ph/ arXiv: quant-ph/ 9701001v1.

[10] Fujimura, T., 2020, "Quantum algorithm for maximum integer multiple-choice generalized knapsack problem by hybrid method of Grover's database search and Shor's data decrease," Glob. J. Pure Appl. Math., **16**, 643-651.

[11] Fujimura, T., 2013, "Quantum algorithm for traveling salesman problem by numbering method," Glob. J. Pure Appl. Math., **9**, 545-551.

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

[13] Fujimura, T., 2020, "Quantum algorithm for longest path problem by hybrid method of Grover's database search and Shor's data decrease," Adva. Theo. Appl. Math., **15**, 1-10.