Journal of Informatics and Mathematical Sciences
Volume 4 (2012), Number 1, pp. 123–127

http://www.rgnpublications.com

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

Shi Ling and Cheng Xue Guang

Abstract. We consider the problem of two-machine flow-shop scheduling with a single server and unit server times, we show that this problem is *NP*-hard in the strong sense and present a simple greedy algorithm for it with worst-case bound $\frac{3}{2}$.

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 server times. Each job J_j requires two operations $O_{1,j}$ and $O_{2,j}$, 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 $p_{i,j}$. For each job, the second operation cannot be started before the first operation is completed. A unit 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, 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, the problem can be described as the $F2,\ S1|s_{i,j} = 1|C_{\max}$ problem. It is well known, S.M. Johnson [1], the $F2$ $||C_{\max}$ problem has a maximal polynomial solvable. P. Brucker [2] and C.A. Glass [3] proved that the $F2,\ S1|s_{i,j} = s|C_{\max}$ problem and the$F2,\ S1||C_{\max}$ problem are *NP*-hard in the strong sense. The $F2,\ S1|s_{i,j} = 1|C_{\max}$ problem is still open problem [4]. In this paper, we will show that this problem is *NP*-hard in the strong sense, and present a simple greedy algorithm for it.

2010 *Mathematics Subject Classification.* 90B35.
Key words and phrases. Two-machine; Flow-shop; Single server; Complexity; *NP*-hardness, Worst-case analysis.

2. Complexity of the $F2,\ S1|s_{i,j}=1|C_{\max}$ problem

Lemma 1 ([4]). *Consider the $F2,\ S1|s_{i,j}=1|C_{\max}$ problem with processing times $p_{i,j}$ and server times $s_{i,j}$, where $i=1,2$ and $j=1,2,\ldots,n$. Then*

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

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

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

Proof. We prove that the $F2,\ S1|s_{i,j}=1|C_{\max}$ problem is *NP*-hard in the strong sense through a reduction from the Numerical Matching with Target Sums (*NMTS*) problem, which is known to be *NP*-hard in the strong sense [6], to the $F2$, $S1|s_{i,j}=1|C_{\max}$ problem. The *NMTS* problem is then stated as:

Given three sets $X=x_i,x_2,\ldots,x_r$, $Y=y_1,y_2,\ldots,y_r$ and $Z=z_1,z_2,\ldots,z_r$ of positive integers, where $\sum_{i=1}^{r}x_i=\sum_{i=1}^{r}y_i+\sum_{i=1}^{r}z_i$, does there exist permutation $y_{j_1},y_{j_2},\ldots,y_{j_r}$ and $z_{j_1},z_{j_2},\ldots,z_{j_r}$ such that $x_i=y_{j_r}+z_{j_r}$ for $i=1,2,\ldots,r$. Given any instance of *the NMTS* problem, we define the following instance of the $F2$, $S1|s_{i,j}=1|C_{\max}$ problem with five types of jobs:

(1) U-jobs: $s_{1,j}=1$, $p_{1,j}=1$; $s_{2,j}=1$, $p_{2,j}=3K+x_j+3$, $j=1,2,\ldots,r$
(2) V-jobs: $s_{1,j}=1$, $p_{1,j}=2K+y_{j-r}$; $s_{2,j}=1$, $p_{2,j}=1$, $j=1,2,\ldots,r$
(3) W-jobs: $s_{1,j}=1$, $p_{1,j}=K+z_{j-r}$; $s_{2,j}=1$, $p_{2,j}=1$, $j=1,2,\ldots,r$
(4) P-jobs: $s_{1,j}=1$, $p_{1,j}=5; s_{2,j}=1$, $p_{2,j}=1$, $j=1,2,\ldots,r$
(5) Q-jobs: $s_{1,4r+1}=1$, $p_{1,4r+1}=1$; $s_{2,4r+1}=1$, $p_{2,4r+1}=1$.

The threshold $y=10r+3Kr+K+3$ and the corresponding decision problem are: Is there a schedule S with makespan $C(S)$ not greater than $y=10r+3Kr+K+3$? Observe that all processing times are equal to b. To prove the theorem we show that in this constructed instance of the $F2,\ S1|s_{i,j}=1|C_{\max}$ problem a schedule S_0 satisfying $C_{\max}(S^0)\le y=10n+3Kn+K+3$ exists if and only if *NMTS* has a solution. Suppose that *NMTS* has a solution. 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 in order of the sequence σ, i.e., in the sequence

$$\sigma=\{\sigma_{U_{1,1}},\sigma_{V_{1,1}},\sigma_{W_{1,1}},\sigma_{P_{1,1}},\ldots,\sigma_{U_{1,n}},\sigma_{V_{1,n}},\sigma_{W_{1,n}},\sigma_{P_{1,n}},\sigma_{1,4r+1}\}.$$

While machine M_2 process the jobs in the sequence

$$\tau=\{\tau_{U_{2,1}},\tau_{V_{2,1}},\tau_{W_{2,1}},\tau_{P_{2,1}},\ldots,\tau_{U_{2,n}},\tau_{V_{2,n}},\tau_{W_{2,n}},\tau_{P_{2,n}},\tau_{2,4r+1}\}$$

as indicated in Figure 1.

