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

Wei Cheng

A Thesis Submitted to the
University of North Carolina Wilmington in Partial Fulfillment
Of the Requirements for the Degree of
Master of Science

Department of Mathematics and Statistics
University of North Carolina Wilmington

2007

Approved by

Advisory Committee

__________________________
Chair

__________________________
Accepted by

Dean, Graduate School
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>ii</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>iii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>iv</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>v</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2 NOTATION AND PRELIMINARIES</td>
<td>3</td>
</tr>
<tr>
<td>3 UNAVAILABLE INTERVAL ON $M_1$</td>
<td>4</td>
</tr>
<tr>
<td>3.1 YHA algorithm ($\pi_1$)</td>
<td>4</td>
</tr>
<tr>
<td>3.2 Decreasing ratio ($\pi_2$)</td>
<td>5</td>
</tr>
<tr>
<td>3.3 Largest job $p, q$ on machine 2 ($\pi_3$)</td>
<td>5</td>
</tr>
<tr>
<td>3.4 Random sequence $p$ ($\pi_4$, $\pi_5$)</td>
<td>6</td>
</tr>
<tr>
<td>3.5 Heuristic H1:</td>
<td>6</td>
</tr>
<tr>
<td>4 UNAVAILABLE INTERVAL ON $M_2$</td>
<td>16</td>
</tr>
<tr>
<td>4.1 YHA algorithm ($\pi_1$)</td>
<td>16</td>
</tr>
<tr>
<td>4.2 Decreasing ratio ($\pi_2$)</td>
<td>16</td>
</tr>
<tr>
<td>4.3 Largest job $q$ on machine 1 ($\pi_3$)</td>
<td>17</td>
</tr>
<tr>
<td>4.4 Largest job $p$ on machine 2 ($\pi_4$)</td>
<td>17</td>
</tr>
<tr>
<td>4.5 Heuristic H2:</td>
<td>18</td>
</tr>
<tr>
<td>5 COMPUTATIONAL RESULTS</td>
<td>25</td>
</tr>
<tr>
<td>6 CONCLUSIONS</td>
<td>27</td>
</tr>
</tbody>
</table>
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 a heuristic algorithm from Wang and Cheng to minimize makespan and use simulation to determine the actual error bound.
I would like to express my sincere appreciation to Dr. John Karlof for guidance on my thesis work. I also want to thank Dr. Matthew TenHuisen and Dr. Yaw Chang for serving on my thesis advisory committee and offering constructive ideas. Also, I want to thank all the faculty members in the Mathematics and Statistics Department who gave me valuable advice and help. I also want to thank my wife Lifang Du for her patience and help. This thesis is dedicated to my parents.
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Values considered in $\pi_1$</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>Values considered in $\pi_2$</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Values considered in $\pi_3$</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>Values considered in $\pi_1$</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>Values considered in $\pi_2$</td>
<td>16</td>
</tr>
<tr>
<td>6</td>
<td>Values considered in $\pi_3$</td>
<td>17</td>
</tr>
<tr>
<td>7</td>
<td>Values considered in $\pi_4$</td>
<td>17</td>
</tr>
<tr>
<td>8</td>
<td>Computational results for heuristic 1</td>
<td>26</td>
</tr>
<tr>
<td>9</td>
<td>Computational results for heuristic 2</td>
<td>27</td>
</tr>
</tbody>
</table>
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Example of $F2/setup, r - a(M_1)/C_{max}$, where the 10-15 area on $M_1$ is the unavailable time</td>
</tr>
<tr>
<td>2</td>
<td>(a)Solution of order $\pi_1$; (b)Solution of order $\pi_2$; (c)Solution of order $\pi_3$ and $\pi_4$; (d)Solution of order $\pi_5$</td>
</tr>
<tr>
<td>3</td>
<td>Illustrations of (1), $J_k = b_{\pi_2(k)}$</td>
</tr>
<tr>
<td>4</td>
<td>Illustrations of (2), (a)Order $\pi_2$; (b)Order $\pi_3$</td>
</tr>
<tr>
<td>5</td>
<td>a is for $\pi^\star$ which is no idle time at all, b is for $\pi_1$, $\Delta_1 = t_1 - s_1$.</td>
</tr>
<tr>
<td>6</td>
<td>Illustrations of equation [6] of $\pi_3$; $J_u$ on $M_2$ equal to $b_{\pi_3(u)}$</td>
</tr>
<tr>
<td>7</td>
<td>Illustrations of equation [7] of $\pi_3$; $J_u$ on $M_2$ equal to $b_{\pi_3(u)}$</td>
</tr>
<tr>
<td>8</td>
<td>Illustrations of equation [8] of $\pi_3$; compare $\max{s^1_p + a_p + \Delta_1, s^2_p}$. $s^1_p + a_p + \Delta_1$ in (a), $s^2_p$ in (b).</td>
</tr>
<tr>
<td>9</td>
<td>Illustrations of equation [9] of $\pi_3$; $J_u$ on machine 2 equal to $b_{\pi_3(u)}$.</td>
</tr>
<tr>
<td>10</td>
<td>Illustrations of equation [10] of $\pi_4$; this means no idle time on $M_2$.</td>
</tr>
<tr>
<td>12</td>
<td>Illustrations of equation of $\pi_4$ [12] and $\pi_5$ [13].</td>
</tr>
<tr>
<td>13</td>
<td>(a)Solution of order $\pi_1, \pi_3$; (b)Solution of order $\pi_2$; (c)Solution of order $\pi^\star$.</td>
</tr>
<tr>
<td>14</td>
<td>(a)Solution of order $\pi_1$; (b)Solution of order $\pi_2$; (c)Solution of order $\pi_3$; (d)Solution of order $\pi_4$</td>
</tr>
<tr>
<td>15</td>
<td>(a)$J_h$ finished after $s_2$ on $M_2$; (a)$J_h$ finished before $s_2$ on $M_2$.</td>
</tr>
<tr>
<td>16</td>
<td>(a)$J_q$ finished before $t_2$ on $M_1$; (a)$J_q$ finished after $t_2$ on $M_1$.</td>
</tr>
<tr>
<td>17</td>
<td>$\Delta_2 = t_2 - s_2$.</td>
</tr>
<tr>
<td>18</td>
<td>Here has four conditions a,b,c,d.</td>
</tr>
<tr>
<td>19</td>
<td>(a)Solution of order $\pi_1$; (b)Solution of order $\pi_2, \pi_4$; (c)Solution of order $\pi_3$; (d)Solution of order $\pi^\star$.</td>
</tr>
</tbody>
</table>
1 INTRODUCTION

