Heuristics for two-machine flowshop scheduling with setup times and an availability constraint

Xiuli Wang[a] T. C. Edwin Cheng[b, #]

[a]College of Electrical Engineering
Zhejiang University, P. R. China

[b]Department of Logistics
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong

Abstract

This paper studies the two-machine flowshop scheduling problem with anticipatory setup times and an availability constraint imposed on only one of the machines where interrupted jobs can resume their operations. We present two heuristics and show that their worst-case error bounds are no larger than 2/3.

Keywords: Flowshop scheduling; Heuristics; Error bound

[#]Corresponding author.

1. Introduction

Machine scheduling problems with availability constraints motivated by preventive maintenance have received increasing attention from researchers. The studies in the literature on this topic mainly deal with three situations, namely *resumable*, *nonresumable*, and *semiresumable*. If a job cannot be finished before the unavailable period of a machine and the job can continue after the machine becomes available again, it is called resumable. On the other hand, if the job has to restart rather than continue, the situation is called nonresumable. If the unfinished job will have to partially restart after the machine becomes available again, the situation is called semiresumable. The recent research results on this subject can be found in the review papers by Lee et al. [1], Sanlaville and Schmidt [2], and Schmidt [3].

The two-machine flowshop scheduling problem with availability constraints was first studied by Lee [4]. Under the resumable assumption, he proved that the problem is NP-hard when an availability constraint is imposed on only one machine and proposed a pseudo-polynomial dynamic programming algorithm to solve the problem optimally. He also developed two heuristics. The first heuristic is for solving the problem where the availability constraint is imposed on machine 1, which has a worst-case error bound of 1/2. The second heuristic is for solving the problem where the availability constraint is imposed on machine 2, which has a worst-case error bound of 1/3. Lee [5] further studied the semiresumable case and developed a pseudo-polynomial dynamic programming algorithm and heuristics. For the resumable case, Cheng and Wang [6] developed an improved heuristic when the availability constraint is imposed on the first machine, and the heuristic has a worst-case error bound of 1/3. Breit [7] presented an improved heuristic for the problem with an availability constraint only on the second machine and showed that the heuristic has a worst-case error bound of 1/4. Cheng and Wang [8] considered a special case of the problem where the availability constraint is imposed on each machine, and the two availability constraints are consecutive. They developed a heuristic and showed that it has a worst-case error bound of 2/3 for the nonresumable situation. In addition, the two-machine flowshop scheduling problem with availability constraints has also been studied under the no-wait processing environment by Cheng and Liu [9, 10]. For the general flowshop scheduling problem with availability constraints, Aggoune [11] proposed a heuristic based on a genetic algorithm and a tabu search.

In all the above-mentioned flowshop scheduling models, setup times are not considered; in other words, setup times are assumed to be included in processing times. However, in many industrial settings, it is necessary to treat setup times as separated from processing times (see, for example, [12, 13]). In this paper we consider the two-machine flowshop scheduling problem with anticipatory setup times, where the availability constraint is imposed on only one machine. The setup times are anticipatory, i.e., the setup for the second operation of any job on machine 2 can start before the completion of its first operation on machine 1 whenever there is some idle time on machine 2. We assume that the processing order of jobs is the same on each machine. That is, we confine ourselves to finding solutions that are permutation schedules for the problem. We also assume that all the jobs and their setups are resumable. The objective is to minimize the makespan. It is evident from Lee [4] that our problem is NP-hard. In the next section, we introduce the notation and some preliminaries. In Sections 3 and 4, we study the cases where the availability constraint is imposed on machines 1 and 2, respectively. Some concluding remarks are given in the last section.

2. Notation and preliminaries

For the problem under consideration, we introduce the following notation to be used throughout this paper.

$S = \{J_1, \cdots, J_n\}$: a set of n jobs;

M_1, M_2: machine 1 and machine 2;

$\Delta_l = t_l - s_l$: the length of the unavailable interval on M_l, where M_l is unavailable from time s_l to t_l, $0 \le s_l \le t_l$, $l = 1, 2$;

s_i^1, s_i^2: setup times of J_i on M_1 and M_2, respectively, where $s_i^1 > 0, s_i^2 > 0$;

a_i, b_i: processing times of J_i on M_1 and M_2, respectively, where $a_i > 0, b_i > 0$;

$\pi = [J_{\pi(1)}, \cdots, J_{\pi(n)}]$: a permutation schedule, where $J_{\pi(i)}$ is the *i*th job in π;

π*: an optimal schedule;

C_{Hx}: the makespan yielded by heuristic Hx;

C*: the optimal makespan.