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

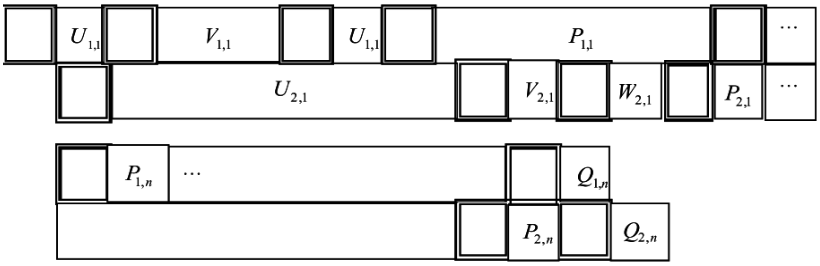

Figure 1. Gantt chart for the $F2,\ S1|s_{i,j}=1|C_{\max}$ problem

scheduling problem has a solution σ and τ with $C(\sigma,\tau)\le y$. By setting in (1), we get for all sequences σ and τ:

$$C(\sigma,\tau)\ge s_{1,1}+\sum_{\lambda=1}^{n}(s_{2,\tau_\lambda}+p_{2,\tau_\lambda})=10n+3Kn+K+3=y\,.$$

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

(1) There is no idle time on machine M_1 until the completion of the last job on it. Machine M_1 process jobs in the interval $[0,10n+3Kn+K+2]$.
(2) There is no idle time on machine M_2 until the completion of the last job on it. Machine M_2 process jobs in the interval $[1,10n+3Kn+K+3]$.
(3) $Q_{1,1}$, $Q_{2,1}$ are the last jobs on machine M_1, M_2, respectively.

Now, we will prove that $x_1=y_1+z_1$, that is

$$s_{1,V_{1,1}}+p_{1,V_{1,1}}+s_{1,W_{1,2}}+p_{1,W_{1,2}}+s_{1,P_{1,1}}=p_{2,U_{2,1}}\,.$$

If $s_{1,V_{1,1}}+p_{1,V_{1,1}}+s_{1,W_{1,2}}+p_{1,W_{1,2}}+s_{1,P_{1,1}}>p_{2,U_{2,1}}$, then there is a idle time between $p_{2,U_{2,1}}$ and $s_{2,P_{2,1}}$, which contradicts (2), if $s_{1,V_{1,1}}+p_{1,V_{1,1}}+s_{1,W_{1,2}}+p_{1,W_{1,2}}+s_{1,P_{1,1}}<p_{2,U_{2,1}}$ then there is a idle time between $s_{1,P_{1,1}}$ and $p_{1,P_{1,1}}$, which contradicts (1). Thus, we have $s_{1,V_{1,1}}+p_{1,V_{1,1}}+s_{1,W_{1,2}}+p_{1,W_{1,2}}+s_{1,P_{1,1}}=p_{2,U_{2,1}}$. Since $s_{1,V_{1,1}}=s_{1,W_{1,1}}=s_{1,P_{1,1}}=1$, $p_{1,U_{1,1}}=2K+y_1$, $p_{1,V1,1}=K+z_1$, $p_{2,P_{1,1}}=3+3K+x_1$ then $1+2K+y_1+K+z_1=3+3K+x_1$, that is $x_1=y_1+z_1$. This give a solution to *NMTS*. Analogously, we show that $x_j=y_j+z_j(j=1,2,\dots,n)$. Thus, $x_j=y_j+z_j$, $j=2,3,\dots,n$ defines a solution of the *NMTS*. □

3. Algorithm for the $F2,\ S1|s_{i,j}=1|C_{\max}$ problem

For the $F2,\ S1|s_{i,j}=1|C_{\max}$ problem, we consider a simple greedy algorithm.

Algolrithm 1.

(1) Schedule all jobs in shortest processing times (*SPT*) first on machine M_1, that is increasing processing times order.
(2) schedule all jobs in shortest processing times (*SPT*) first on machine M_2, that is in increasing processing times order, too.

