HEURISTICS FOR TWO-MACHINE FLOWSHOP SCHEDULING WITH SETUP TIMES AND AN AVAILABILITY CONSTRAINT

Wei Cheng
Health Monitor Network, Parmus, NJ

John Karlof
Department of Mathematics and Statistics
University of North Carolina Wilmington
karlof@uncw.edu

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 a heuristic algorithm developed by Wang and Cheng to minimize makespan and use simulation to estimate its actual error bound. Wang and Cheng showed the worst-case error bounds are no larger than $\frac{2}{3}$ but, did not consider the average error bound.

1 INTRODUCTION

The problem of minimizing the makespan (the total completion time) of machine scheduling problems with availability constraints,(i.e. where one or more machines are unavailable for specified lengths of time, such as for routine maintenance), has attracted much research attention over the years. The two machine flowshop scheduling problem with availability contraints was first studied by Lee [3] in 1997. Under the job resumable assumption, he proved that the problem is NP hard even when an unavailability constraint is imposed on only one machine. Lee developed two heuristics to solve the problem. The first heuristic solved the problem when the unavailability constraint is imposed on the first machine and has a worst case error bound of $\frac{1}{2}$ while the second heuristic solved the problem when the unavailability constraint is imposed on the second machine and has a worst case error bound of $\frac{1}{3}$.

Definition 1 *Suppose C_H is the makespan of a machine scheduling problem obtained from heuristic H and $C^\star$ is the optimal makespan. Then the error bound for heuristic H is $\frac{C_H - C^\star}{C^\star}$.*

Also considering the resumable case, Chang and Wang [5] developed an improved heuristic with a worst case error bound of $\frac{1}{3}$ when the unavailability constraint is placed only on the first machine. Breit [1] presented an improved heuristic with a worst case error bound of $\frac{1}{4}$ for the problem with an availability constraint only on the second machine. Chang and Wang [2] considered a special case of the problem when availability constraints are placed on both machines consecutively. The heuristic they developed had a worst case error bound of $\frac{2}{3}$.

In all the above mentioned flowshop scheduling models, setup times are not considered; that is, setup times are assumed to be included in the processing times. However, in many industrial settings, it is necessary to treat setup times as separated from processing times (for example [4, 6]). The two machine flowshop scheduling problem with anticipatory setup times when an availability contraint is imposed on one machine was studied by Wang and Chang [7]. They present two heuristics with worst case error bounds no larger than $\frac{2}{3}$ for solving the problem when the availability constraint is imposed on machines 1 and 2 respectively.

The purpose of this paper is to estimate by simulation the actual error bounds of the algorithms presented in [7]. In section 2, we introduce the notation and present the parallel machine scheduling problem with the unavailable time on machine 1 and in section 3, we present the algorithm from [7] for this case and fill in the details of the proofs in [7]. The algorithm and proof for the case when the availability constraint is imposed on machine 2 is similar, thus those are omitted. In section 5, we program both algorithms in JAVA and present estimates of the actual error bounds using simulation.

2 PROBLEM STATEMENT AND NOTATION

Problem Statement: Given a two machine flowshop scheduling problem with job set up times, the resumable assumption (a job or set up may be stopped and then resummed from the stopping point), and a fixed interval of unavailability time on one of the machines, find the permutation of the jobs that minimizes the makespan.

The following notation will 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_i^1, s_i^2 :setup times of J_i on M_1 and M_2 , respectively, where $s_i > 0, s_i > 0$;
- a_i, b_i :processing times of J_i on M_1 and M_2 , respectively, where $a_i > 0, b_i > 0$;
- π: = $[J_{\pi(1)}$,...,$J_{\pi(n)}]$:a permutation schedule, where $J_{\pi(i)}$ is the ith job in π ;
- $\pi^\star$:an optimal schedule;
- C_{H_x}:the makespan yielded by heuristic H_x;

- $C^\star$:the optimal makespan.

Example 1 *Unavailability time on machine 1 with* $\Delta_1 = 5, s_1 = 10, t_1 = 15, n = 3$. *Let* $s_1^1 = 3, a_1 = 4, s_2^1 = 5, a_2 = 4, s_3^1 = 4, a_3 = 5,\ s_1^2 = 2, b_1 = 6, s_2^2 = 4, b_2 = 8, s_3^2 = 2, b_3 = 3$, *A schedule* $\pi = [J_1, J_2, J_3]$ *is shown in Fig. 1.*