The subject of machine scheduling problems with availability constraints has attracted much research attention over the years. The two-machine flowshop scheduling problem with availability constraints was first studied by Lee [7]. Under the resumable assumption, he proved that the problem is NP-hard when an availability constraint is imposed on only one machine. 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 $\frac{1}{2}$. The second heuristic is for solving the problem where the availability constraint is imposed on machine 2, which has a worst-case error bound $\frac{1}{3}$. Lee [8] further studied and developed a pseudo-polynomial dynamic programming algorithm and heuristics. For the resumable case, Cheng and Wang [3] developed an improved heuristic when the availability constraint is imposed on the first machine, and the heuristic has a worst-case error bound $\frac{1}{3}$. Breit [2] 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 $\frac{1}{4}$. Cheng and Wang [4] 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 $\frac{2}{3}$. 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 [5,6]. For the general flowshop scheduling problem with availability constraints, Aggoune [1] proposed a heuristic based on a genetic algorithm and a tabu search.

Definition 1 The objective is to minimize total completion time, called the makespan.

Definition 2 Error bound $= (C_{Hi} - C^\star)/C^\star$
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 (for example [9,10]). The two-machine flowshop scheduling problem with anticipatory setup times, where an availability constraint is imposed on only one machine has been studied by Wang and Cheng [11]. They study the cases where the availability constraint is imposed on machines 1 and 2 and present two heuristics and show that their worst-case error bounds are no larger than $\frac{2}{3}$.

In this paper, we present the heuristic algorithm developed by Wang and Cheng for the two-machine flowshop scheduling problem with setup times where an availability constraint is imposed on machines 1 and 2. In section 2, we introduce the notation and present the parallel machine scheduling problem with the unavailable time on machine 1. In section 3, we present a heuristic for minimizing the makespan for the case where the availability constraint is imposed on machine 1. We first introduce the Yoshida and Hitomi algorithm [12] for the classical two-machine permutation flowshop scheduling problem with setup times and no unavailable time, then present a lemma, the heuristic algorithm of Wang and Cheng, and show that its worst-error bound is no larger than $\frac{2}{3}$. In section 4, we study the case where the availability constraint is imposed on machine 2, present a lemma and algorithm we also show that the worst-error bound is no larger than $\frac{2}{3}$. In section 5, we program the heuristic in JAVA and estimate the actual error bound by simulation for both cases.
For the problem under consideration, we introduce the following notation to be used throughout this paper.

- \( S = J_1, ..., J_n \): 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 \( t_l \), \( 0 \leq s_l \leq t_l \), \( l = 1, 2 \);
- \( s^1_i, s^2_i \): setup times of \( J_i \) on \( M_1 \) and \( M_2 \), respectively, where \( s^1_i > 0, s^2_i > 0 \);
- \( a_i, b_i \): processing times of \( J_i \) on \( M_1 \) and \( M_2 \), respectively, where \( a_i > 0, b_i > 0 \);
- \( \pi: = [J_{\pi(1)}, ..., J_{\pi(n)}] \): a permutation schedule, where \( J_{\pi(i)} \) is the \( i \)th job in \( \pi \);
- \( \pi^\star \): an optimal schedule;
- \( C_{H_x} \): the makespan yielded by heuristic \( H_x \);
- \( C^\star \): the optimal makespan.
- \( F2/\text{setup}, r - a(M_1)/C_{\text{max}} \): the makespan minimization problem in a two-machine flowshop with setup times and a resumable availability constraint on \( M_i \).

Fig. 1 A schedule \( \pi \) for the example.

As an example, consider a problem instance of \( F2/\text{setup}, r - a(M_1)/C_{\text{max}} \) with \( n = 3 \). Let \( s^1_1 = 3, a_1 = 4, s^2_1 = 5, a_2 = 4, s^1_3 = 4, a_3 = 5, s^2_3 = 2, b_1 = 6, s^1_2 = 4, b_2 = 8, s^2_3 = 2, b_3 = 3, s_1 = 10, \) and \( t_1 = 15 \). A schedule \( \pi = [J_1, J_2, J_3] \) for the instance is shown in Fig. 1.
3 UNAVAILABLE INTERVAL ON $M_1$

In this section we present a heuristic for the problem $F_2/\text{setup}, r - a(M_1)/C_{\text{max}}$ and evaluate its worst-case error bound by Wang and Cheng [11]. The basic ideas of this 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. They developed the schedules $\pi_1, \pi_2, \pi_3, \pi_4, \pi_5$ and then choose the one with the shortest makespan.

3.1 YHA algorithm ($\pi_1$)

The Yoshida and Hitomi algorithm (YHA) works in the following manner:

Divide $S$ into two disjoint subsets $A$ and $B$, where $A = \{J_i|s_i^1 + a_i - s_i^2 \leq b_i\}$ and $B = \{J_i|s_i^1 + a_i - s_i^2 > b_i\}$. 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 $B$.

Let $s_1^1 = 9, a_1 = 3, s_2^1 = 2, a_2 = 4, s_3^1 = 3, a_3 = 2, s_1^2 = 7, b_1 = 4, s_2^2 = 1, b_2 = 7, s_3^2 = 2, b_3 = 3, s_1 = 20, \text{ and } t_1 = 25$.