Following the notation of Lee [1], we denote the problem under study as *F2*/*setup*, *r-a*(M_1)/C_{max}, i.e., the makespan minimization problem in a two-machine flowshop with setup times and a resumable availability constraint on M_l. As an example, consider a problem instance of *F2*/*setup*, *r-a*(M_1)/C_{max} with $n = 3$. Let $s_1^1 = 2$, $s_1^2 = 3$, $a_1 = 8$, $b_1 = 5$, $s_2^1 = 6$, $s_2^2 = 10$, $a_2 = 5$, $b_2 = 4$, $s_3^1 = 5$, $s_3^2 = 7$, $a_3 = 3$, $b_3 = 6$, $s_1 = 19$, and $t_1 = 26$. A schedule $\pi = [J_1, J_2, J_3]$ for the instance is shown in Fig. 1.

The classical two-machine permutation flowshop scheduling problem with setup times, denoted as *F2*/*permu*, *setup*/C_{max}, can be optimally solved by the Yoshida and Hitomi algorithm (YHA) in O(*nlogn*) time [14]. YHA works in the following manner:

Divide *S* into two disjoint subsets *A* and *B*, where $A = \{J_i \mid s_i^1 + a_i - s_i^2 \le b_i\}$ and $B = \{J_i \mid s_i^1 + a_i - s_i^2 > b_i\}$. Sequence the jobs in *A* in nondecreasing order of $s_i^1 + a_i - s_i^2$ and the jobs in *B* in nonincreasing order of b_i. Arrange the ordered subset *A* first, followed by the ordered subset *B*.

3. The unavailable interval is on *M₁*

In this section we develop a heuristic for the problem *F2*/*setup*, *r-a*(M_1)/C_{max} and evaluate its worst-case error bound. The basic ideas of our heuristic are to combine a few simple heuristic rules and then improve the schedules by re-arranging the order of some special jobs with large setup times or large processing times on M_2 in different situations.

Heuristic H1:

(1) Find jobs J_p and J_q such that

$$s_p^2 + b_p \geq s_q^2 + b_q \geq \max\{s_i^2 + b_i \mid J_i \in S \setminus \{J_p, J_q\}\}.$$

(2) Sequence the jobs by YHA. Let the corresponding schedule be π_1 and the corresponding makespan be $C_{max}(\pi_1)$.

(3) Sequence the jobs in nonincreasing order of $(s_i^2 + b_i)/(s_i^1 + a_i)$. Let the schedule be π_2 and the corresponding makespan be $C_{max}(\pi_2)$.

(4) Place job J_p in the first position and keep the other $n-1$ jobs in the same positions as those in Step (3). Let the corresponding schedule be π_3.

(5) If $(s_p^1 + a_p) + (s_q^1 + a_q) \leq s_1$, then sequence jobs J_p and J_q as the first two jobs such that the completion time of the last one is minimized. The remaining $n-2$ jobs are sequenced randomly. Let the corresponding schedule be π_4.

(6) Select the schedule with the minimum makespan from the above four schedules. Let $C_{H1}=\min\{C_{max}(\pi_1), C_{max}(\pi_2), C_{max}(\pi_3), C_{max}(\pi_4)\}$.

The time complexity of Heuristic H1 is O($nlogn$). In the following, we analyze the performance bound of Heuristic H1.

Let π be a schedule for the problem $F2/setup,\ r\text{-}a(M_1)/C_{max}$. We define the critical job $J_{\pi(k)}$ in π as the last job in π such that its starting time on M_2 is equal to its finishing time on M_1.

Lemma 1. For schedule π_2 defined in Step (3) of Heuristic H1, we assume that the completion time of the critical job $J_{\pi_2(k)}$ on M_1 is t, and let $J_{\pi(v)}$ be the last job that finishes no later than time t on M_1 in a schedule π. The following inequality holds:

$$C_{max}(\pi_2) \le C_{max}(\pi) + b_{\pi_2(k)} + s^2_{\pi(v+1)}.$$

Proof For schedule π_2, its makespan is

$$C_{max}(\pi_2) = t + b_{\pi_2(k)} + \sum_{j=k+1}^{n}(s^2_{\pi_2(j)} + b_{\pi_2(j)}). \qquad (1)$$

Under the assumption of Lemma 1, we have

$$\sum_{j=1}^{v}(s^1_{\pi(j)} + a_{\pi(j)}) \le \sum_{j=1}^{k}(s^1_{\pi_2(j)} + a_{\pi_2(j)}),$$

then

$$\sum_{j=v+1}^{n}(s^1_{\pi(j)} + a_{\pi(j)}) \ge \sum_{j=k+1}^{n}(s^1_{\pi_2(j)} + a_{\pi_2(j)}). \qquad (2)$$

Since all the jobs are sequenced in nonincreasing order of $(s^2_{\pi_2(j)} + b_{\pi_2(j)})/(s^1_{\pi_2(j)} + a_{\pi_2(j)})$ in π_2, and from (2), it is not difficult to check that the following inequality holds:

$$\sum_{j=v+1}^{n}(s^2_{\pi(j)} + b_{\pi(j)}) \ge \sum_{j=k+1}^{n}(s^2_{\pi_2(j)} + b_{\pi_2(j)}). \qquad (3)$$

For schedule π, we have

$$C_{max}(\pi) \ge t + \sum_{j=v+1}^{n}(s^2_{\pi(j)} + b_{\pi(j)}) - s^2_{\pi(v+1)}. \qquad (4)$$

Therefore, from (1), (3) and (4), we have

$$C_{max}(\pi_2) \le C_{max}(\pi) + b_{\pi_2(k)} + s^2_{\pi(v+1)}. \qquad \square$$

Theorem 1. For the problem *F2/setup, r-a*(M_1)/C_{max}, $(C_{\mathrm{H1}} - C^*)/C^* \le 2/3$.