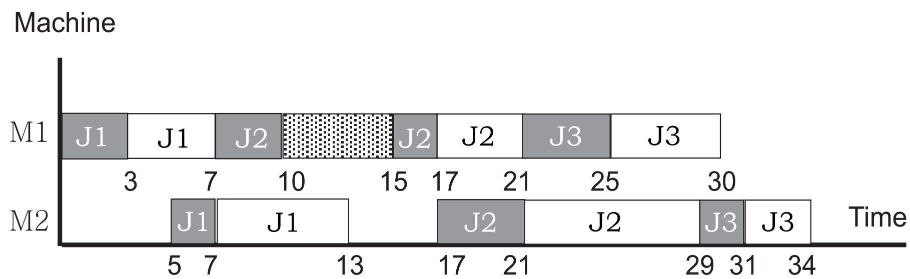

Figure 1: A schedule for example 1 with the makespan = 34

3 ALGORITHM FOR THE UNAVAILABLE INTERVAL ON M_1

In this section we develop a heuristic by Wang and Cheng [7] and evaluate its worst-case error bound. The basic idea of the heuristic is 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.

3.1 YHA algorithm (π_1)

The Yoshida and Hitomi algorithm (YHA) [8] optimally solves the flowshop scheduling problem with setup times. It 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, $b_3 = 3$, $s_1 = 20$, and $t_1 = 25$.

Job number	Set A	Set B
1	None	$s_1^1 + a_1 - s_1^2$=9+3-7=5>4
2	$s_2^1 + a_2 - s_2^2$=2+4-1=5<7	None
3	$s_3^1 + a_3 - s_3^2$=3+2-2=3<3	None

Table 1: Values considered in π_1

Thus $\pi_1 = \{J_3, J_2, J_1\}$. See Figure 2(a).

3.2 Decreasing ratio (π_2)

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

Job number	$(s_i^2 + b_i)/(s_i^1 + a_i)$
1	$(s_1^2 + b_1)/(s_1^1 + a_1) = 11/12$
2	$(s_2^2 + b_2)/(s_2^1 + a_2) = 8/6$
3	$(s_3^2 + b_3)/(s_3^1 + a_3) = 5/5$

Table 2: Values considered in π_2

Then $\pi_2 = \{J_2, J_3, J_1\}$. See Figure 2(b).

3.3 Largest job p, q on machine 2 (π_3)

Determine jobs J_p and J_q such that

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

Job number	$s_i^2 + b_i$
1	$s_1^2 + b_1$=7+4=11
2	$s_2^2 + b_2$=1+7=8
3	$s_3^2 + b_3$=2+3=5

Table 3: Values considered in π_3

For π_3 put job J_p first and keep the other $n-1$ jobs in the same order as π_2. Then $\pi_3 = \{J_1, J_2, J_3\}$. See Figure 2(c).

3.4 Random sequences π_4 and π_5

Test if $(s_p^1 + a_p) + (s_q^1 + a_q) \leq s_1$ if not then no π_4, π_5. Otherwise make two sequences
π_4: Choose 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).
π_5: Choose 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 H1:

