Journal of Applied Mathematics & Bioinformatics, vol. 1, no.2, 2011, 33-38
ISSN: 1792-6602 (print), 1792-6939 (online)
International Scientific Press, 2011

On a Two-machine Flow-shop Scheduling Problem with a Single Server and Unit Processing Times

Shi Ling[1] and Cheng Xue-guang[2,*]

Abstract

We consider the problem of two-machine flow-shop scheduling with a single server and unit processing times, and show that this problem is NP-hard in the strong sense.

Mathematics Subject Classification : 90B35

Keywords: two-machine, flow-shop, single server, complexity, NP-hardness

1 Introduction

In the two-machine flow-shop scheduling problem we study, the input instance consists of n jobs with a single server and unit processing times. In the two-machine flow-shop scheduling problem we study, the input instance consists of n jobs with a single server and unit processing times. Each job J_j requires

[1] Department of Mathematics, Hubei University for Nationalities, Enshi 445000, China, e-mail: Shiling59@126.com
[2] School of Mathematics and Statistics, Wuhan University, Wuhan 430072, China, e-mail: chengxueguang6011@msn.com
[*] Corresponding author.

Article Info: *Revised* : June 13, 2011. *Published online* : November 30, 2011.

two operations $O_{1,j}$ and $O_{2,j}\,(j=1,2,...,n)$, which are performed on machine M_1 and M_2, respectively. The processing times of job J_j on machine M_i, i.e., the duration of operation $O_{i,j}$, is $p_{i,j}=1$, $(i=1,2)$. For each job, the second operation cannot be started before the first operation is completed. A setup times $s_{i,j}$ is needed before the first job is processed on machine M_i.Each setup operation must be performed by the server M_S, which can only perform one operation at a time. The objective is to compute a non-preemptive schedule of those jobs on two machines that minimize makespan. In the standard scheduling notation [5], the problem can be described as the $F2,S1\left|p_{i,j}=1\right|C_{\max}$ problem.

It is well known, S.M. Johnson [4], the $F2\left|\ \right|C_{\max}$ problem has a maximal polynomial solvable. P. Brucker [1] has shown that the $F2,S1\left|p_{i,j}=p\right|C_{\max}$ problem is NP-hard in the ordinary sense. The $F2,S1\left|p_{i,j}=1\right|C_{\max}$ problem is still open problem [3]. In this paper, we will show that the $F2,S1\left|p_{i,j}=1\right|C_{\max}$ problem is NP-hard in the strong sense.

2 Complexity of the $F2,S1\left|p_{i,j}=1\right|C_{\max}$ problem

Lemma 1 [3] Consider the $F2,S1\left|p_{ij}=1\right|C_{\max}$ problem with unit processing times $p_{i,j}=1$, where $i=1,2$ and $j=1,2,...,n.$ Then

$$C(\sigma,\tau)=\max_{1\le k\le n}\{\sum_{j\le\sigma^{-1}(k)}(s_{1,\sigma(j)}+p_{1,\sigma(j)})+\sum_{j\ge\tau^{-1}(k)}(s_{2,\tau(j)}+p_{2,\tau(j)})\} \tag{1}$$

where $\sigma^{-1}(k)$ and $\tau^{-1}(k)$ denote the positions of job k in sequences σ and τ, respectively.

For a schedule S, let $I_i(S)$, $(i=1,2)$ denote the total idle times on machine M_i, we have

$$C_{\max}(S) = \max\{\sum_{j=1}^{n}(s_{1,j} + p_{1,j}) + I_1(S), \sum_{j=1}^{n}(s_{2,j} + p_{2,j}) + I_2(S)\} \qquad (2)$$

Theorem 1 The $F2, S1\left|p_{i,j} = 1\right|C_{\max}$ problem is NP-hard in the strong sense.