Proof If $\sum_{i=1}^{n}(s_i^1 + a_i) \le s_1$, it is obvious that schedule π_1 obtained from Heuristic H1 is an optimal schedule for the problem under study. Hence, in the following text, we assume that $\sum_{i=1}^{n}(s_i^1 + a_i) > s_1$.

Notice that since all the jobs are resumable for the problem *F2/setup, r-a*(M_1)/C_{max}, we have $C_{max}(\pi_1) \le C^* + \Delta_1$. If $\Delta_1 \le 2C^*/3$, then we are done. So, in the following, we focus on the situation where $\Delta_1 > 2C^*/3$.

Because $\Delta_1 > 2C^*/3$ and $\sum_{i=1}^{n}(s_i^1 + a_i) + \Delta_1 < C^*$, we have

$\sum_{i=1}^{n}(s_i^1 + a_i) < C^*/3$. Let $S' = \{J_i \mid s_i^2 + b_i > C^*/3, i = 1, 2, \cdots, n\}$. It is obvious that $|S'| \le 2$. When $|S'| = 0$, for an optimal schedule π^*, according to Lemma 1, we have $C_{max}(\pi_2) \le C^* + b_{\pi_2(k)} + s_{\pi^*(v+1)}^2 < 5C^*/3$. Thus, we only need to consider the following two cases.

Case 1: $|S'| = 1$

In this case, $S' = \{J_p\}$. If $s_p^2 \le C^*/3$ and $b_p \le C^*/3$, then from Lemma 1, we are done. Otherwise, we consider schedule π_3 obtained in Step (4) of Heuristic H1.

For subcase $s_p^1 + a_p \le s_1$, suppose that the critical job does not exist in π_3, then $C_{max}(\pi_3) = \sum_{i=1}^{n}(s_{\pi_3(i)}^2 + b_{\pi_3(i)}) = C^*$. Otherwise, we denote the critical job as $J_{\pi_3(u)}$. If $\sum_{i=1}^{u}(s_{\pi_3(i)}^1 + a_{\pi_3(i)}) \le s_1$, then

$$\begin{aligned} C_{max}(\pi_3) &= \sum_{i=1}^{u}(s_{\pi_3(i)}^1 + a_{\pi_3(i)}) + (\sum_{i=u+1}^{n}(s_{\pi_3(i)}^2 + b_{\pi_3(i)}) + b_{\pi_3(u)}) \\ &\le C^*/3 + C^* = 4C^*/3; \end{aligned}$$

otherwise, since $\sum_{i=1}^{u}(s_{\pi_3(i)}^1 + a_{\pi_3(i)}) > s_1$, J_p is the first job in π_3 and $s_p^1 + a_p \le s_1$, then $u > 1$. Thus, we have

$$\begin{aligned} C_{max}(\pi_3) &= (\sum_{i=1}^{u}(s_{\pi_3(i)}^1 + a_{\pi_3(i)}) + \Delta_1) + (b_{\pi_3(u)} + \sum_{i=u+1}^{n}(s_{\pi_3(i)}^2 + b_{\pi_3(i)})) \\ &\le C^* + 2C^*/3 = 5C^*/3. \end{aligned}$$

For subcase $s_p^1 + a_p > s_1$, we have $s_p^1 + a_p + \Delta_1 + b_p \le C^*$. If the critical job does not exist or job J_p is the critical job, then we have

$$\begin{aligned} C_{max}(\pi_3) &= \max\{s_p^1 + a_p + \Delta_1, s_p^2\} + b_p + \sum_{J_i \in S \setminus J_p}(s_{\pi_3(i)}^2 + b_{\pi_3(i)}) \\ &\le C^* + 2C^*/3 = 5C^*/3; \end{aligned}$$

otherwise, for the critical job $J_{\pi_3(u)}$, $u > 1$, we have

$$\begin{aligned} C_{max}(\pi_3) &= (\sum_{i=1}^{u}(s_{\pi_3(i)} + a_{\pi_3(i)}) + \Delta_1) + b_{\pi_3(u)} + \sum_{i=u+1}^{n}(s_{\pi_3(i)}^2 + b_{\pi_3(i)}) \\ &\le C^* + 2C^*/3 = 5C^*/3. \end{aligned}$$

Case 2: $|S'|=2$

Similar to Case 1, it is not difficult to check that schedule π_2 or π_3 may yield a solution with an error bound of no more than 2C*/3. In the following, we further prove that the error bound of schedule π_4 obtained in Step (5) is no more than C*/3 for this case.