Then, $J_2, J_3 \in A$, and $J_1 \in B$. Because $s_2^1 + a_2 - s_2^2 > s_3^1 + a_3 - s_3^2$, then the order in set $A$ will be $\{J_3, J_2\}$ (nondecreasing order). The final order will be $\pi_1\{J_3, J_2, J_1\}$. See Figure 2(a).
<table>
<thead>
<tr>
<th>Job number</th>
<th>Set A</th>
<th>Set B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>None</td>
<td>$s_1 + a_1 - s_1^2 = 9 + 3 - 7 = 5 &gt; 4$</td>
</tr>
<tr>
<td>2</td>
<td>$s_1^2 + a_2 - s_2 = 2 + 4 - 1 = 5 &lt; 7$</td>
<td>None</td>
</tr>
<tr>
<td>3</td>
<td>$s_2^1 + a_3 - s_3^2 = 3 + 2 - 2 = 3 &lt; 3$</td>
<td>None</td>
</tr>
</tbody>
</table>

Table 1: Values considered in $\pi_1$

3.2 Decreasing ratio($\pi_2$)

Next we sequence the jobs in nonincreasing order of $(s_i^2 + b_i)/(s_i^1 + a_i)$.

<table>
<thead>
<tr>
<th>Job number</th>
<th>$(s_i^2 + b_i)/(s_i^1 + a_i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$(s_1^2 + b_1)/(s_1^1 + a_1) = 11/12$</td>
</tr>
<tr>
<td>2</td>
<td>$(s_2^2 + b_2)/(s_2^1 + a_2) = 8/6$</td>
</tr>
<tr>
<td>3</td>
<td>$(s_3^2 + b_3)/(s_3^1 + a_3) = 5/5$</td>
</tr>
</tbody>
</table>

Table 2: Values considered in $\pi_2$

Then we get $\frac{s_2^2 + b_2}{s_2^1 + a_2} > \frac{s_3^2 + b_3}{s_3^1 + a_3} > \frac{s_1^2 + b_1}{s_1^1 + a_1}$. So the order will be $\pi_2 \{J_2, J_3, J_1\}$. See Figure 2(b).

3.3 Largest job $p, q$ on machine 2 ($\pi_3$)

Next we need find jobs $J_p$ and $J_q$ such that

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

<table>
<thead>
<tr>
<th>Job number</th>
<th>$s_i^2 + b_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$s_1^2 + b_2 = 7 + 4 = 11$</td>
</tr>
<tr>
<td>2</td>
<td>$s_2^2 + b_2 = 1 + 7 = 8$</td>
</tr>
<tr>
<td>3</td>
<td>$s_3^2 + b_3 = 2 + 3 = 5$</td>
</tr>
</tbody>
</table>

Table 3: Values considered in $\pi_3$

Let $p = 1$ and $q = 2$, For $\pi_3$ put job $J_p$ first and keep other $n - 1$ jobs in the same order as $\pi_2$. Then the order will be $\pi_3 \{J_1, J_2, J_3\}$. See Figure 2(c).
3.4 Random sequence $p(\pi_4, \pi_5)$

Test if $(s_{p_1}^1 + a_p) + (s_{q_1}^1 + a_q) \leq s_1$ if not then no $\pi_4, \pi_5$, otherwise make two sequences

$\pi_4$: Place $J_p$ and $J_q$ as the first two jobs. the remaining $n - 2$ jobs are sequenced randomly. $\pi_4 \{J_1, J_2, J_3\}$. See Figure 2(c).

$\pi_5$: Place $J_q$ and $J_p$ as the first two jobs. the remaining $n - 2$ jobs are sequenced randomly. $\pi_5 \{J_2, J_1, J_3\}$. See Figure 2(d).

3.5 Heuristic $H_1$:

(1) Find jobs $J_p$ and $J_q$ such that

Figure 2: (a) Solution of order $\pi_1$; (b) Solution of order $\pi_2$; (c) Solution of order $\pi_3$ and $\pi_4$; (d) Solution of order $\pi_5$
\[ s_p^2 + b_p \geq s_q^2 + b_q \geq \max\{s_i^2 + b_i | J_i \in S \setminus \{J_p, J_q\}\}. \]

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

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

(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 \( \pi_3 \) and the corresponding makespan be \( C_{\text{max}}(\pi_3) \).

(5) If \((s_p^1 + a_p) + (s_q^1 + a_q) \leq s_1\), then sequence jobs \( J_p, J_q \) as the first two jobs. The remaining \( n - 2 \) jobs are sequenced randomly. Let the corresponding schedule be \( \pi_4 \) and the corresponding makespan be \( C_{\text{max}}(\pi_4) \).

(6) If \((s_p^1 + a_p) + (s_q^1 + a_q) \leq s_1\), then sequence jobs \( J_q, J_p \) as the first two jobs. The remaining \( n - 2 \) jobs are sequenced randomly. Let the corresponding schedule be \( \pi_5 \) and the corresponding makespan be \( C_{\text{max}}(\pi_5) \).

(7) Select the schedule with the minimum makespan from the above five schedules. Let \( C_{H1} = \min\{C_{\text{max}}(\pi_1), C_{\text{max}}(\pi_2), C_{\text{max}}(\pi_3), C_{\text{max}}(\pi_4), C_{\text{max}}(\pi_5)\} \).

In the following, we analyze the performance bound of heuristic H1.

**Definition 3** Let \( \pi \) be a schedule for the problem \( F2/\text{setup}, r - a(M_1)/C_{\text{max}} \). We define the critical job \( J_{\pi(k)} \) as the last job such that its starting time on \( M_2 \) is equal to its finishing time on \( M_1 \).

