Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 17, Number 1 (2021), pp. 9-15
© Research India Publications
http://www.ripublication.com/gjpam.htm

Quantum Algorithm for Bin-Packing 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 bin-packing problem by a quarter method and its example are reported. It is decided whether n pieces of luggage are packed into k boxes, where each weight of the luggage is M or less, and the maximum storage weight of the box is M. A computational complexity of a classical computation is k^n. The computational complexity becomes about $3(\log_2 k)n$ by this quantum algorithm. Therefore, a decreased process becomes possible.

Keywords: Quantum algorithm, bin-packing problem, quarter method, computational complexity, decreased process.

AMS subject classification: Primary 81-08; Secondary 68R05, 68W40.

1. INTRODUCTION

Kudo reported localization phenomena in the constrained quantum annealing of graph coloring from the viewpoint of analogy to a tightbinding chain under effective fields [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 traveling salesman problem are converged quickly by a quarter method [10]. Its computational complexity is decreased. The bin-packing problem [11] is examined by the quarter method this time. Therefore, its result is reported.

2. BIN-PACKING PROBLEM

It is decided whether n pieces of luggage are packed into k boxes, where each weight of the luggage is M or less, and the maximum storage weight of the box is M [11].

3. QUANTUM ALGORITHM

It is assumed that there are n pieces of luggage and k boxes, where each weight of the luggage is M or less, the maximum storage weight of the box is M, each weight of luggage is x_i [$1 \leq i \leq n$. i is an integer.], and a_i [$1 \leq i \leq n$. i is an integer.] is 0 or 1 or $\cdots$ or $k - 2$ or $k - 1$. When the number of the n times repeated permutation of 0, 1, $\cdots$, $k - 2$, and $k - 1$ is k^n [$= W_0$], $a_1k^{n-1} + a_2k^{n-2} + \cdots + a_nk^0 = \Sigma_{i=1\to n}\, a_ik^{n-i} = U[X]$ [X is the number of datum.] is the numbering datum from 0 to $k^n - 1$ [For example, $U[X = 0]$ is $a_1 = 0$, $a_2 = 0$, $\cdots$, $a_{n-1} = 0$, and $a_n = 0$, and $U[X = k^n - 1]$ is $a_1 = k - 1$, $a_2 = k - 1$, $\cdots$, $a_{n-1} = k - 1$, and $a_n = k - 1$.]. g is the minimum integer that follows $k^n/k! \leq 4^g = 2^{2g}$, because a number of combinations of an answer is at least $k!$. $U[X = 0] = 0$, $U[X = (W_0/4) - 1 - k!]$, $U[X = (W_0/16) - 1 - k!]$, $\cdots$, $U[X = (W_0/4^{g-1}) - 1 - k!]$, and $U[X = (W_0/4^g) - k!] \approx 0$ are computed. [→ See Appendix-1] Next, a quantum algorithm is shown as the following.

First of all, quantum registers $|a_1\rangle$, $|a_2\rangle$, $\cdots$, $|a_n\rangle$, $|b_1\rangle$, $|b_2\rangle$, $|c_1\rangle$, $|c_2\rangle$, $\cdots$, $|c_k\rangle$, $|d\rangle$, $|e_1\rangle$, and $|e_2\rangle$ are prepared. When F is the minimum integer that is $\log_2(4k)$ or more, each of $|a_f\rangle$ that f is an integer from 1 to n is consisted of F qubits. [→ See Appendix-2] States of $|a_f\rangle$, $|b_1\rangle$, $|b_2\rangle$, $|c_1\rangle$, $|c_2\rangle$, $\cdots$, $|c_k\rangle$, $|d\rangle$, $|e_1\rangle$, and $|e_2\rangle$ are a_f, b_1, b_2, c_1, c_2, $\cdots$, c_k, d, e_1, and e_2, respectively.

Step 1: Each qubit of $|a_f\rangle$, $|b_1\rangle$, $|b_2\rangle$, $|c_1\rangle$, $|c_2\rangle$, $\cdots$, $|c_k\rangle$, $|d\rangle$, $|e_1\rangle$, and $|e_2\rangle$ is set $|0\rangle$.

Step 2: The Hadamard gate H [3, 4] acts on each qubit of $|a_f\rangle$. It changes them for entangled states. The total states are $(2^F)^n$. [$2^F \approx 4k$] [$|a_f\rangle$ is consisted of F qubits. Each qubit is acted on by H. Therefore, Fn of H are necessary.]