For schedule π_4, if no critical job exists, then

$$C_{max}(\pi_4) = \sum_{i=1}^{n}(s^2_{\pi_4(i)} + b_{\pi_4(i)}) = C^*;$$

otherwise, for the critical job $J_{\pi_4(u)}$, if $u > 2$, we have

$$\begin{aligned} C_{max}(\pi_4) &\le \sum_{i=1}^{u}(s^1_{\pi_4(i)} + a_{\pi_4(i)}) + \Delta_1 + (\sum_{i=u+1}^{n}(s^2_{\pi_4(i)} + b_{\pi_4(i)}) + b_{\pi_4(u)}) \\ &\le C^* + C^*/3 = 4C^*/3. \end{aligned}$$

If $u \le 2$, then u must be equal to 1; otherwise, a contradiction arises because $C^*/3 > \sum_{i=1}^{n}(s^1_i + a_i) > (s^1_p + a_p) + (s^1_q + a_q) \ge \min\{s^2_p + b_p, s^2_q + b_q\} > C^*/3$. Thus, we have

$$\begin{aligned} C_{max}(\pi_4) &\le \max\{s^1_p + a_p, s^1_q + a_q\} + (b_{\pi_4(1)} + \sum_{i=2}^{n}(s^2_{\pi_4(i)} + b_{\pi_4(i)})) \\ &\le C^*/3 + C^* = 4C^*/3. \quad \square \end{aligned}$$

From the proof of Theorem 1, we see that Steps (1)-(4) of Heuristic H1 can produce a solution with an error bound of no more than 2C*/3, and schedule π_4 in Step (5) can produce a solution with an error bound of no more than C*/3 in some special situations.

Although we do not know whether the bound is tight or not, the following instance shows that the worst-case error bound of H1 is no less than 1/2. Consider an instance with $s^1_1 = 4$, $a_1 = 4$, $s^2_1 = 7$, $b_1 = 3h$, $s^1_2 = h$, $a_2 = h$, $s^2_2 = 3h$, $b_2 = 8$, $s^1_3 = \varepsilon$, $a_3 = \varepsilon$, $s^2_3 = 1$, $b_3 = 1$, $s_1 = 8$, and $t_1 = 4h + 8$, where $h >> 1$ and $0 < \varepsilon < 8/(3h+7)$. It is easy to check that $\pi^* = [J_1, J_2, J_3]$ with $C^* = 6h + 18$ (see

Fig. 2(c)). Applying Heuristic H1, we obtain $\pi_1 = \pi_3 = [J_2, J_3, J_1]$ with $C_{max}(\pi_1) = C_{max}(\pi_3) = 9h + 17$ (see Fig. 2(a)), and $\pi_2 = [J_3, J_1, J_2]$ with $C_{max}(\pi_2) = 10h + 2\varepsilon + 16$ (see Fig. 2(b)). Since $(s_p^1 + a_p) + (s_q^1 + a_q) = 2h + 8 > s_1$, we need not consider Step (5) of H1. Thus, $C_{\mathrm{H1}} = 9h + 17$. Hence, we see that $(C_{\mathrm{H1}} - C^*)/C^*$ approaches 1/2 as h approaches infinity.

4. The unavailable interval is on M_2

In this section we provide a heuristic for the problem $F2/setup, r\text{-}a(M_2)/C_{max}$ and analyze its worst-case error bound.

Heuristic H2:

(1) Find two jobs J_p and J_q such that

$$s_p^2 + b_p \geq \max\{s_i^2 + b_i \mid J_i \in S \setminus \{J_p\}\}$$

and

$$s_q^1 + a_q \geq \max\{s_i^1 + a_i \mid J_i \in S \setminus \{J_q\}\}.$$

(2) Sequence the jobs by YHA. Let the corresponding schedule be π_1 and the corresponding makespan be $C_{max}(\pi_1)$.

(3) Sequence the jobs in nonincreasing order of $(s_i^2 + b_i)/(s_i^1 + a_i)$. Let the schedule be π_2 and the corresponding makespan be $C_{max}(\pi_2)$.

(4) Sequence job J_q in the last position, and sequence the remaining $n-1$ jobs by YHA. Let the corresponding schedule be π_3.

(5) Sequence job J_p in the first position, and sequence the remaining $n-1$ jobs in the same positions as those in Step (3). Let the corresponding schedule be π_4.

(6) Choose the schedule with the minimum makespan from the above four

schedules. Let $C_{\mathrm{H2}} = \min\{C_{max}(\pi_1), C_{max}(\pi_2), C_{max}(\pi_3), C_{max}(\pi_4)\}$.

