Journal of Computations & Modelling, vol.2, no.3, 2012, 145-149
ISSN: 1792-7625 (print), 1792-8850 (online)
Scienpress Ltd, 2012

Two-machine Flow Shop Scheduling Problem with a Single Server and Equal Processing Times

Shi ling[1] and Cheng xue-guang[2]

Abstract

We study the problem of two-machine flow-shop scheduling with a single server and equal processing times, we 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 this paper, we consider the problem in which we have two machines M_1, M_2, a single server M_s and n jobs J_j with equal processing times

---

[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

Article Info: *Received* : September 24, 2012. *Revised* : October 18, 2012
Published online : November 20, 2012

$p_{1,j} = p_{2,j} = p$ and server times $s_{1,j}, s_{2,j}$ on machine M_1 and M_2, respectively. The problem can be described as the $F2, S1|p_{i,j} = p|C_{\max}$ problem.

It is well known, S.M. Johnson [1], the $F2||C_{\max}$ problem has a maximal polynomial solvable. P. Brucker [2] shows that the $F2, S1|p_{i,j} = p|C_{\max}$ problem is NP-hard in the binary sense. In this paper, we will show that this problem is NP-hard in the strong sense.

2 Main result

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

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

where $\sigma^{-1}(k)$ and $\tau^{-1}(k)$ denote the positions of job k in sequence σ and τ, respectively. For a schedule S, let $I_{i,j}(S)$ $(i = 1,2; j = 1,2,...,n)$ denote the total idle times of job J_j on machine M_i, we have that

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

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

Proof. We prove that the $F2, S1|p_{i,j} = p|C_{\max}$ problem is NP-hard in the strong sense through a reduction from the $3-Partition$ problem [4], which is known to be NP-hard in the strong sense, to the $F2, S1|p_{i,j} = p|C_{\max}$ problem. The $3-Partition$ problem is then stated as:

$3-Partition$**:** Given a set of positive integers $X = \{x_1, x_2,..., x_{3r}\}$, and a positive integer b with:

$$\sum_{j=1}^{3r} x_j = rb\,,\quad b/4 < x_j < b/2\,,\quad \forall j = 1, 2, ..., r \tag{3}$$

Decide whether there exists a partition of X into r disjoint 3-element subset $\{X_1, X_2, ..., X_r\}$ such that

$$\sum_{j \in X_i} x_j = b \qquad (i = 1,2,...,r) \tag{4}$$

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

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

(2) U -job: $s_{1,j} = 0,\ p_{1,j} = b, s_{2,j} = b,\ p_{2,j} = b \qquad (j = 1,2,...,r)$

The threshold $y = 5br + 4b$ and the corresponding decision problem are: Is there a schedule S with makespan $C(S)$ not greater than $y = 5br + 4b$?

Observe that all processing times are equal to b .To prove the theorem we show that in this constructed instance of the $F2, S1 \mid p_{i,j} = p \mid C_{\max}$ problem a schedule S_0 satisfying $C_{\max}(S_0) \le y = 5br + 4b$ exists if and only if the $3-Partition$ problem has a solution.

Suppose that the $3-Partition$ problem has a solution, and $X_j \quad (j = 1, 2, ..., r)$ are the required subsets of set X . Notice that each set X_j contains precisely elements, since $b/4 < x_j < b/2$, and $\sum_{j=1}^{3r} x_j = rb$ for all $j = 1,2,...,r$.

Let σ denotes a sequence of the elements of set X for which $X_j = \{\sigma(3j-2), \sigma(3j-1), \sigma(3j)\}$, for $j = 1,2,...,r$. The desired schedule S_0 exists and can be described as follows. No machine has intermediate idle time. Machine M_1 process the P -jobs and U -jobs in order of the sequence σ , i.e., in the sequence

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

While machine M_2 process the P -jobs and U -jobs in the order of sequence τ , i.e., in the sequence

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

Then, 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:

$$C(\sigma,\tau) \ge (s_{1,1} + p_{1,1} + s_{1,2} + p_{1,2} + s_{1,3} + p_{1,3}) + \sum_{\lambda=1}^{n} (s_{2,\tau_\lambda} + p_{2,\tau_\lambda}) = 5rb + 4b = y$$

Thus, for sequences σ and τ with $C(\sigma,\tau) = y$. We may conclude that

(a) Machine M_1 process jobs in the interval $[0,5rb]$, without idle times. In the interval $[5jb,(5j+4)b]$ $(j = 0,1,...,r-1)$, machine M_1 process P-jobs, in the interval $[4jb,5jb]$ $(j = 1,2,...,r)$ machine M_1 process U-jobs,

(b) Machine M_2 process jobs in the interval $[4b,5rb+4b]$, without idle times. In the interval $[(6+5j)b,(9+5j)b]$ $(r = 0,1,...,r-1)$, machine M_2 process P-jobs, in the interval $[(4+5j)b,(6+5j)b]$ $(j = 0,1,...,r-1)$ machine M_2 process U-jobs. Now, we will prove that the $\sum_{j\in X_i} (s_{1,i} + p_{1,i}) = 4b$.

If $\sum_{j\in X_i} (s_{1,i} + p_{1,i}) \ge 4b$, then U_{21}-job cannot start processing at time $4b$, which contradicts (2).

If $\sum_{j\in X_i} (s_{1,i} + p_{1,i}) \le 4b$, then there is idle time before machine M_1 process job $U_{1,1}$, which contradicts (1). Thus, we have $\sum_{j\in X_i} (s_{1,i} + p_{1,i}) = 4b$.

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

$$\sum_{j\in X_i} (s_{1,i} + p_{1,i}) = (s_{1,1} + p_{1,1} + s_{1,i} + p_{1,2} + s_{1,3} + p_{1,3}) = 3b + \sum_{j\in X_i} x_j = 4b,$$

$$\sum_{j\in X_i} x_j = b.$$

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

Analogously, we show that the remaining sets $X_2, X_3, ..., X_r$ separated by the jobs

$1,2,...,r$ contain 3-element and fulfill $\sum_{j \in X_i} x_j = b$ for $j = 1,2,...,r$. Thus, $X_1, X_2,..., X_r$ define a solution of the $3-Partition$ problem.

References

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

[2] 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**(2), (2005), 398-407.

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

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