Step 3: It is assumed that a quantum gate (A) changes $|b_1\rangle$ for $|1\rangle$ in $a_f < k$, or it changes $|b_1\rangle$ for $|0\rangle$ in the others of a_f, it changes $|b_2\rangle$ for $|b_2 + a_f k^{n-f}\rangle$ at $|a_f\rangle$, and it changes $|c_h\rangle$ [$1 \le h \le k$. h is an integer.] for $|c_h + x_f\rangle$ at $a_f = h - 1$. As a target state for $|b_1\rangle$ is 1, quantum phase inversion gates (PI) and quantum inversion about mean gates (IM) [3, 6, 7] act on $|b_1\rangle$. [Grover's database search. The same gates action is shown in the following.] [3, 6, 7]. When G is $(2^F/k)^{1/2} \approx (4k/k)^{1/2} = 2$, the total number that (PI) and (IM) act on $|b_1\rangle$ is $G = 2$, because they are a couple. Next, an observation gate (OB) observes $|b_1\rangle$. [Shor's data decrease. The same gate action is shown in the following.] [3, 5]. [→ See Appendix-2] These actions are repeated sequentially from $|a_1\rangle$ to $|a_n\rangle$. Therefore, each state of $|a_f\rangle$ is 0, 1, ⋯ , $k-2$, and $k-1$, and the total states become k^n [= W_0].

Step 4: It is assumed that a quantum gate (B) changes $|d\rangle$ for $|d + 1\rangle$ in $c_h \le M$, or it doesn't change $|d\rangle$ in the others of c_h. These actions are repeated sequentially from $|c_1\rangle$ to $|c_h\rangle$.

Step 5: It is assumed that a quantum gate (C) doesn't changes $|e_1\rangle$ at $d = k$, or it changes $|e_1\rangle$ for $|e_1 + 1 + b_2\rangle$ in the others of d.

Step 6: It is assumed that a quantum gate (D_1) changes $|e_2\rangle$ for $|1\rangle$ in $U[X = 0] = 0 \le e_1 \le U[X = (k^n/4) - 1 - k!] = (k^n/4) - 1 - k!$, or it changes $|e_2\rangle$ for $|0\rangle$ in the others of e_1. As the target state for $|e_2\rangle$ is 1, (PI) and (IM) act on $|e_2\rangle$. The number of the data that is included in $0 \le e_1 \le (k^n/4) - 1 - k!$ is $W_1 \approx k^n/4$. [→ See Appendix-1] When L_1 is $(W_0/W_1)^{1/2} \approx (k^n/(k^n/4))^{1/2} = 2$, the total number that (PI) and (IM) act on $|e_2\rangle$ is $L_1 = 2$. Next, (OB) observes $|e_2\rangle$, and the data of W_1 remain.

Similarly, (D_i) [$2 \le i \le g - 1$. i is the integer.] changes $|e_2\rangle$ for $|1\rangle$ in $0 \le e_1 \le U[X = (W_0/4^i) - 1 - k!] = (k^n/4^i) - 1 - k!$, or it changes $|e_2\rangle$ for $|0\rangle$ in the others of e_1. As the target state for $|e_2\rangle$ is 1, (PI) and (IM) act on $|e_2\rangle$. The number of the data that is included in $0 \le e_1 \le (k^n/4^i) - 1 - k!$ is $W_i \approx W_0/4^i = k^n/4^i$. When L_i is $(W_{i-1}/ W_i)^{1/2} \approx ((k^n/4^{i-1})/(k^n/4^i))^{1/2} = 2$, the total number that (PI) and (IM) act on $|e_2\rangle$ is $L_i = 2$. Next, (OB) observes $|e_2\rangle$, and the data of W_i remain. These actions are repeated sequentially from 2 to $g - 1$ at i.

(D_g) changes $|e_2\rangle$ for $|1\rangle$ at $e_1 = 0$ [$0 \le e_1 \le U[X = (W_0/4^g) - 1 - k!] = (k^n/4^g) - 1 - k! \approx 0$], or it changes $|e_2\rangle$ for $|0\rangle$ in the others of e_1. As the target state for $|e_2\rangle$ is 1, (PI) and (IM) act on $|e_2\rangle$. The number of the data that is included at $e_1 = 0$ is $W_g \approx k! \approx k^n/4^g$. When L_g is $(W_{g-1}/W_g)^{1/2} \approx ((k^n/4^{g-1})/(k^n/4^g))^{1/2} = 2$, the total number that (PI) and (IM) act on $|e_2\rangle$ is $L_g = 2$. Next, (OB) observes $|a_f\rangle$, $|b_1\rangle$, $|b_2\rangle$, $|c_h\rangle$, $|d\rangle$, $|e_1\rangle$, and $|e_2\rangle$, and one of the data of W_g remains. Therefore, one example of combinations that are $c_h \le M$ is obtained.