Since Steps (2) and (3) of H2 dominate the algorithm, the complexity of Heuristic H2 is O($nlogn$).

For the problem $F2/setup, r\text{-}a(M_2)/C_{max}$, since an unavailable period exists on M_2, we assume that all the jobs must be processed on M_1 and M_2 as early as possible, and, for a given π, define again the critical job $J_{\pi(k)}$ in it as the last job in π such that its starting time on M_2 is equal to its finishing time on M_1 or the job in π before which the last idle time on M_2 occurs.

Lemma 2. For schedule π_2 defined in Step (3) of Heuristic H2, we assume that the completion time of the critical job $J_{\pi_2(k)}$ on M_1 is t and let π be a given schedule.

i) If $t \le s_2$ or $t > t_2$, let $J_{\pi(v)}$ be the last job that finishes no later than time t on M_1 in π, then $C_{max}(\pi_2) \le C_{max}(\pi) + b_{\pi_2(k)} + s^2_{\pi(v+1)}$.

ii) If $s_2 < t \le t_2$, let $J_{\pi_2(h)}$ be the job that finishes just before time s_2 on M_1 in π_2, and $J_{\pi(u)}$ the last job that finishes no later than $J_{\pi_2(h)}$ on M_1 in π, then $C_{max}(\pi_2) \le C_{max}(\pi) + (s^1_{\pi_2(h+1)} + a_{\pi_2(h+1)}) + (s^1_{\pi(u+1)} + a_{\pi(u+1)})$.

Proof i) Similar to the proof of Lemma 1.

ii) Let I_{π_2} be the total idle time on M_2 in π_2. Under the assumption that $J_{\pi_2(h)}$ finishes just before time s_2 on M_1 in π_2, we have $I_{\pi_2} \le s_2 - \sum_{j=1}^{h}(s^2_{\pi_2(j)} + b_{\pi_2(j)})$. Let I_π be the total idle time on M_2 in π. So,

$$I_\pi \ge s_2 - \sum_{j=1}^{u}(s^2_{\pi(j)} + b_{\pi(j)}) - (s^1_{\pi_2(h+1)} + a_{\pi_2(h+1)}) - (s^1_{\pi(u+1)} + a_{\pi(u+1)}).$$

Notice that since $\sum_{j=1}^{u}(s^1_{\pi(j)} + a_{\pi(j)}) \le \sum_{j=1}^{h}(s^1_{\pi_2(j)} + a_{\pi_2(j)})$ and all the jobs are sequenced in nonincreasing order of $(s_i^2 + b_i)/(s_i^1 + a_i)$ in π_2, it is not difficult to prove that $\sum_{j=h+1}^{n}(s_{\pi_2(j)}^2+b_{\pi_2(j)}) \le \sum_{j=u+1}^{n}(s_{\pi(j)}^2+b_{\pi(j)})$. We know that $C_{max}(\pi_2)=\sum_{j=1}^{n}(s_{\pi_2(j)}^2+b_{\pi_2(j)})+\Delta_2+I_{\pi_2}$ and $C_{max}(\pi)=\sum_{j=1}^{n}(s_{\pi(j)}^2+b_{\pi(j)})+\Delta_2+I_{\pi}$. Hence,

$$\begin{aligned}C_{max}(\pi_2) &\le s_2+\sum_{j=h+1}^{n}(s_{\pi_2(j)}^2+b_{\pi_2(j)})+\Delta_2\\ &\le s_2+\sum_{j=u+1}^{n}(s_{\pi(j)}^2+b_{\pi(j)})+\Delta_2\\ &= C_{max}(\pi)+(s_2-\sum_{j=1}^{u}(s_{\pi(j)}^2+b_{\pi(j)})-I_{\pi})\\ &\le C_{max}(\pi)+(s_{\pi_2(h+1)}^1+a_{\pi_2(h+1)})+(s_{\pi(u+1)}^1+a_{\pi(u+1)}).\end{aligned}$$

This completes the proof. □

The following theorem establishes the worst-case error bound of Heuristic H2 for the resumable case.

Theorem 2. For the problem *F2/setup, r-a*(M_2)/C_{max}, $(C_{\mathrm{H2}}-C^*)/C^*\le 2/3$.

Proof We know that YHA can produce an optimal solution for $F2/permu, setup/C_{max}$. Since when $t_2=0$, $F2/setup, r-a(M_2)/C_{max}$ is equivalent to $F2/permu, setup/C_{max}$, it is obvious that $C_{max}(\pi_1)-C^*\le t_2$. If $t_2\le 2C^*/3$, then we are done. So, in the following, we focus on the case where $t_2>2C^*/3$.