**Lemma 1** For schedule \( \pi_2 \) 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 \( \pi \). The following inequality holds:
\[ C_{\text{max}}(\pi_2) \leq C_{\text{max}}(\pi) + b_{\pi_2(k)} + s_{\pi_2(v+1)}^2. \]

**Proof.** For schedule \( \pi_2 \), its makespan is

\[ C_{\text{max}}(\pi_2) = t + b_{\pi_2(k)} + \sum_{j=k+1}^{n} (s_{\pi_2(j)}^2 + b_{\pi_2(j)}). \tag{1} \]

Since on machine 2 there will be no idle time after \( J_k \), because of the definition of the critical job.

![Figure 3: Illustrations of (1), \( J_k = b_{\pi_2(k)} \)]

Under the assumption of lemma 1, \( J_{\pi(v)} \) is the last job that finishes no later than time \( t \) on \( M_1 \) in a schedule \( \pi \). We have

\[ \sum_{j=1}^{v} (s_{\pi(j)}^1 + a_{\pi(j)}) \leq \sum_{j=1}^{k} (s_{\pi_2(j)}^1 + a_{\pi_2(j)}), \]

and because \( \sum_{j=1}^{n} (s_{\pi(j)}^1 + a_{\pi(j)}) = \sum_{j=1}^{n} (s_{\pi_2(j)}^1 + a_{\pi_2(j)}) \), obviously

\[ \sum_{j=v+1}^{n} (s_{\pi(j)}^1 + a_{\pi(j)}) \geq \sum_{j=k+1}^{n} (s_{\pi_2(j)}^1 + a_{\pi_2(j)}). \tag{2} \]

Since all the jobs are sequenced in nonincreasing order of \( (s_{\pi_2(j)}^2 + b_{\pi_2(j)})/(s_{\pi_2(j)}^1 + b_{\pi_2(j)}) \) in \( \pi_2 \), and because after critical job \( k \) on \( M_1 \), there is no idle time, we have

\[ \sum_{j=k+1}^{n} (s_{\pi_2(j)}^2 + b_{\pi_2(j)}) > \sum_{j=k+1}^{n} (s_{\pi_2(j)}^1 + a_{\pi_2(j)}). \tag{3} \]
From (2) and (3)

$$\sum_{j=\nu+1}^{n} (s_{\pi(j)}^2 + b_{\pi(j)}) \geq \sum_{j=k+1}^{n} (s_{\pi_2(j)}^2 + b_{\pi_2(j)}).$$  \hspace{1cm} (4)

For schedule \(\pi\), we have

$$C_{\text{max}}(\pi) \geq t + \sum_{j=\nu+1}^{n} (s_{\pi_2(j)}^2 + b_{\pi_2(j)}) - s_{\pi(v+1)}^2.$$  \hspace{1cm} (5)

Therefore, from (1), (4) and (5), we have

$$C_{\text{max}}(\pi_2) = t + b_{\pi_2(k)} + \sum_{j=k+1}^{n} (s_{\pi_2(j)}^2 + b_{\pi_2(j)})$$

$$\leq t + b_{\pi_2(k)} + \sum_{j=\nu+1}^{n} (s_{\pi(j)}^1 + a_{\pi(j)})$$

$$\leq C_{\text{max}}(\pi) + b_{\pi_2(k)} + s_{\pi(v+1)}^2.$$  

**Theorem 1** For the problem \(F2/setup, r - a(M_l)/C_{\text{max}}, (CH_1 - C^\star)/C^\star \leq 2/3.\)

**Proof.** If \(\sum_{i=1}^{n} (s_i^1 + a_i) \leq s_1\), it is obvious that \(C_{\text{max}}(\pi_1) = C^\star\) from Yoshida and Hitomi algorithm (YHA)[11]. So we assume \(\sum_{i=1}^{n} (s_i^1 + a_i) > s_1.\)
Notice that since all the jobs are resumable for the problem $F2/\text{setup}, r - a(M_1)/C_{\text{max}}$, then $\pi_1$ is the best schedule without an unavailable time then we have $C_{\text{max}}(\pi_1) \leq C^* + \Delta_1$. See figure 5.

![Machine Diagram](image)

Figure 5: a is for $\pi^*$ which is no idle time at all, b is for $\pi_1$, $\Delta_1 = t_1 - s_1$.

If $\Delta_1 \leq 2C^*/3$, then we are done. So, in the following, we focus on the situation where $\Delta_1 > 2C^*/3$.

Because $\Delta_1 > 2C^*/3$ and $\sum_{i=1}^{n}(s_1^i + a_i) + \Delta_1 < C^*$, we have $\sum_{i=1}^{n}(s_1^i + a_i) < C^*/3$. Let $S' = \{J_i|s_1^i + b_i > C^*/3, i = 1, 2, ..., n\}$. It is obvious $|S'| \leq 2$.

Case 1: $|S'| = 0$

For an optimal schedule $\pi^*$, according to lemma 1, we have $C_{\text{max}}(\pi_2) \leq C^* + b_{\pi_2(k)} + s_1^2(\pi_2(k)) < 5C^*/3$.

Case 2: $|S'| = 1$

In this case, $S' = \{J_p\}$. If $s_p^2 \leq C^*/3$ and $b_p \leq C^*/3$, then $b_{\pi_2(k)} \leq b_p \leq C^*/3$ and $s_1^2(\pi_2) \leq s_1^2(\pi_2) \leq C^*/3$, then from lemma 1 $C_{\text{max}}(\pi_2) \leq C^* + C^*/3 + C^*/3 \leq 5C^*/3$, we are done. Otherwise at least one of $s_p^2 \geq C^*/3$ or $b_p \geq C^*/3$ will be exist, then we consider schedule $\pi_3$ of Heuristic H1.

For subcase $s_1^1 + a_p \leq s_1$, suppose that the critical job does not exist in $\pi_3$, then there is no idle time on machine 2 that implies $C_{\text{max}}(\pi_3) = \sum_{i=1}^{n}(s_1^2(\pi_3(i)) + b_{\pi_3(i)}) = C^*$.