(1) Find jobs J_p and J_q such that

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

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

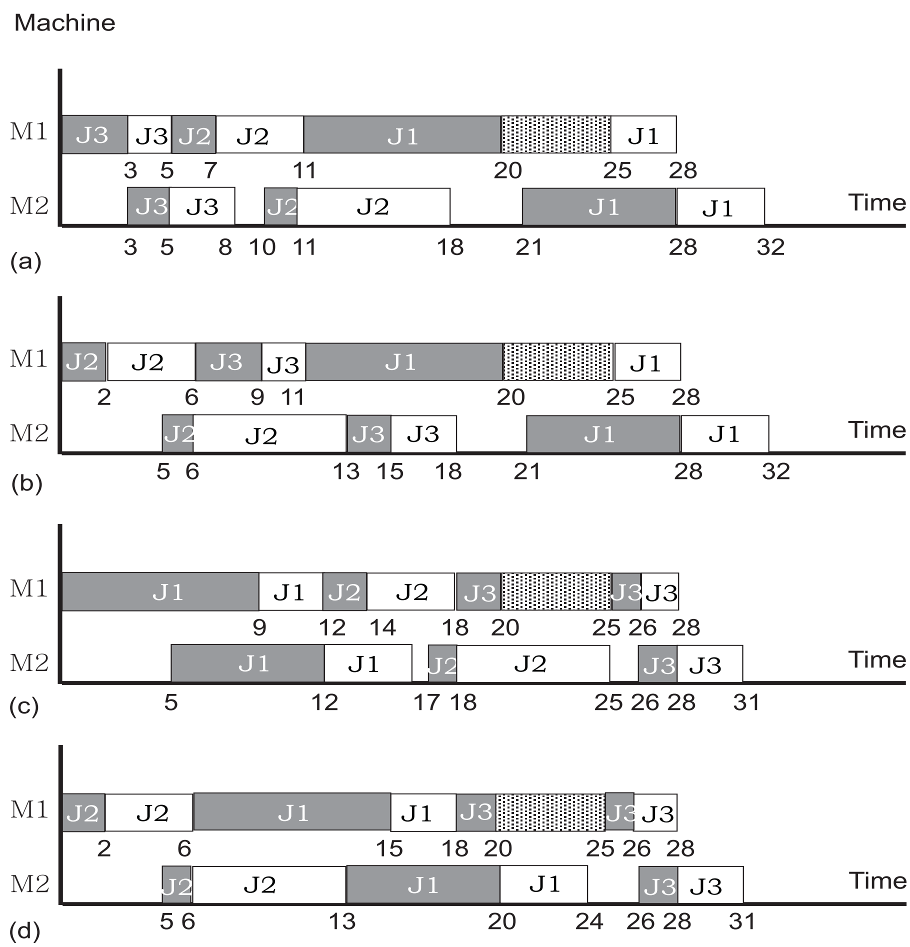

Figure 2: (a) π_1; (b) π_2; (c) π_3 and π_4; (d) π_5

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

(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). The schedule is π_3 and the corresponding makespan is $C_{\max}(\pi_3)$.

(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. The schedule is π_4 and the corresponding makespan is $C_{\max}(\pi_4)$.

(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. The schedule is π_5 and the corresponding makespan is $C_{\max}(\pi_5)$.

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

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

Definition 2 *Let π be any schedule. 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 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:*

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

Proof. There is no idle time on machine 2 after the critical job, so if there is no critical job then $C_{\max}(\pi_2) = \sum_{i=1}^{n}(s^2_{\pi_2(i)} + b_{\pi_2(i)}) = C^\star$. So, we will always assume there is a critical job for each of the schedules π_i and we have for π_2,

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

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 π. We have

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

and because $\sum_{j=1}^{n}(s^1_{\pi(j)} + a_{\pi(j)}) = \sum_{j=1}^{n}(s^1_{\pi_2(j)} + a_{\pi_2(j)})$,

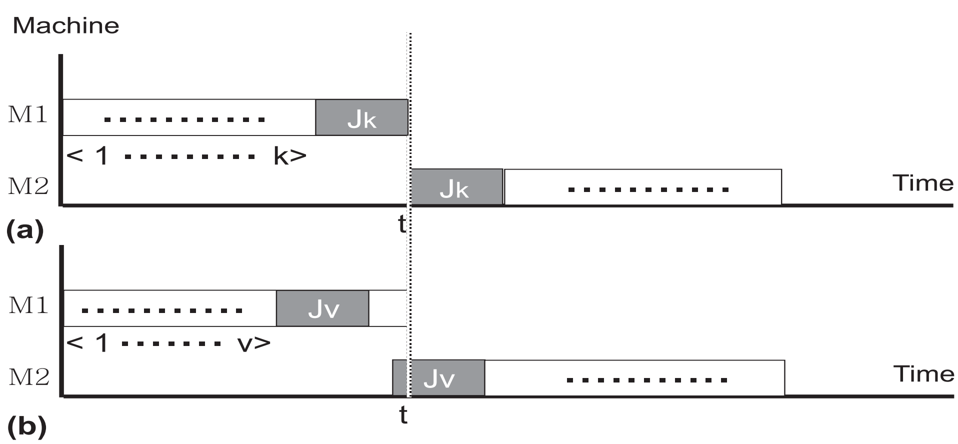

Figure 3: illustrations of (2), (a)Order π_2; (b)Order π;

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

Since all the jobs are sequenced in nonincreasing order of $(s^2_{\pi_2(j)} + b_{\pi_2(j)})/(s^1_{\pi_2(j)} + b_{\pi_2(j)})$ in π_2, and because after critical job k on M_1, there is no idle time, we have

$$\sum_{j=k+1}^{n} (s^2_{\pi_2(j)} + b_{\pi_2(j)}) > \sum_{j=k+1}^{n} (s^1_{\pi_2(j)} + a_{\pi_2(j)}). \tag{3}$$

From (2) and (3),

$$\sum_{j=v+1}^{n} (s^2_{\pi(j)} + b_{\pi(j)}) \geq \sum_{j=k+1}^{n} (s^2_{\pi_2(j)} + b_{\pi_2(j)}). \tag{4}$$