Let $S'=\{J_i \mid s_i^1+a_i>C^*/3, i=1, 2, \cdots, n\}$ and $S''=\{J_i \mid s_i^2+b_i>C^*/3, i=1, 2, \cdots, n\}$. We can easily show that $|S'|\le 2$ and $|S''|\le 2$ from the lower bound $\max\{\sum_{i=1}^{n}(s_i^1+a_i), \sum_{i=1}^{n}(s_i^2+b_i)+\Delta_2\}\le C^*$. When $|S'|=0$ and $|S''|=0$, from i) and ii) of Lemma 2, we have $C_{max}(\pi_2)\le 5C^*/3$. Hence, in the remainder of proof, we only need to consider the following two situations.

Case 1: $|S''|=0$ and $|S'|>0$

In this case, we consider schedule π_3. If no critical job exists in π_3, then $C_{max}(\pi_3) = \sum_{i=1}^{n}(s^2_{\pi_3(i)} + b_{\pi_3(i)}) + \Delta_2 = C^*$. Next, we assume that there exists a critical job in π_3. Let J_q be the critical job, then

$$C_{max}(\pi_3) \le \max\{\sum_{i=1}^{n}(s^1_{\pi_3(i)} + a_{\pi_3(i)}), t_2\} + b_q \le C^* + C^*/3 = 4C^*/3.$$

Otherwise, suppose $J_{\pi_3(k)}$ ($k < n$) is the critical job, then we have

$$C_{max}(\pi_3) \le \sum_{i=1}^{k}(s^1_{\pi_3(i)} + a_{\pi_3(i)}) + (\Delta_2 + b_{\pi_3(k)} + \sum_{i=k+1}^{n}(s^2_{\pi_3(i)} + b_{\pi_3(i)}))$$
$$\le 2C^*/3 + C^* = 5C^*/3.$$

Case 2: $|S''| \ge 1$

We check schedule π_4 obtained in Step (5) of Heuristic H2. If no critical job exists in π_4, then $C_{max}(\pi_4) = \sum_{i=1}^{n}(s^2_{\pi_4(i)} + b_{\pi_4(i)}) + \Delta_2 = C^*$. In the following, we assume that there exists a critical job in π_4. Since $|S''| \ge 1$, we assume that $s^2_p + b_p > C^*/3$ for J_p. If

$$\sum_{i=1}^{n}(s^1_i + a_i) \ge \max\{\max\{s^1_p + a_p, s^2_p\} - \max\{s^1_p + a_p - s_2, 0\}, s^2_p\} + b_p + \alpha\Delta_2,$$

where $\alpha = 1$ if $s_2 < \max\{s^1_p + a_p, s^2_p\} + b_p$; otherwise, $\alpha = 0$. Then, we have

$$C_{max}(\pi_4) \le \sum_{i=1}^{n}(s^1_i + a_i) + (1-\alpha)\Delta_2 + \sum_{J_i \in S\backslash\{J_p\}}(s^2_i + b_i)$$
$$\le C^* + 2C^*/3 = 5C^*/3.$$

Otherwise, J_p is the critical job. From $s^2_p + b_p > C^*/3$ and $\max\{s^1_p + a_p, s^2_p\} + b_p < C^*$, we obtain that $\max\{s^1_p + a_p - s^2_p, 0\} < 2C^*/3$; so

$$C_{max}(\pi_4) \le \max\{s^1_p + a_p - s^2_p, 0\} + \Delta_2 + \sum_{i=1}^{n}(s^2_i + b_i) < 2C^*/3 + C^* = 5C^*/3.$$

The proof is completed. □

Although we do not know whether the bound is tight or not, the following instance shows that the worst-case error bound of H2 is no less than 1/3. Consider an instance with $s^1_1 = h-2$, $a_1 = 1$, $s^2_1 = h-3$, $b_1 = 1$, $s^1_2 = h/2+1$, $a_2 = h/2+1$,

$s_2^2 = 1$, $b_2 = h + 2$, $s_3^1 = h$, $a_3 = 1$, $s_3^2 = 1$, $b_3 = 5$, $s_2 = h$, and $t_2 = 2h$, where $h >> 1$. It is easy to check that $\pi^* = [J_1, J_2, J_3]$ with $C^* = 3h + 9$ (see Fig. 3(d)). Applying Heuristic H2, we obtain $\pi_1 = [J_2, J_3, J_1]$ with $C_{max}(\pi_1) = 4h + 6$ (see Fig. 3(a)), $\pi_2 = \pi_4 = [J_2, J_1, J_3]$ with $C_{max}(\pi_2) = C_{max}(\pi_4) = 4h + 6$ (see Fig. 3(b)), and $\pi_3 = [J_3, J_1, J_2]$ with $C_{max}(\pi_3) = 4h + 6$ (see Fig. 3(c)). Thus, $C_{\mathrm{H2}} = 4h + 6$. Hence, we see that $(C_{\mathrm{H2}} - C^*)/C^*$ approaches 1/3 as h approaches infinity.