Otherwise, we denote the critical job as $J_{\pi_3(u)}$. If $\sum_{i=1}^{u}(s_1^1(\pi_3(i)) + a_{\pi_3(i)}) \leq s_1$, see figure 6. then
Figure 6: Illustrations of equation [6] of $\pi_3$; $J_u$ on $M_2$ equal to $b_{\pi_3(u)}$.

\[
C_{\text{max}}(\pi_3) = \sum_{i=1}^{u} (s_{1_{\pi_3(i)}} + a_{\pi_3(i)}) + \sum_{i=u+1}^{n} (s_{2_{\pi_3(i)}} + b_{\pi_3(i)}) + b_{\pi_3(u)} \\
\leq C^\star/3 + C^\star = 4C^\star/3 \tag{6}
\]

Otherwise, let $\sum_{i=1}^{u} (s_{1_{\pi_3(i)}} + a_{\pi_3(i)}) > s_1$, $J_p$ is the first job in $\pi_3$ and $s_1 + a_p \leq s_1$, then $u > 1$. see figure 7. Thus, we have

\[
C_{\text{max}}(\pi_3) = \sum_{i=1}^{u} (s_{1_{\pi_3(i)}} + a_{\pi_3(i)}) + \Delta_1 + \sum_{i=u+1}^{n} (s_{2_{\pi_3(i)}} + b_{\pi_3(i)}) + b_{\pi_3(u)} \\
\leq C^\star + 2C^\star/3 = 5C^\star/3 \tag{7}
\]

For subcase $s_1^p + a_p > s_1$, we have $s_1^p + a_p + \Delta_1 + b_p \leq C^\star$. If the critical job does not exist or job $J_p$ is the critical job, see figure 8. then we have
Figure 8: Illustrations of equation [8] of \( \pi_3 \); compare \( \max \{ s_p^1 + a_p + \Delta_1, s_p^2 \} \).

\[
C_{\text{max}}(\pi_3) = \max \{ s_p^1 + a_p + \Delta_1, s_p^2 \} + b_p + \sum_{J \in S \setminus J_p} (s_{\pi_3(i)}^2 + b_{\pi_3(i)}) \\
\leq C^\bullet + \frac{2C^\bullet}{3} = \frac{5C^\bullet}{3} \quad (8)
\]

Otherwise, for the critical job \( J_{\pi_3(u)}, u > 1 \), see figure 9, we have

Figure 9: Illustrations of equation [9] of \( \pi_3 \); \( J_u \) on machine 2 equal to \( b_{\pi_3(u)} \).

\[
C_{\text{max}}(\pi_3) = \left( \sum_{i=1}^{u} (s_{\pi_3(i)}^1 + a_{\pi_3(i)}) + \Delta_1 \right) + b_{\pi_3(u)} + \sum_{i=1}^{n} (s_{\pi_3(i)}^2 + b_{\pi_3(i)}) \\
\leq C^\bullet + \frac{2C^\bullet}{3} = \frac{5C^\bullet}{3} \quad (9)
\]
Case 3: $|S'| = 2$

Similar to case 2 to check that schedule $\pi_2$ or $\pi_3$ 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 $\pi_4$ obtained in Step (5) is no more than $C^*/3$ for this case.

For schedule $\pi_4$, if no critical job exists, see figure 9, then this is obviously

\[ C_{\text{max}}(\pi_4) = \sum_{i=1}^{n} (s_{\pi_4(i)}^2 + b_{\pi_4(i)}) = C^*. \] (10)

Otherwise, for the critical job $J_{\pi_4(u)}$, if $u > 2$, See figure 11, we have from figure 10,

\[ C_{\text{max}}(\pi_4) = \sum_{i=1}^{u} (s_{\pi_4(i)}^1 + a_{\pi_4(i)}) + \Delta_1 + \sum_{i=u+1}^{n} (s_{\pi_4(i)}^2 + b_{\pi_4(i)}) + b_{\pi_4(u)} \leq C^* + C^*/3 = 4C^*/3 \] (11)
If \( u = 2 \), then we obtain a contradiction.

\[
\sum_{i=1}^{n}(s_i^1 + a_i) < C^\star - \Delta_1 < C^\star - 2C^\star/3 = C^\star/3
\]

\[
C^\star/3 > \sum_{i=1}^{n}(s_i^1 + a_i) > (s_p^1 + a_p) + (s_q^1 + a_q) \geq \min\{s_p^2 + b_p, s_q^2 + b_q\} > C^\star/3.
\]

So obviously \( u \) must be equal to 1. Thus, see figure 12, we have

\[
\text{Figure 12: Illustrations of equation of } \pi_4 \text{ [12] and } \pi_5 \text{ [13].}
\]

\[
C_{\text{max}}(\pi_4) = (s_p^1 + a_p) + b_{\pi_4(p)} + \sum_{i=2}^{n}(s_{\pi_4(i)}^2 + b_{\pi_4(i)})
\]

\[
\leq C^\star/3 + C^\star = 4C^\star/3 \quad (12)
\]

Similarly for \( \pi_5 \), see figure 11, we need to change \( p \) to \( q \).

\[
C_{\text{max}}(\pi_5) = (s_q^1 + a_q) + b_{\pi_5(1)} + \sum_{i=2}^{n}(s_{\pi_5(i)}^2 + b_{\pi_5(i)})
\]

\[
\leq C^\star/3 + C^\star = 4C^\star/3 \quad (13)
\]

