Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 8, Number 2 (2012), pp. 183–189

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

Quantum Algorithm for Minimum Multiprocessor Scheduling Problem by Central Limit Theorem

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 minimum multiprocessor scheduling problem by central limit theorem and its example are reported. When n tasks are parted by m processors, and a sum of length of each task in the k-th processor [$0 \leq k \leq m-1$. k is an integer.] is t_k, it is decided whether t_k is a finish time D or less or not. A computational complexity of a classical calculation is m^n. The computational complexity becomes about $3(m \cdot log_2 m)n$ by the quantum algorithm that uses quantum phase inversion gates, quantum inversion about mean gates, and the standard normal distribution. Therefore, a polynomial time process becomes possible.

AMS subject classification: Primary 81-08; Secondary 68R05, 68W40.
Keywords: Quantum algorithm, minimum multiprocessor scheduling problem, central limit theorem, computational complexity, standard normal distribution, polynomial time.

1. Introduction

Deutsch and Jozsa [1–3] started a quantum algorithm for a high-speed process by a parallel computation that uses quantum entangled states. Shor [2–4] revealed a polynomial time process for the factorization, and Grover [2,5,6] showed a square root time process for the database search. A quantum algorithm for the knapsack problem has recently been reported by Fujimura [7]. Its computational complexity becomes about square root for the computational complexity of a classical computation. In the minimum multiprocessor scheduling problem [8] this time, a polynomial time process becomes possible by the central limit theorem. Therefore, its result is reported.

2. Minimum Multiprocessor Scheduling Problem

When n tasks are parted by m processors, and a sum of length of each task in the k-th processor $[0 \leq k \leq m-1.$ k is an integer.] is t_k, it is decided whether t_k is a finish time D or less or not [8].

3. Quantum Algorithm

Step 1: It is assumed that n tasks are $x_0, x_1, \cdots, x_{n-1}$ that are lengths, and when they are parted by m processors $[1 \leq m \leq n.$ m is an integer. m_0 is an initial value of m.], and a sum of length of each task in the k-th processor $[0 \leq k \leq m_0 - 1.$ k is an integer.] is t_k, it is decided whether t_k is a finish time D or less or not. Therefore, it is assumed that there are $D - d \leq p \leq D$ [d is an allowed width and an integer. p is an integer.]. A distribution of x_i $[0 \leq i \leq n-1.$ i is an integer.] becomes the following. When a random variable X_i becomes x_i as a probability $1/m$, a mean is x_i/m and a dispersion is $(m-1)(x_i/m)^2$. Therefore, when a total mean is $\mu = \sum_{i=0}^{n-1}(x_i/m)$ and a total dispersion $\sigma^2 = \sum_{i=0}^{n-1}(m-1)(x_i/m)^2$, $\left(\sum_{i=0}^{n-1} X_i - \mu\right) \Big/ \sigma$ follows the normal distribution from the central limit theorem. When the standard normal distribution $f(z)$ $[\sigma = 1]$ is $\int_0^z (e^{-z^2/2}/(2\pi)^{1/2})dz$, and values of $\int_{u_g}^{v_g} (e^{-z^2/2}/(2\pi)^{1/2})dz$ are $1/2^2$, $1/2^4$, $1/2^6$, $1/2^8$, $\cdots$, and $1/2^{2g}$ [g is a natural number], each value of z is assumed u_g and v_g that their range are containd values from $D-d$ to D. u_g and v_g are obtained from the table of $f(z)$. Each total number of the data between $\mu + u_g\sigma$ and $\mu + v_g\sigma$ is $m^n/2^2, m^n/2^4, m^n/2^6, m^n/2^8, \cdots$, respectively. A height at Q is $m^n e^{-((Q-\mu)/\sigma)^2/2}/((2\pi)^{1/2}\sigma)[= H(Q)]$.

Next, a quantum algorithm is shown as the following.

First of all, quantum registers $|a_0>, |a_1>, \cdots, |a_{n-1}>, |b>$, and $|c>$ are prepared. When α is a minimum integer that is $log_2 m$ or more, each of $|a_f>$ that f is an integer from 0 to $n-1$ is consisted of α quantum bits [= qubits]. States of $|a_0>, |a_1>, \cdots, |a_{n-1}>, |b>$, and $|c>$ are $a_0, a_1, \cdots, a_{n-1}, b$, and c, respectively.