For schedule π, we have

$$C_{\max}(\pi) \geq t + \sum_{j=v+1}^{n} (s^2_{\pi_2(j)} + b_{\pi_2(j)}) - s^2_{\pi(v+1)}. \tag{5}$$

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

$$\begin{aligned} C_{\max}(\pi_2) &= t + b_{\pi_2(k)} + \sum_{j=k+1}^{n} (s^2_{\pi_2(j)} + b_{\pi_2(j)}) \\ &\leq t + b_{\pi_2(k)} + \sum_{j=v+1}^{n} (s^1_{\pi(j)} + a_{\pi(j)}) \\ &\leq C_{\max}(\pi) + b_{\pi_2(k)} + s^2_{\pi(v+1)}. \end{aligned}$$

Theorem 1 $(C_{H1} - C^\star)/C^\star \leq 2/3$.

Proof. If $\sum_{i=1}^{n}(s^1_i + a_i) \leq s_1$, it is obvious that $C_{\max}(\pi_1) = C^\star$ from the Yoshida and Hitomi algorithm(YHA)[14]. So we assume $\sum_{i=1}^{n}(s^1_i + a_i) > s_1$.

Since all the jobs are resumable and π_1 is the best scedule without unavailable time, we have $C_{\max}(\pi_1) \leq C^\star + \Delta_1$. So if $\Delta_1 \leq 2C^\star/3$, then we are finished. So, in the following, we assume $\Delta_1 > 2C^\star/3$.

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

Case 1:$| S' | = 0$

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

Case 2:$| S' | = 1$ (Jobs k and v are still as defined in lemma 1.)

In this case, $S' = \{J_p\}$. If $s_p^2 \leq C^\star/3$ and $b_p \leq C^\star/3$, then $b_{\pi_2(k)} \leq C^\star/3$ and $s^2_{\pi^\star(v+1)} \leq C^\star/3$ and from lemma 1 $C_{\max}(\pi_2) \leq C^\star + C^\star/3 + C^\star/3 \leq 5C^\star/3$. Otherwise, if $s_p^2 > C^\star/3$ or $b_p > C^\star/3$,we consider schedule π_3 obtained in step (4) of heuristic H1 and let the critical job of π_3 be $J_{\pi_3(u)}$. First we suppose that $s_p^1 + a_p \leq s_1$. Now, if $\sum_{i=1}^{u}(s^1_{\pi_3(i)} + a_{\pi_3(i)}) \leq s_1$, see figure 4, then

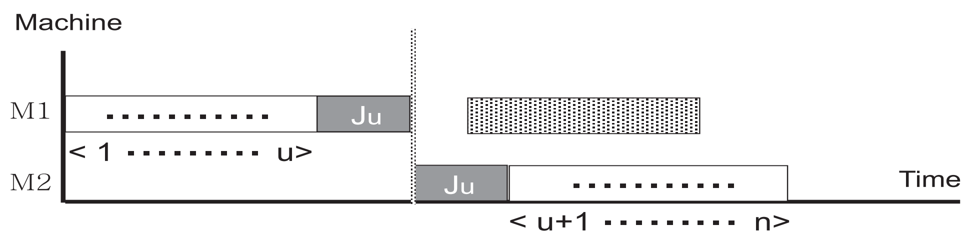

Figure 4: Illustrations of π_3; J_u on M_2 equal to $b_{\pi_3(u)}$.

$$\begin{aligned}
C_{\max}(\pi_3) &= \sum_{i=1}^{u}(s^1_{\pi_3(i)} + a_{\pi_3(i)}) + (\sum_{i=u+1}^{n}(s^2_{\pi_3(i)} + b_{\pi_3(i)}) + b_{\pi_3(u)}) \\
&\leq C^\star/3 + C^\star \\
&= 4C^\star/3
\end{aligned}$$

otherwise, let $\sum_{i=1}^{u}(s^1_{\pi_3(i)} + a_{\pi_3(i)}) > s_1$, J_p is the first job in π_3 and $s_p^1 + a_p \leq s_1$, then $u > 1$, see figure 5. Thus, we have

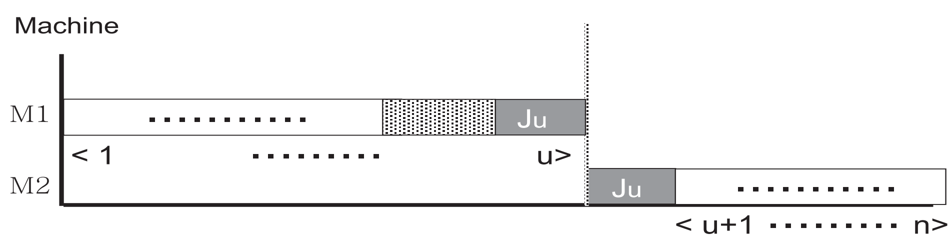

Figure 5: Illustration of equation with π_3; J_u on M_2 equal to $b_{\pi_3(u)}$.