Theorem 2. *The $F2,\ S1|s_{i,j}=1|C_{\max}$ problem, let S_0 be a schedule created by Algorithm 1, S^* be the optimal solution for the $F2,\ S1|s_{i,j}=1|C_{\max}$ problem, then $C_{\max}(S^0)/C_{\max}(S^*)\le 3/2$. The bound is tight.*

Proof. Let $T_{i,j}$, $I_{i,j}$ denote the start time and idle time of job J_j on the machine M_i, $i=1,2$ respectively. According to Algorithm 1, Schedule the jobs in increasing order of $p_{1,j}$ on the machine M_1, with total idle time $I_{1,j}$.Schedule the jobs in increasing order of $p_{2,j}$ on the machine M_2 with the total idle time $I_{2,j}$. For any $j(1\le j\le n)$, we have

$$\begin{aligned}
C_j &= T_{1,j}+s_{1,j}+p_{1,j}+s_{2,j}+p_{2,j}\\
&= \sum_{i=1}^{j-1}(s_{1,i}+p_{1,i})+I_{1,j}+s_{1,j}+p_{1,j}+s_{2,j}+p_{2,j}\\
&= \sum_{i=1}^{j}(s_{1,i}+p_{1,i}+I_{1,j}+s_{2,j}+p_{2,j})\\
&\le j+jp_{1,j}+I_{1,j}+1+p_{2,j}\,,\\
C_j &= T_{2,j}+s_{2,j}+p_{2,j}\\
&= \sum_{i=1}^{j-1}(s_{2,j}+p_{2,j})+I_{2,j}+s_{2,j}+p_{2,j}\\
&= \sum_{i=1}^{j}(s_{2,j}+p_{2,j})+I_{2,j}\\
&\le (j+jp_{2,n}+I_{2,j})\,,\\
2C_{\max}(S^0) &\le n+np_{1,n}+I_{1,j}+n+np_{2,n}+I_{2,j}+1+p_{2,n}\\
&= (n+np_{1,n}+I_{1,j})+(n+np_{2,n}+I_{2,n})+(1+p_{2,n})\\
&\le 3C_{\max}(S^*)\\
C_{\max}(S^0)/C_{\max}(S^*) &\le 3/2\,.
\end{aligned}$$

To prove the bound is tight, introduce the following example as shown in Figure 2 and Figure 3.

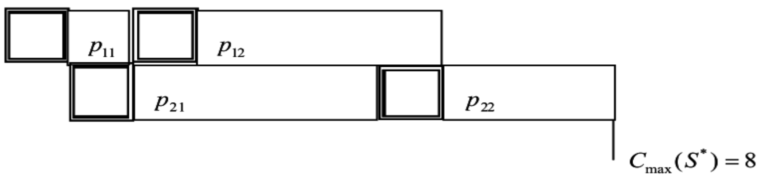

Figure 2. $C_{\max}(S^*)$

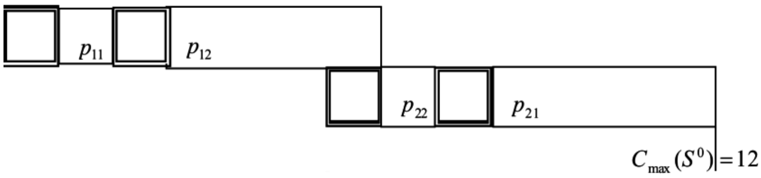

Figure 3. $C_{\max}(S^0)$

So we have $C_{\max}(S^0)/C_{\max}(S^*) = 12/8 = 3/2$, the bound is tight. □

References

[1] P. Brucker, S. Knust and G.Q. Wang et al., Complexity results for flow-shop problems with a single server, *European J. Oper. Res.* **165**(2) (2005), 398–407.
[2] M.R. Garey, D.S. Johnson and R. Sethi, The complexity of flow-shop and job-shop scheduling, *Math. Oper. Res.* **1**(2) (1976), 17–129.
[3] C.A. Glass, Y.M. Shafransky and V.A. Strusevich, Scheduling for parallel dedicated machines with a single server, *Naval Research Logistics* **47** (2000), 304–328.
[4] W.C. Yu, *The two-machine flow shop problem with delays and the one machine total tardiness problem*, Technische Universiteit Eindhoven, 1996.
[5] `http:www.mathematik.uni-osnabruckde/research/OR.class`
[6] M.R. Garey and D.S. Johnson, *Complexity and Intractability: A Guide to The Theory of* NP-*Completeness*, W.H.Freeman, San Francisco, CA, 1979.

Shi Ling, *Department of Mathematics, Hubei University for Nationalities, Enshi 445000, China.*
E-mail: shiling59@126.com

Cheng Xue Guang, *School of Mathematics and Statistics, Wuhan University, Wuhan 430072, China.*
E-mail: chengxueguang6011@msn.com

Received May 13, 2011
Accepted October 23, 2011