5. Conclusions

In this paper we studied the two-machine flowshop scheduling problem with anticipatory setup times and a resumable availability constraint imposed on only one of the machines. Since the problem is NP-hard, we developed two polynomial-time heuristics and analyzed their worst-case error bounds.

Acknowledgement

This research was supported in part by The Hong Kong Polytechnic University under a grant from the *Area of Strategic Development in China Business Services*.

References

[1] C.-Y. Lee, L. Lei, M. Pinedo, Current trends in deterministic scheduling, Annals of Operations Research, 70 (1997) 1-41.

[2] E. Sanlaville, G. Schmidt, Machine scheduling with availability constraints, Acta Informatica, 35 (1998) 795-811.

[3] G. Schmidt, Scheduling with limited availability, European Journal of Operational Research, 121 (2000) 1-15.

[4] C.-Y. Lee, Minimizing the makespan in the two-machine flowshop scheduling problem with an availability constraint, Operations Research Letters, 20 (1997) 129-139.

[5] C.-Y. Lee, Two-machine flowshop scheduling with availability constraints, European Journal of Operational Research, 114 (1999) 420-429.

[6] T.C.E. Cheng, G. Wang, An improved heuristic for two-machine flowshop scheduling with an availability constraint, Operations Research Letters, 26 (2000) 223-229.

[7] J. Breit, An improved approximation algorithm for two-machine flow shop scheduling with an availability constraint, Information Processing Letters, 90 (2004) 273-278.

[8] T.C.E. Cheng, G. Wang, Two-machine flowshop scheduling with consecutive availability constraints, Information Processing Letters, 71 (1999) 49-54.

[9] T.C.E. Cheng, Z. Liu, Approximability of two-machine no-wait flowshop scheduling with availability constraints, Operations Research Letters, 31 (2003) 319-322.

[10] T.C.E. Cheng, Z. Liu, 3/2-approximation for two-machine no-wait flowshop scheduling with availability constraints, Information Processing Letters, 88 (2003) 161-165.

[11] R. Aggoune, Minimizing the makespan for the flow shop scheduling problem with availability constraints, European Journal of Operational Research, 153 (2004) 534-543.

[12] C. Rajedran, H. Ziegler, Heuristics for scheduling in a flow shop with setup, processing and removal times separated, Production Planning and Control, 8 (1997) 568-576.

[13] D.R. Sule, KY Huang, Sequencing on two and three machines with setup, processing and removal times separated. International Journal of Production Research, 21 (1983) 723-32.

[14] T. Yoshida, K. Hitomi, Optimal two-stage production scheduling with setup times separated, AIIE Transactions, 11 (1979) 261-263.

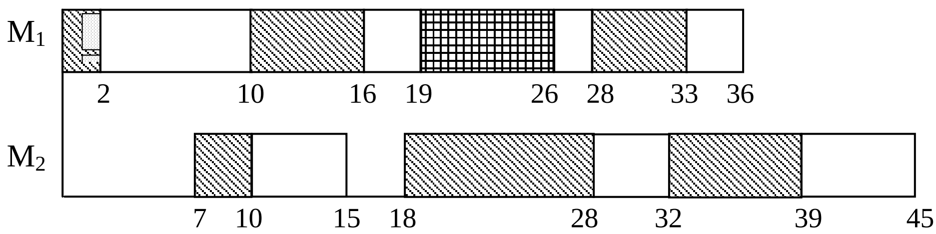
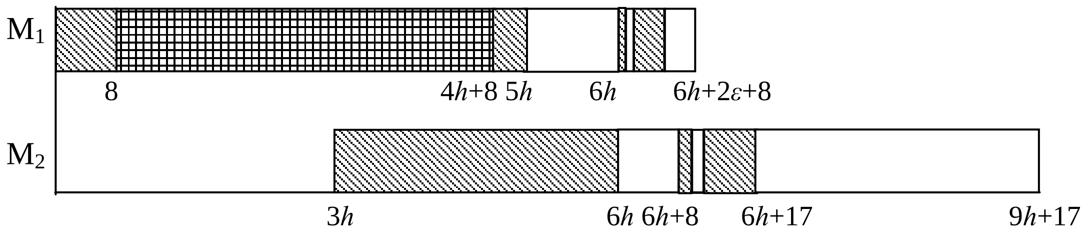
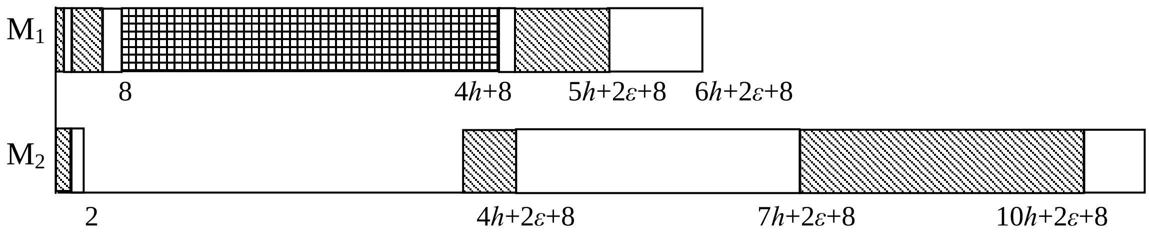