Step 2: Each qubit of $|a_0>, |a_1>, \cdots, |a_{n-1}>, |b>$, and $|c>$ is set $|0>$.

Step 3: The Hadamard gate $\boxed{\text{H}}$ [2, 3] acts on each qubit of $|a_0>, |a_1>, \cdots, |a_{n-2}>$, and $|a_{n-1}>$. It changes them for entangled states. The total states are $(2^\alpha)^n$.

Step 4: It is assumed that a quantum gate (A) changes $|b>$ for $|b+1>$ at $a_f \geq m$, or it doesn't change $|b>$ at $a_f < m$. As a target state for $|b>$ is 0, quantum phase inversion gates (PI) and quantum inversion about mean gates (IM) [2,5,6] act on $|b>$. When β is a minimum even integer that is $(2^{\alpha}/m)^{1/2}$ or more, the total number that (PI) and (IM) act on $|b>$ is β because they are a couple. Next, an observation gate (OB) observes $|b>$. These actions are repeated sequentially from $|a_0>$ to $|a_{n-1}>$. Therefore, each state of $|a_f>$ is $0, 1, \cdots, m-2$, or $m-1$, and the total states become $m^n[= W_0]$.

Step 5: It is assumed that a quantum gate (B) changes $|b>$ for $|b+x_0>$ at $a_0 = 0$, or it doesn't change $|b>$ at $a_0 \neq 0$. It changes $|b>$ for $|b+x_1>$ at $a_1 = 0$, or it doesn't change $|b>$ at $a_1 \neq 0$. This action repeats to $|a_{n-1}>$. Therefore, $|b>$ becomes from $|0>$ to $|x_0 + x_1 + \cdots + x_{n-1}>$.

Step 6: It is assumed that a quantum gate (C_1) doesn't change $|c>$ in $\mu + u_1\sigma \leq b \leq \mu + v_1\sigma$, or it changes $|c>$ for $|c+1>$ in the others of b. As the target state for $|c>$ is 0, (PI) and (IM) act on $|c>$. The number of the data that is included in $\mu + u_1\sigma \leq b \leq \mu + v_1\sigma$ is $W_1 \approx m^n/2^2$. When γ_1 is a minimum even integer that is $(W_0/W_1)^{1/2}$ or more, the total number that (PI) and (IM) act on $|c>$ is $\gamma_1 \approx 2$. Next, (OB) observes $|c>$, and the data of W_1 remain.

Similarly, (C_R) [$2 \leq R \leq g-1$. R is an integer. g that is an integer follows $\sum_{D-d \leq p \leq D} (m^n e^{-((p-\mu)/\sigma)^2/2}/((2\pi)^{1/2}\sigma)) \approx m^n/2^{2g}$.] doesn't change $|c>$ in $\mu + u_R\sigma \leq b \leq \mu + v_R\sigma$, or it changes $|c>$ for $|c+1>$ in the others of b. As the target state for $|c>$ is 0, (PI) and (IM) act on $|c>$. The number of the data that is included in $\mu + u_R\sigma \leq b \leq \mu + v_R\sigma$ is $W_R \approx m^n/2^{2R}$. When γ_R is the minimum even integer that is $(W_{R-1}/W_R)^{1/2}$ or more, the total number that (PI) and (IM) act on $|c>$ is $\gamma_R \approx 2$. Next, (OB) observes $|c>$, and the data of W_R remain. These actions are repeated sequentially from 2 to $g-1$ at R.