From the proof of theorem 1, we see that Steps (1)...(5) of Heuristic H1 can produce a solution with an error bound of no more than \( 2C^\star/3 \), and schedule \( \pi_4 \) in step (5), \( \pi_5 \) in step (6) can produce a solution with an error bound of no more than \( C^\star/3 \) 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 \( s_1^1 = h, a_1 = h, s_1^2 = 3h, b_1 = 7, s_2^1 = 3, a_2 = 4, s_2^2 = 6, b_2 = 3h, s_3^1 = m, a3 = m, s_3^2 = 1, b_3 = 1, s_1 = 8, \) and \( t_1 = 4h + 8 \), where \( h \gg 1 \) and
0 < m < $7/(3h + 6)$. Applying heuristic H1, we obtain $\pi_1 = \pi_3 = [J_1, J_3, J_2]$ with $C_{\text{max}}(\pi_1) = C_{\text{max}}(\pi_3) = 9h + 15$ (see figure 13(a)), and $\pi_2 = [J_3, J_2, J_1]$ with $C_{\text{max}}(\pi_2) = 10h + 2m + 14$ (see figure 13(b)). Since $(s^1_{p} + a_{p}) + (s^1_{q} + a_{q}) = 2h + 7 > s_1$, we need not consider Step (5) of H1. Thus, $C_{H1} = 9h + 15$. It is easy to check that $\pi^* = [J_2, J_1, J_3]$ with $C^* = 6h + 16$ (see figure 13(c)). Hence, we see that $(C_{H1} - C^*)/C^*$ approaches 1/2 as h approaches infinity.

Figure 13: (a)Solution of order $\pi_1$, $\pi_3$; (b)Solution of order $\pi_2$; (c)Solution of order $\pi^*$. 
In this section we present a heuristic for the problem $F2/setup, r - a(M_2)/C_{\text{max}}$ and evaluate its worst-case error bound by Wang and Cheng [12].

### 4.1 YHA algorithm ($\pi_1$)

Equivalent to heuristic $H1$’s $\pi_1$. For example let $s_1^1=2$, $a_1 = 3$, $s_2^1=4$, $a_2 = 2$, $s_3^1=8,a_3 = 3$, $s_1^2=4$, $b_1 = 2$, $s_2^2=3$, $b_2 = 5$, $s_3^2=6$, $b_3 = 4$, $s_2 = 15$, and $t_2 = 20$.

<table>
<thead>
<tr>
<th>Job number</th>
<th>Set A</th>
<th>Set B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$s_1^1 + a_1 - s_2^2 = 2 + 3 - 4 = 1 &lt; 2$</td>
<td>None</td>
</tr>
<tr>
<td>2</td>
<td>$s_1^2 + a_2 - s_2^2 = 4 + 2 - 3 = 3 &lt; 5$</td>
<td>None</td>
</tr>
<tr>
<td>3</td>
<td>None</td>
<td>$s_1^2 + a_3 - s_2^2 = 8 + 3 - 6 = 5 &gt; 4$</td>
</tr>
</tbody>
</table>

Table 4: Values considerer in $\pi_1$

Then, $J_1, J_2 \in A$, and $J_3 \in B$, because $s_1^2 + a_2 - s_2^2 > s_1^1 + a_1 - s_1^1$, then the order in set A will be $\{J_1, J_2\}$ (nondecreasing order) followed by the job in set B. Finally the order will be $\{J_1, J_2, J_3\}$, this is $\pi_1$. See Figure 14(a).

### 4.2 Decreasing ratio ($\pi_2$)

Similar to heuristic $H1$’s $\pi_2$. Sequence the jobs in nonincreasing order of $(s_1^2 + b_i)/(s_1^1 + a_i)$.

<table>
<thead>
<tr>
<th>Job number</th>
<th>$(s_1^2 + b_i)/(s_1^1 + a_i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$(s_1^2 + b_1)/(s_1^1 + a_1) = 6/5$</td>
</tr>
<tr>
<td>2</td>
<td>$(s_2^2 + b_2)/(s_2^1 + a_2) = 8/6$</td>
</tr>
<tr>
<td>3</td>
<td>$(s_3^2 + b_3)/(s_3^1 + a_3) = 10/11$</td>
</tr>
</tbody>
</table>

Table 5: Values considerer in $\pi_2$

Then we get $(s_2^2 + b_2)/(s_1^2 + a_2) > (s_1^2 + b_1)/(s_1^1 + a_1) > (s_3^2 + b_3)/(s_3^1 + a_3)$. Finally the order will be $\{J_2, J_1, J_3\}$, this is $\pi_2$. See Figure 14(b).
4.3 Largest job $q$ on machine 1 ($\pi_3$)

Next we need find job $J_q$ such that

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

<table>
<thead>
<tr>
<th>Job number</th>
<th>$s_i^1 + a_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$s_1^1 + a_1 = 2 + 3 = 5$</td>
</tr>
<tr>
<td>2</td>
<td>$s_2^1 + a_2 = 4 + 2 = 6$</td>
</tr>
<tr>
<td>3</td>
<td>$s_3^1 + a_3 = 8 + 3 = 11$</td>
</tr>
</tbody>
</table>

Table 6: Values considerer in $\pi_3$

Then we find $q = 3$, and we know the order of $\pi_1$, we just need to move job $J_p$ in the last position and keep the other $n - 1$ jobs in the same positions as those in $\pi_1$, then the order will be $\{ J_3, J_1, J_2 \}$, this is $\pi_3$. See Figure 14(c).