4. NUMERICAL COMPUTATION

It is assumed that there are $n = 6$, $x_1 = 3$, $x_2 = 2$, $x_3 = 5$, $x_4 = 1$, $x_5 = 6$, $x_6 = 4$, $k = 3$, $M = 7$, and $g = 4$ [$3^6/3! = 726/6 = 121.5 \le 4^g = 4^4 = 256$].

First of all, quantum registers $|a_1\rangle$, $|a_2\rangle$, $\cdots$, $|a_6\rangle$, $|b_1\rangle$, $|b_2\rangle$, $|c_1\rangle$, $|c_2\rangle$, $|c_3\rangle$, $|d\rangle$, $|e_1\rangle$, and $|e_2\rangle$ are prepared. When F is the minimum integer that is $\log_2 (4k) = \log_2 (4{\cdot}3) \approx 2 + 1.6 = 3.6 \le 4$, each of $|a_f\rangle$ that f is an integer from 1 to 6 is consisted of 4 qubits. States of $|a_1\rangle$, $|a_2\rangle$, $\cdots$, $|a_6\rangle$, $|b_1\rangle$, $|b_2\rangle$, $|c_1\rangle$, $|c_2\rangle$, $|c_3\rangle$, $|d\rangle$, $|e_1\rangle$, and $|e_2\rangle$ are a_1, a_2, $\cdots$, a_6, b_1, b_2, c_1, c_2, c_3, d, e_1, and e_2, respectively.

Step 1: Each qubit of $|a_f\rangle$, $|b_1\rangle$, $|b_2\rangle$, $|c_1\rangle$, $|c_2\rangle$, $|c_3\rangle$, $|d\rangle$, $|e_1\rangle$, and $|e_2\rangle$ is set $|0\rangle$.

Step 2: $\boxed{\mathrm{H}}$ acts on each qubit of $|a_f\rangle$. It changes them for entangled states. The total states are $(2^4)^6$. [$2^4 \approx 4{\cdot}3$] [$|a_f\rangle$ is consisted of 4 qubits. Each qubit is acted on by $\boxed{\mathrm{H}}$. Therefore, Fn of $\boxed{\mathrm{H}}$ are necessary.]

Step 3: (A) changes $|b_1\rangle$ for $|1\rangle$ in $a_f < k$, or it changes $|b_1\rangle$ for $|0\rangle$ in the others of a_f, it changes $|b_2\rangle$ for $|b_2 + a_f 3^{6-f}\rangle$ at $|a_f\rangle$, and it changes $|c_h\rangle$ [$1 \le h \le 3$. h is an integer.] for $|c_h + x_f\rangle$ at $a_f = h - 1$. As a target state for $|b_1\rangle$ is 1, (PI) and (IM) act on $|b_1\rangle$. When G is $(2^F/k)^{1/2} = (2^4/3)^{1/2} \approx (4{\cdot}3/3)^{1/2} = 2$, the total number that (PI) and (IM) act on $|b_1\rangle$ is $G = 2$. Next, (OB) observes $|b_1\rangle$. These actions are repeated sequentially from $|a_1\rangle$ to $|a_6\rangle$. Therefore, each state of $|a_f\rangle$ is 0, 1, or 2, and the total states become 3^6 [$= W_0$].

Step 4: (B) changes $|d\rangle$ for $|d + 1\rangle$ in $c_h \le 7$, or it doesn't change $|d\rangle$ in the others of c_h. These actions are repeated sequentially from $|c_1\rangle$ to $|c_3\rangle$.

Step 5: (C) doesn't changes $|e_1\rangle$ at $d = 3$, or it changes $|e_1\rangle$ for $|e_1 + 1 + b_2\rangle$ in the others of d.

Step 6: (D_1) changes $|e_2\rangle$ for $|1\rangle$ in $0 \le e_1 \le U[X = (k^n/4) - 1 - k!] = (3^6/4) - 1 - 6$, or it changes $|e_2\rangle$ for $|0\rangle$ in the others of e_1. As the target state for $|e_2\rangle$ is 1, (PI) and (IM) act on $|e_2\rangle$. The number of the data that is included in $0 \leq e_1 \leq (3^6/4) - 1 - 6$ is $W_1 \approx 3^6/4$. When L_1 is $(W_0/W_1)^{1/2} \approx (3^6/(3^6/4))^{1/2} = 2$, the total number that (*PI*) and (*IM*) act on $|e_2\rangle$ is $L_1 = 2$. Next, (*OB*) observes $|e_2\rangle$, and the data of W_1 remain.