(C_g) doesn't change $|c>$ in $\mu + u_g\sigma \leq b \leq \mu + v_g\sigma$, or it changes $|c>$ for $|c+1>$ in the others of b. As the target state for $|c>$ is 0, (PI) and (IM) act on $|c>$. The number of the data that is included in $\mu + u_g\sigma \leq b \leq \mu + v_g\sigma$ is $W_g \approx \sum_{D-d \leq p \leq D} (m^n e^{-((p-\mu)/\sigma)^2/2}/((2\pi)^{1/2}\sigma)) \approx m^n/2^{2g}$. When γ_g is the minimum even integer that is $(W_{g-1}/W_g)^{1/2} \approx ((m^n/2^{2(g-1)})/(m^n/2^{2g}))^{1/2} = 2 \leq 2 = \gamma_g$, the total number that (PI) and (IM) act on $|c>$ is $\gamma_g \approx 2$. Next, (OB) observes $|a_0>, |a_1>, \cdots, |a_{n-1}>, |b>$, and $|c>$, and one of the data of W_g remain. When c is 0, a sum of x_f of $a_f = 0$ becomes $p[D-d \leq p \leq D]$.

Step 7: It is assumed that $n-$ (a number of $a_f = 0$) is set new $n[= n_{new}]$, x_f of $a_f \neq 0$ is renumbered for new $x_f[0 \leq f \leq n_{new}-1]$, and m is set $m-1[= m_{new}]$. When m_{new} is 1, the computational process ends, but when m_{new} isn't 1, it is returned to the step 1.

After all, m_0 combinations are obtained.

4. Numerical Computation

Step 1-1: It is assumed that there are $n = 6, x_0 = 5, x_1 = 3, x_2 = 8, x_3 = 7, x_4 = 6, x_5 = 1, m = 3, D = 11, d = 2, \mu = 10, \sigma = 6.394, g = 2, (-u_1) = v_1 = 0.3187, (-u_2) = v_2 = 0.07838$ [$\mu + u_1\sigma = 10 - 0.3187 \times 6.394 \approx 8$, $\mu + v_1\sigma = 10 + 0.3187 \times 6.394 \approx 12$, $\mu + u_2\sigma = 10 - 0.07838 \times 6.394 \approx 9$, $\mu + v_2\sigma = 10 + 0.07838 \times 6.394 \approx 11$.], $H(9) = H(11) = 44.92$, and $H(10) = 45.48$.

First of all, $|a_0>, |a_1>, \cdots, |a_5>, |b>$, and $|c>$ are prepared. When α is the minimum integer that is $log_2 3 \approx 1.6 \leq 2 = \alpha$, each of $|a_f>$ that f is the integer from 0 to 5 is consisted of 2 qubits. States of $|a_0>, |a_1>, \cdots, |a_5>, |b>$, and $|c>$ are $a_0, a_1, \cdots, a_5, b$, and c, respectively.

Step 2-1: Each qubit of $|a_0>, |a_1>, \cdots, |a_5>, |b>$, and $|c>$ is set $|0>$.

Step 3-1: $\boxed{\text{H}}$ acts on each qubit of $|a_0>, |a_1>, \cdots, |a_4>$, and $|a_5>$. It changes them for entangled states. The total states are $(2^2)^6$.

Step 4-1: (A) changes $|b>$ for $|b+1>$ at $a_f \geq 3$, or it doesn't change $|b>$ at $a_f < 3$. As the target state for $|b>$ is 0, (PI) and (IM) act on $|b>$. When β is the minimum even integer that is $(2^\alpha/m)^{1/2} = (2^2/3)^{1/2} \approx 1.2 \leq 2 = \beta$, the total number that (PI) and (IM) act on $|b>$ is $\beta = 2$. Next, (OB) observes $|b>$. These actions are repeated sequentially from $|a_0>$ to $|a_5>$. Therefore, each state of $|a_f>$ is 0, 1, or 2, and the total states become $3^6[= W_0]$.

Step 5-1: (B) changes $|b>$ for $|b + x_0>$ at $a_0 = 0$, or it doesn't change $|b>$ at $a_0 \neq 0$. It changes $|b>$ for $|b + x_1>$ at $a_1 = 0$, or it doesn't change $|b>$ at $a_1 \neq 0$. This action repeats to $|a_5>$. Therefore, $|b>$ becomes from $|0>$ to $|x_0 + x_1 + \cdots + x_5>$.