4.4 Largest job $p$ on machine 2 ($\pi_4$)

Next we need find job $J_p$ such that

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

<table>
<thead>
<tr>
<th>Job number</th>
<th>$s_i^2 + b_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$s_1^2 + b_1 = 1 + 4 + 2 = 6$</td>
</tr>
<tr>
<td>2</td>
<td>$s_2^2 + b_2 = 3 + 5 = 8$</td>
</tr>
<tr>
<td>3</td>
<td>$s_3^2 + b_3 = 6 + 4 = 10$</td>
</tr>
</tbody>
</table>

Table 7: Values considerer in $\pi_4$

Then we find $p = 3$, and we know the order of $\pi_2$, just need to move job $J_p$ in the first position and keep the other $n - 1$ jobs in the same positions as those in $\pi_2$, then the order will be $\{ J_3, J_2, J_1 \}$, this is $\pi_4$. See Figure 14(d).
Figure 14: (a) Solution of order $\pi_1$; (b) Solution of order $\pi_2$; (c) Solution of order $\pi_3$; (d) Solution of order $\pi_4$

4.5 Heuristic H2:

1. Find jobs $J_p$ and $J_q$ such that

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

and

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

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

18
(3) Sequence the jobs in nonincreasing order of \((s_i^2 + b_i)/(s_i^2 + a_i)\). Let the corresponding schedule be \(\pi_2\) and the corresponding makespan be \(C_{\text{max}}(\pi_2)\).

(4) Sequence job \(J_q\) in the last position and sequence the remaining \(n - 1\) jobs by YHA. Let the corresponding schedule be \(\pi_3\) and the corresponding makespan be \(C_{\text{max}}(\pi_3)\).

(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 \(\pi_4\) and the corresponding makespan be \(C_{\text{max}}(\pi_4)\).

(6) Select the schedule with the minimum makespan from the above four schedules. Let \(C_{H2} = \min \{C_{\text{max}}(\pi_1), C_{\text{max}}(\pi_2), C_{\text{max}}(\pi_3), C_{\text{max}}(\pi_4)\}\).

For the problem \(F_2/\text{setup}, r - a(M_2)/C_{\text{max}}\), 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 \(\pi\), define again the critical job \(J_{\pi(k)}\) as the last job in \(\pi\) such that its starting time on \(M_2\) is equal to its finishing time on \(M_1\) or the job in \(\pi\) before which the last idle time on \(M_2\) occurs.

Lemma 2 For schedule \(\pi_2\) 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 \(\pi\) be a given schedule.

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

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

Proof. (i) Similar to the proof of Lemma 1.
(ii) Let $I_{\pi_2}$ be the total idle time on $M_2$ in $\pi_2$. Under the assumption that $J_{\pi_2(k)}$ finishes just before time $s_2$ on $M_1$ in $\pi_2$, we have $I_{\pi_2} \leq s_2 - \sum_{j=1}^{h} (s_{\pi_2(j)}^2 + b_{\pi_2(j)})$. See figure 15.

Let $I_{\pi}$ be the total idle time on $M_2$ in $\pi$. So clearly,

\[
I_{\pi} \geq s_2 - \sum_{j=1}^{u} (s_{\pi(j)}^2 + b_{\pi(j)}) \\
\geq s_2 - \sum_{j=1}^{u} (s_{\pi(j)}^2 + b_{\pi(j)}) - (s_{\pi_2(h+1)} - a_{\pi_2(h+1)}) - (s_{\pi(u+1)} - a_{\pi(u+1)})
\]

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

This completes the proof. \(\square\)

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

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

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

Let \(S' = \{J_i|s_i^1 + a_i > C^*/3, i = 1, 2, ..., n\}\) and \(S'' = \{J_i|s_i^2 + b_i > C^*/3, i = 1, 2, ..., n\}.\) We can easily show that \(|S'| \leq 2\) and \(|S''| \leq 2\) from the lower bound \(\max \{\sum_{i=1}^{n}(s_i^1 + a_i) + \sum_{i=1}^{n}(s_i^2 + b_i)\} \leq C^*.\) When \(|S'| = 0\) and \(|S''| = 0,\) from (i) and (ii) of Lemma 2, we have \(C_{\text{max}}(\pi_2) \leq 5C^*/3.\) 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 \(\pi_3.\) If no critical job exists in \(\pi_3,\) this means no idle time on \(M_2,\) then \(C_{\text{max}}(\pi_3) = \sum_{i=1}^{n}(s_{\pi_3(i)} + b_{\pi_3(i)}) + \Delta_2 = C^*.\) Next, we assume that there exists a critical job in \(\pi_3.\) Let \(J_q\) be the critical job, see figure 16. then

\[ C_{\text{max}}(\pi_3) \leq \max \{\sum_{i=1}^{n}(s_{\pi_3(i)} + a_{\pi_3(i)}), t_2\} + b_q \leq C^* + C^*/3 = 4C^*/3.\]
Otherwise, let $J_{\pi_3(k)}$ ($k < n$) is the critical job, see figure 17. then we have

$$C_{\text{max}}(\pi_3) \leq \sum_{i=1}^{k} (s_{\pi_3(i)} + a_{\pi_3(i)}) + (\Delta_2 + b_{\pi_3(k)}) + \sum_{i=k+1}^{n} (s_{\pi_3(i)}^2 + b_{\pi_3(i)})$$

$$\leq C^\star + 2C^\star / 3 = 5C^\star / 3.$$ 

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

We check schedule $\pi_4$ obtained in Step (5) of Heuristic H2. If no critical job exists in $\pi_4$, then $C_{\text{max}}(\pi_4) = \sum_{i=1}^{n} (s_{\pi_4(i)}^2 + b_{\pi_4(i)}) + \Delta_2 = C^\star$. In the following, we assume that there exists a critical job in $\pi_4$. Since $|S''| \geq 1$, we assume that $s_p^2 + b_p > C^\star / 3$ for $J_p$, see figure 18. If

$$\sum_{i=1}^{n} (s_i^1 + a_i) \geq \max\{\max\{s_p^1 + a_p, s_p^2\} - \max\{s_p^1 + a_p - s_2, 0\}, s_p^2\} + b_p + \alpha \Delta_2,$$
where \( \alpha = 1 \) if \( s_2 < \max\{s^1_p + a_p, s^2_p\} + b_p \); otherwise, \( \alpha = 0 \). See figure 18. Then, we have

\[
C_{\max}(\pi_4) \leq \sum_{i=n}^n (s^1_i + a_i) + (1 - \alpha) \Delta_2 + \sum_{J_i \in \mathcal{S} \setminus \{J_p\}} (s^1_i + a_i) \\
\leq C^\star + 2C^\star/3 = 5C^\star/3.
\]