Similarly, (D_i) [$2 \leq i \leq 4 - 1 = 3$. i is the integer.] changes $|e_2\rangle$ for $|1\rangle$ in $0 \leq e_1 \leq U[X = (W_0/4^i) - 1 - 6] = (3^6/4^i) - 1 - 6$, or it changes $|e_2\rangle$ for $|0\rangle$ in the others of e_1. As the target state for $|e_2\rangle$ is 1, (*PI*) and (*IM*) act on $|e_2\rangle$. The number of the data that is included in $0 \leq e_1 \leq (3^6/4^i) - 1 - 6$ is $W_i \approx 3^6/4^i$. When L_i is $(W_{i-1}/ W_i)^{1/2} \approx ((3^6/4^{i-1})/(3^6/4^i))^{1/2} = 2$, the total number that (*PI*) and (*IM*) act on $|e_2\rangle$ is $L_i = 2$. Next, (*OB*) observes $|e_2\rangle$, and the data of W_i remain. These actions are repeated sequentially from 2 to 3 at i.

(D_4) changes $|e_2\rangle$ for $|1\rangle$ at $e_1 = 0$, or it changes $|e_2\rangle$ for $|0\rangle$ in the others of e_1. As the target state for $|e_2\rangle$ is 1, (*PI*) and (*IM*) act on $|e_2\rangle$. The number of the data that is included at $e_1 = 0$ is $W_4 \approx k! = 6 \approx k^n/4^g = 3^6/4^4$. When L_4 is $(W_3/W_4)^{1/2} \approx ((3^6/4^3)/(3^6/4^4))^{1/2} = 2$, the total number that (*PI*) and (*IM*) act on $|e_2\rangle$ is $L_4 = 2$. Next, (*OB*) observes $|a_1\rangle, |a_2\rangle, \cdots, |a_6\rangle, |b_1\rangle, |b_2\rangle, |c_1\rangle, |c_2\rangle, |c_3\rangle, |d\rangle, |e_1\rangle$, and $|e_2\rangle$, and one of the data of W_4 remains. For example, when $a_1, a_2, a_3, a_4, a_5, a_6, b_1, b_2, c_1, c_2, c_3, d, e_1$, and e_2 are 0, 1, 1, 2, 2, 0, 1, 132, 7, 7, 7, 3, 0, and 1, respectively, it is obtained that 3 combinations that are (1, 6), (2, 5), and (3, 4).

5. DISCUSSION AND SUMMARY

The computational complexity of this quantum algorithm [= S] becomes the following. In the order of the actions by the gates, the number of them is Fn at $\boxed{\text{H}}$, n at (A), $Gn = 2n$ at (*PI*) and (*IM*), n at (*OB*), k at (B), 2 at (C), g at (D_i) [$1 \leq i \leq g$. i is the integer.], $\Sigma_{i=1\to g} L_i = 2g$ at (*PI*) and (*IM*), and g at (*OB*). Therefore, S becomes $(F + 4)n + k + 2 + 4g$. In the example of the numerical computation at section 4, S is 69. The computational complexity of the classical computation [= Z] is $k^n = 3^6 = 729$. After all, S/Z becomes about 1/11. When n is large enough, S becomes about $3(\log_2 k)n$, where F is about $\log_2 (4k)$, g is about $(1/2)\log_2 (k^n/k!) \approx (n/2)\log_2 k$, and $k!$ is about $k^k e^{-k}(2k)^{1/2}$ [Stirling's formula]. And then, S/Z is about $3(\log_2 k)n/k^n \approx n/k^n$. For example, as for $n = 100$ and $k = 4$, S/Z is about $1/10^{58}$. Therefore, a decreased process becomes possible.

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

APPENDIX-1

It is assumed that the number of data is N, the value of data of $N/4$ is Y, and values of data of $3N/4$ are the others. When the probability amplitudes of data of Y 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 Y 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

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

After all, the data of $N/4$ of Y remain [3, 6, 7, 10]. [→ This is a quarter method-1.]

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

APPENDIX-2

It is assumed that the state of $|b_1\rangle$ is 1, and there is $\log_2(4k) \le F$. [→ $4k \approx 2^F$] 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

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

After all, the data of state of 1 [$(4k)/4 \to k$] remain [3, 6, 7, 10]. [→ This is the quarter method-2.]

REFERENCES

[1] Kudo, K., 2020, “Localization in the constrained quantum annealing of graph coloring,” J. Phys. Soc. Jpn., **89**, 064001, https:// doi. org/ 10. 7566/ JPSJ. 89. 064001.

[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 traveling salesman problem by quarter method,” Glob. J. Pure Appl. Math., **16**, 711-721.

[11] Fujimura, T., 2014, “Quantum algorithm for bin-packing problem by numbering method,” Glob. J. Pure Appl. Math., **10**, 63-66.