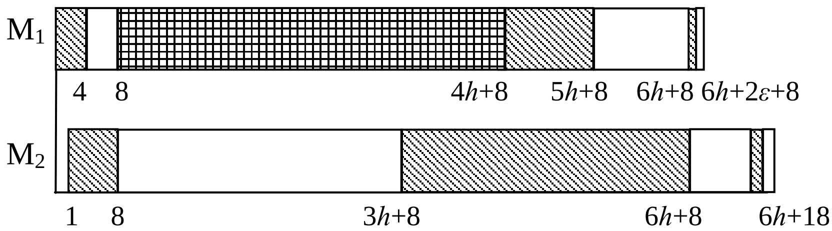
Fig. 1. A schedule π for the example instance.



(a) Schedule $\pi_1 = \pi_3 = [J_2, J_3, J_1]$.

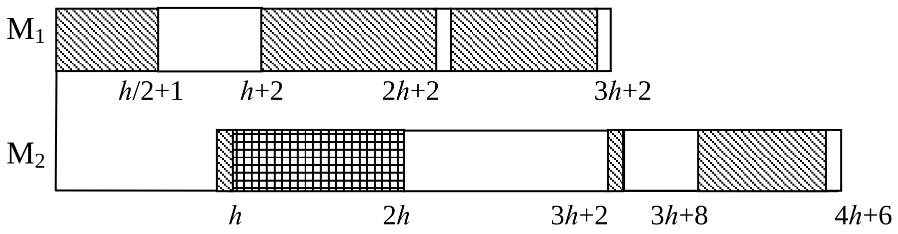


(b) Schedule $\pi_2 = [J_3, J_1, J_2]$.

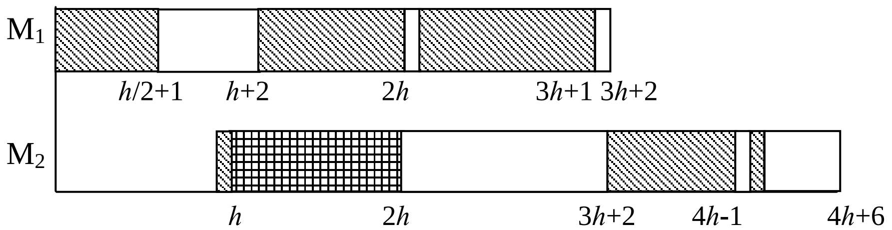


(c) Optimal schedule $\pi^* = [J_1, J_2, J_3]$.

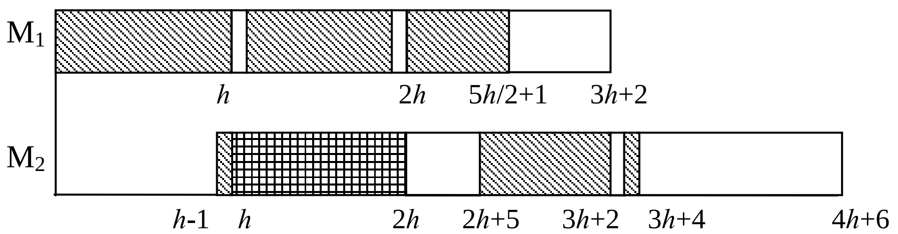
Fig. 2(a). Solution of Steps (2) and (4); Fig. 2(b). Solution of Step (3); Fig. 2(c). Optimal solution.



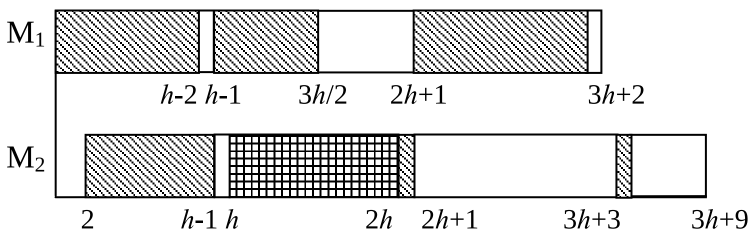
(a) Schedule $\pi_1 = [J_2, J_3, J_1]$.



(b) Schedule $\pi_2 = \pi_4 = [J_2, J_1, J_3]$.



(c) Schedule $\pi_3 = [J_3, J_1, J_2]$.



(d) Optimal schedule $\pi^* = [J_1, J_2, J_3]$.

Fig. 3(a). Solution of Step (2); Fig. 3(b). Solution of Steps (3) and (5); Fig. 3(c). Solution of Step (4); Fig. 3(d). Optimal solution.