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

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

The proof is complete. \( \Box \)
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 $\frac{1}{3}$. Consider an instance with $s_1 = h$, $a_1 = 2$, $s_2 = 2$, $b_1 = 5$, $s_3 = 1 + h/2$, $a_2 = 1 + h/2$, $s_4 = 1$, $b_2 = h + 2$, $s_5 = h - 3$, $a_3 = 2$, $s_6 = h - 2$, $b_3 = 2$, $s_7 = h$, and $t_1 = 2$, where $h \gg 1$. Applying heuristic H2, we obtain $\pi_1 = [J_2, J_1, J_3]$ with $C_{\text{max}}(\pi_1) = 4h + 9$ (see figure 19(a)), and $\pi_2 = \pi_4 = [J_2, J_3, J_1]$ with $C_{\text{max}}(\pi_2) = C_{\text{max}}(\pi_4) = 4h + 9$ (see figure 19(b)), and $\pi_3 = [J_1, J_3, J_2]$ with $C_{\text{max}}(\pi_3) = 4h + 8$ (see figure 19(c)). Thus $C_{H2} = 4h + 8$. It is easy to check that $\pi^\star = [J_3, J_2, J_1]$ with $C^\star = 3h + 11$ (see figure 19(d)). Hence, we see that $(C_{H2} - C^\star)/C^\star$ approaches $1/3$ as $h$ approaches infinity.

Figure 19: (a)Solution of order $\pi_1$; (b)Solution of order $\pi_2$, $\pi_4$; (c)Solution of order $\pi_3$; (d)Solution of order $\pi^\star$. 

24
The heuristic was implemented in Java, on a Pentium-4 PC clocked at 2.7 GHz under the operating system Windows XP. We ran randomly generated job numbers with \( n = 6, 7, 8, 9, 10, 11, 12 \). For each job set we ran 2 different unavailable times on the two machines.

The following table is based on machine 1. The first column is number of jobs, the second column is the number of simulations, Column 3 is the percentage of the simulations that the heuristic yields the optional solution, column 4 is the average error bound and column 5 is the largest error bound.

Based on this program, all jobs’ setup times and processing times are taken to be random integer numbers between 1 and 10. The unavailable time is done by choosing a random number, \( l_1 \) between the values 0.1 and 0.15, and another random number \( k_1 \) between the values 0.2 and 0.25. Then \( s_1 = \lfloor l_1 \cdot \sum_{i=n}^{n} (s_1^i + a_i) \rfloor \); \( t_1 = \lfloor k_1 \cdot \sum_{i=n}^{n} (s_1^i + a_i) \rfloor \).
<table>
<thead>
<tr>
<th>Job size n</th>
<th>Numbers of simulation</th>
<th>Optimal solution percentage</th>
<th>Average Error bound</th>
<th>Largest Error bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>100</td>
<td>77%</td>
<td>0.02512</td>
<td>0.04521</td>
</tr>
<tr>
<td>7</td>
<td>100</td>
<td>88%</td>
<td>0.01231</td>
<td>0.05376</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>85%</td>
<td>0.02891</td>
<td>0.05427</td>
</tr>
<tr>
<td>9</td>
<td>100</td>
<td>84%</td>
<td>0.01929</td>
<td>0.04381</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>78%</td>
<td>0.03119</td>
<td>0.04841</td>
</tr>
<tr>
<td>11</td>
<td>100</td>
<td>84%</td>
<td>0.02867</td>
<td>0.04639</td>
</tr>
<tr>
<td>12</td>
<td>100</td>
<td>75%</td>
<td>0.03243</td>
<td>0.05082</td>
</tr>
</tbody>
</table>

Table 8: Computational results for heuristic 1

The following table is based on machine 2. The first column is number of jobs, the second column is the number of simulations, Column 3 is the percentage of the simulations that the heuristic yields the optional solution, column 4 is the average error bound and column 5 is the largest error bound.

Based on this program, all jobs' setup times and processing times are taken to be random integer numbers between 1 and 10. The unavailable time is done by choosing a random number $l_2$ between the values 0.1 and 0.15, and another random number, $k_2$ between the values 0.2 and 0.25. Then $s_2 = \lfloor l_2 \cdot \sum_{i=n}^{n} (s_i^2 + b_i) \rfloor$; $t_2 = \lfloor k_2 \cdot \sum_{i=n}^{n} (s_i^2 + b_i) \rfloor$. 

26
<table>
<thead>
<tr>
<th>Job numbers size $n$</th>
<th>Numbers of simulation</th>
<th>Optimal solution percentage</th>
<th>Average Error bound</th>
<th>Largest Error bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>100</td>
<td>77%</td>
<td>0.03066</td>
<td>0.04189</td>
</tr>
<tr>
<td>7</td>
<td>100</td>
<td>88%</td>
<td>0.03482</td>
<td>0.04641</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>85%</td>
<td>0.04482</td>
<td>0.07901</td>
</tr>
<tr>
<td>9</td>
<td>100</td>
<td>84%</td>
<td>0.05517</td>
<td>0.06562</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>78%</td>
<td>0.02629</td>
<td>0.04671</td>
</tr>
<tr>
<td>11</td>
<td>100</td>
<td>84%</td>
<td>0.04227</td>
<td>0.08943</td>
</tr>
<tr>
<td>12</td>
<td>100</td>
<td>75%</td>
<td>0.05012</td>
<td>0.07514</td>
</tr>
</tbody>
</table>

Table 9: Computational results for heuristic 2

6 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 presented two polynomial-time heuristics developed by Wang and Cheng and analyzed their error bounds by simulation. From the computational results, we can see that heuristic 1 is more accurate.