Step 6-1: (C_1) doesn't change $|c>$ in $\mu + u_1\sigma \approx 8 \leq b \leq \mu + v_1\sigma \approx 12$, or it changes $|c>$ for $|c + 1>$ in the others of b. As the target state for $|c>$ is 0, (PI) and (IM) act on $|c>$. The number of the data that is included in $8 \leq b \leq 12$ is $W_1 \approx 3^6/2^2$. When γ_1 is the minimum even integer that is $(W_0/W_1)^{1/2} \approx (3^6/(3^6/2^2))^{1/2} = 2 \leq 2 = \gamma_1$, the total number that (PI) and (IM) act on $|c>$ is $\gamma_1 \approx 2$. Next, (OB) observes $|c>$, and the data of W_1 remain.

(C_2) doesn't change $|c>$ in $\mu + u_2\sigma \approx 9 \leq b \leq \mu + v_2\sigma \approx 11$, or it changes $|c>$ for $|c + 1>$ in the others of b. As the target state for $|c>$ is 0. (PI) and (IM) act on $|c>$. The number of the data that is included in $9 \leq b \leq 11$ is $W_2 \approx 3^6/2^4$. When γ_2 is the minimum even integer that is $(W_1/W_2)^{1/2} \approx (3^6/2^2/(3^6/2^4))^{1/2} = 2 \leq 2 = \gamma_2$, the total number that (PI) and (IM) act on $|c>$ is $\gamma_2 \approx 2$. Next, (OB) observes $|a_0>, |a_1>, \cdots, |a_5>, |b>$, and $|c>$, and one of the data of $W_3[\approx H(9) + H(10) + H(11)]$ remain. For example, when

$a_0, a_1, \cdots, a_5, b$, and c are 1, 0, 1, 0, 2, 2, 10, and 0, respectively, it is obtained that there are $x_1 = 3$ and $x_3 = 7$ $[x_1 + x_3 = 10]$ in the 0th processor. But optimal combinations of the 1st and 2nd processors aren't obtained.

Step 7-1: It is set that there are $n_{new} = n - 2 = 4, x_0 = 5, x_1 = 8, x_2 = 6, x_3 = 1$, and $m_{new} = m - 1 = 2$, and the computational process is returned to the step 1.

Step 1-2: It is assumed that there are $n = 4, x_0 = 5, x_1 = 8, x_2 = 6, x_3 = 1, m = 2, D = 11, d = 2, \mu = 10, \sigma = 5.612, g = 2, (-u_1) = v_1 = 0.3187, (-u_2) = v_2 = 0.07838$ $[\mu + u_1\sigma = 10 - 0.3187 \times 5.612 \approx 8, \mu + v_1\sigma = 10 + 0.3187 \times 5.612 \approx 12, \mu + u_2\sigma = 10 - 0.07838 \times 5.612 \approx 10 \geq 9, \mu + v_2\sigma = 10 + 0.07838 \times 5.612 \approx 10 \leq 11.]$, $H(9) = H(11) = 1.119$ and $H(10) = 1.137$.

First of all, $|a_0>, |a_1>, |a_2>, |a_3>, |b>$, and $|c>$ are prepared. When α is the minimum integer that is $log_2 2 = 1 \leq 1 = \alpha$, each of $|a_f>$ that f is the integer from 0 to 3 is consisted of 1 qubit. States of $|a_0>, |a_1>, |a_2>, |a_3>, |b>$, and $|c>$ are a_0, a_1, a_2, a_3, b, and c, respectively.

Step 2-2: Each qubit of $|a_0>, |a_1>, |a_2>, |a_3>, |b>$, and $|c>$ is set $|0>$.

Step 3-2: $\boxed{\text{H}}$ acts on each qubit of $|a_0>, |a_1>, |a_2>$, and $|a_3>$. It changes them for entangled states. The total states are 2^4.

Step 4-2: Each state of $|a_f>$ is 0 or 1, and the total states become $2^4[= W_0]$. Therefore, (A) doesn't act on $|b>$.

Step 5-2: (B) changes $|b>$ for $|b + x_0>$ at $a_0 = 0$, or it doesn't change $|b>$ at $a_0 \neq 0$. It changes $|b>$ for $|b + x_1>$ at $a_1 = 0$, or it doesn't change $|b>$ at $a_1 \neq 0$. This action repeats to $|a_3>$. Therefore, $|b>$ becomes from $|0>$ to $|x_0 + x_1 + x_2 + x_3>$.