Proof We prove the NP-hardness by a reduction from $3-Partition$ [2], which is known to be NP-hard in the strong sense. An instance of the $3-Partition$ problem consists of $3n+2$ natural numbers n, b, and $x_1, x_2, ..., x_{3n}$ with $b/4 < x_i < b/2$ for $1 \le i \le 3n$ and $\sum_{i=1}^{3n} x_i = nb$.Does there exist a partition of the set $\{1,2,...,3n\}$ into n sets $X_1, X_2, ..., X_n$ of triples such that $\sum_{i \in X_j} x_i = b$ for $1 \le j \le n$? In this paper, suppose $b \le 1$, because if $b > 1$, from $\sum_{i \in X_j} x_i = b$ and let $x_i / b = y_i$, we have $(\sum_{i \in X_j} x_i)/b = \sum_{i \in X_j}(x_i / b) = \sum_{i \in X_j} y_i < 1$.

Given any instance of $3-Partition$, we define the following instance of the $F2, S1\left|p_{i,j} = 1\right|C_{\max}$ problem with two types of jobs:

(1) P-job: $s_{1,j} = x_j, p_{1,j} = 1, s_{2,j} = 0, p_{2,j} = 1 \quad (j = 1,2,...,3n)$

(2) U-job: $s_{1,j} = 0, p_{1,j} = 1, s_{2,j} = 1, p_{2,j} = 1 \quad (j = 1,2,...,n)$

The threshold $y = 4n + 2 + (n+1)b$ and the corresponding decision problem is: Is there a schedule S with makespan $C(S)$ not greater than $y = 4n + 2 + (n+1)b$?

Observe that all processing times are equal to b. To prove this theorem we construct instance of the $F2, S1\left|p_{i,j} = 1\right|C_{\max}$ problem a schedule S_0 satisfying $C_{\max}(S_0) \le y = 4n + 2 + (n+1)b$ exists if and only if $3-Partition$ has a solution.

Suppose that $3-Partition$ has a solution, and $X_j (j = 1,2,...,n)$ are the required subsets of set X. Notice each set X_j contains precisely elements, since

$$b/4 < x_j < b/2 \quad , \quad \text{and} \quad \sum_{j=1}^{3m} x_j = nb \quad \text{for all} \quad j = 1,2,...,n.$$

Let σ denote a sequence of the elements of set X for which $X_j = \{\sigma(3j-2), \sigma(3j-1), \sigma(3j)\}$, for $j = 1,2,...,n$ .The desired schedule S_0 exists and can be described as follows.

(a) No machine has intermediate idle time,

(b) Machine M_1 process the P-jobs and U-jobs in order of the sequence σ,

$$\sigma = (P_{\sigma(1,1)}, P_{\sigma(1,2)}, P_{\sigma(1,3)}, U_{1,1}, P_{\sigma(1,4)}, P_{\sigma(1,5)}, P_{\sigma(1,6)}, U_{1,2}, ..., P_{\sigma(1,3n-2)}, P_{\sigma(1,3n-1)}, P_{\sigma(1,3n)}, U_{1,n})$$

(c) While machine M_2 process the P-jobs and U-jobs in the order of sequence τ,

$$\tau = (U_{2,1}, P_{\sigma(2,1)}, P_{\sigma(2,2)}, P_{\sigma(2,3)}, U_{2,2}, ..., U_{2,r}, P_{\sigma(2,3n-2)}, P_{\sigma(2,3n-1)}, P_{\sigma(2,3n)})$$

as indicated in Figure 1.

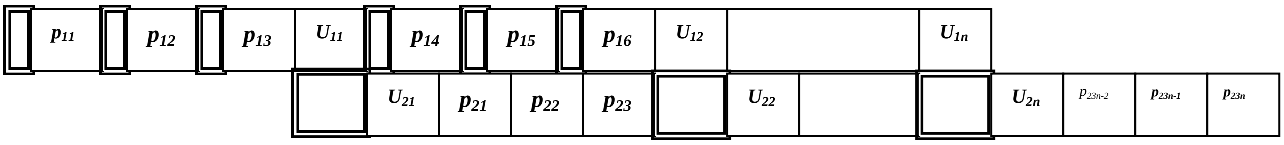

Figure 1: Gantt chart for the $F2, S1 | p_{i,j} = 1 | C_{\max}$ problem

Then we define sequences σ and τ shown in Figure 1. Obviously, these sequences σ and τ fulfills $C(\sigma,\tau) \le y$. Conversely, assume that this flow-shop scheduling problem has a solution σ and τ with $C(\sigma,\tau) \le y$. By setting $\sigma(j) = j (j = 1,2,3)$ in (1), we get for all sequences σ and τ:

$$C(\sigma,\tau) \ge (s_{1,1} + p_{1,1} + s_{1,2} + p_{1,2} + s_{1,3} + p_{1,3}) + \sum_{\lambda=1}^{n} (s_{2,\tau_\lambda} + p_{2,\tau_\lambda}) = 4n + 2 + (n+1)b = y.$$

Thus, for these sequences σ and τ with $C(\sigma,\tau)=y$. We may conclude that: (3) machine M_1 process jobs in the interval $[0,(n-1)+n(3+b)]$, without idle times. In the interval $[(3+b+1)j, j+(j+1)(3+b)]$, $(j=1,2,...,n-1)$, machine M_1 process P-jobs, in the interval $[j-1+j(3+b), j(3+b+1)]$, $(j=1,2,...,n)$ machine M_1 process U-jobs, (4) machine M_2 process jobs in the interval $[3+b, 4n+2+(n+1)b]$, without idle times. In the interval $[9j-10+j(3+b),(j-10+j(3+b)+b+1]$, $(j=1,2,...,n)$ machine M_2 process U-jobs, in the interval $[(j-1)+j(3+b+b+1),(j-1)+j(3+b)+b+1+3)]$ machine M_2 process P-jobs. Now, we will prove that the

$$\sum_{i\in X_1}(s_{1,i}+p_{1,i})=4b \text{ .If} \sum_{i\in X_1}(s_{1,i}+p_{1,i})\geq 4b,$$

then U_{21}-job cannot start processing at time $4b$, which contradicts (4). If $\sum_{i\in X_1}(s_{1,i}+p_{1,i})\leq 4b$, then there is idle time before machine M_1 process job $U_{1,1}$, which contradicts (3). Thus, we have

$$\sum_{i\in X_1}(s_{1,i}+p_{1,i})=4b.$$

Since $p_{1,1}=p_{1,2}=p_{1,3}=b, s_{1,i}=x_i$, then

$$\sum_{i\in X_1}(s_{1,i}+p_{1,i})=(s_{1,1}+p_{1,1}+s_{1,2}+p_{1,2}+s_{1,3}+p_{1,3})=3b+\sum_{i\in X_1}x_i=4b, \sum_{i\in X_1}x_i=b.$$

The set X_1 give a solution to $3-Partition$.

Analogously, we show that the remaining sets $X_2, X_3,..., X_n$ separated by the jobs $1,2,...,n$ contain 3-element and fulfill $\sum_{i\in X_j}x_i=b$ for $j=1,2,...,n$. Thus, $X_1, X_2,..., X_n$ define a solution of $3-Partition$. □

References

[1] P. Brucker, S. Knust, G.Q. Wang, et al., Complexity of results for flow-shop problems with a single server, *European J. Oper. Res.*, 165, (2005), 398-407.

[2] P.C. Gilmore, R.E. Gomory, Sequencing a one-state variable machine: A solvable case of the traveling salesman problem, *Operations Research*, **12**, (1996), 655-679.

[3] http:www.mathematik.uni-osnabruckde/research/OR.class.

[4] S.M. Johnson. Optimal two-and-three-stage production schedules with set-up times included. Naval Res. Quart., **1**, (1954), 61-68.

[5] E.L. Lawer, J.K. Lenstra, A.H.G. Rinnooy Kan, D.B. Shmoys, Sequencing and Scheduling: Algorithms and Complexity, in: S.C. Gtaves, A.H.G. Rinnooy Kan, P.H. Zipkin(Eds.), Handbooks in Operation Research and Management Science, v. 4, Logistics of Production and Inventory, North_Holland, Amsterrdem, 445-522, 1993.

[6] W.C.Yu, *The two-machine flow shop problem with delays and the one machine total tardiness problem*, Technische Universiteit Eindhoven, 1996.