Step 6-2: (C_1) doesn't change $|c>$ in $\mu + u_1\sigma \approx 8 \leq b \leq \mu + v_1\sigma \approx 12$, or it changes $|c>$ for $|c + 1>$ in the others of b. As the target state for $|c>$ is 0, (PI) and (IM) act on $|c>$. The number of the data that is included in $8 \leq b \leq 12$ is $W_1 \approx 2^4/2^2$. When γ_1 is the minimum even integer that is $(W_0/W_1)^{1/2} \approx (2^4/(2^4/2^2))^{1/2} = 2 \leq 2 = \gamma_1$, the total number that (PI) and (IM) act on $|c>$ is $\gamma_1 \approx 2$. Next, (OB) observes $|c>$, and the data of W_1 remain.

(C_2) doesn't change $|c>$ in $\mu + u_2\sigma \approx 10 \geq 9 \leq b \leq \mu + v_2\sigma \approx 10 \leq 11$, or it changes $|c>$ for $|c + 1>$ in the others of b. As the target state for $|c>$ is 0, (PI) and (IM) act on $|c>$. The number of the data that is included in $9 \leq b \leq 11$ is $W_2 \approx 2^4/2^4 = 1$. When γ_2 is the minimum even integer that is $(W_1/W_2)^{1/2} \approx (2^4/2^2/1)^{1/2} = 2 \leq 2 = \gamma_2$, the total number that (PI) and(IM) act on $|c>$ is $\gamma_2 \approx 2$. Next, (OB) observes $|a_0>, |a_1>, |a_2>, |a_3>, |b>$, and $|c>$, and one of the data of $W_2[\approx H(9) + H(10) + H(11)]$ remain. For example, when a_0, a_1, a_2, a_3, b, and c are 0, 1, 0, 1, 11, and 0, respectively, it is obtained that there are $x_0 = 5$ and $x_2 = 6$ $[9 \leq x_0 + x_2 = 11 \leq 11]$ in the 0th processor, and $x_1 = 8$ and $x_3 = 1$ $[9 \leq x_1 + x_3 = 9 \leq 11]$ in the 1st processor.

Therefore, 3 optimal combinations are obtained.

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 at j [$2 \leq j \leq m$. j is an integer.] is $\alpha_j n_j$ at $\boxed{\mathrm{H}}$, n_j at (A), $\beta n_j = 2n_j$ at (PI) and (IM), n_j at (OB), n_j at (B), about g_j at (C_R) [$1 \leq R \leq g_j$. R is an integer.], about $\sum_{R=1}^{g_j} \gamma_R = 2g_j$ at (PI) and (IM), and about g_j at (OB). Therefore, S becomes about $\sum_{j=2}^{m}((\alpha_j + 5)n_j + 4g_j - \Omega_j)$, where when j is $2^1, 2^2, 2^3, 2^4, \cdots$, a computational complexity of the step 4 [$= \Omega_j$] is deducted from S. In the example of the section 4, S is 66. The computational complexity of the classical computation [$= Z$] is $m^n = 3^6 = 729$. After all, S/Z becomes about $1/11$. When n is large enough, S becomes about $\sum_{j=2}^{m}((\alpha_j + 5)n_j + 4g_j - \Omega_j) < \sum_{j=2}^{m}((log_2 j + 5)n_j + 2n_j \cdot log_2 j) = \sum_{j=2}^{m}(3 \cdot log_2 j + 5)n_j < (m-1)(3 \cdot log_2 m + 5)n \approx 3(m \cdot log_2 m)n$, where a maximum value of g_j is about $(n_j/2)log_2 j$, and S/Z is about $3(m \cdot log_2 m)n/m^n \approx n/m^n$. For example, as for $n = 100$ and $m = 5$, S/Z is about $100/5^{100} \approx 100/10^{70} = 1/10^{68}$.

Therefore, a polynomial time 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 [in Japanese], 2005.

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

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

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

[8] Crescenzi P., and Kann V., Eds., A compendium of NP optimization problems, 2005, [Online], Available: http://www.csc.kth.se/~viggo/wwwcompendium/.