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

For subcase $s_p^1 + a_p > s_1$, we have $s_p^1 + a_p + \Delta_1 + b_p \leq C^\star$. If the critical job does not exist or job J_p is the critical job, then we have, see figure 6:

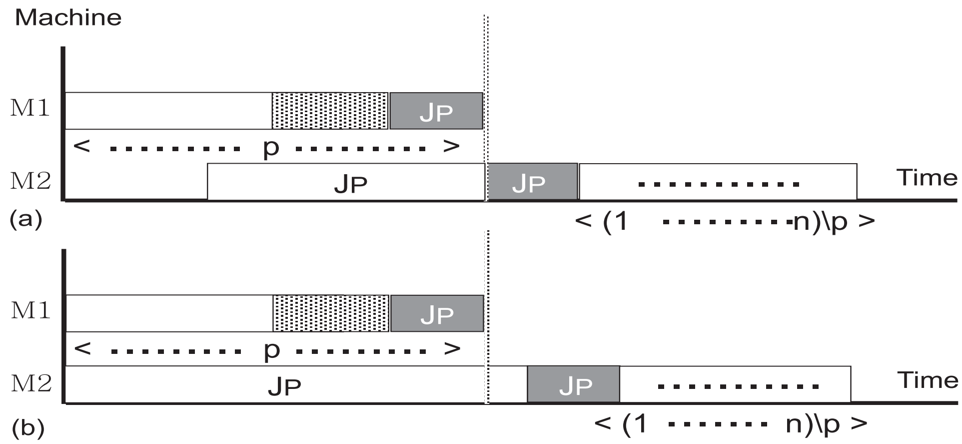

Figure 6: Compare $\max\{s_p^1 + a_p + \Delta_1, s_p^2\}$. $s_p^1 + a_p + \Delta_1$ in (a), s_p^2 in (b).

$$\begin{aligned} C_{\max}(\pi_3) &= \max\{s_p^1 + a_p + \Delta_1, s_p^2\} + b_p + \sum_{J_i \in S \setminus J_p} (s^2_{\pi_3(i)} + b_{\pi_3(i)}) \\ &\leq C^\bigstar + 2C^\bigstar/3 \\ &= 5C^\bigstar/3. \end{aligned}$$

Otherwise, for the critical job $J_{\pi_3(u)}$, $u > 1$, see figure 7, we have

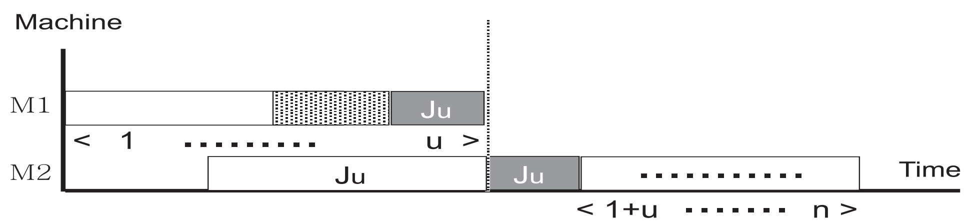

Figure 7: Illustration of π_3; J_u on machine 2 equal to $b_{\pi_3(u)}$.

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

Case 3:$\mid S' \mid = 2$

In this case, we show that the error bound of schedule π_4 obtained in step (5) is no more than $C^\bigstar/3$.

Suppose $J_{\pi_4(u)}$ is the critical job for π_4, then if $u > 2$, see figure 8, we have and because

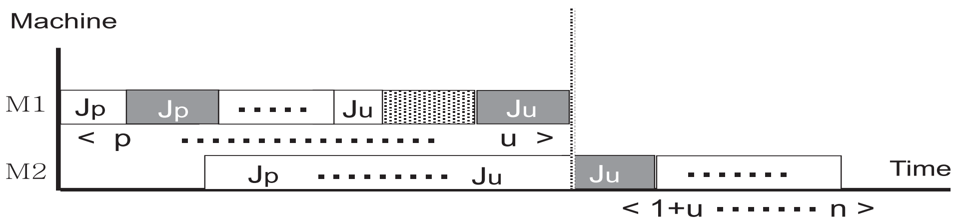

Figure 8: Illustration of π_4.

$| S' |= 2$ and $u > 2$ that means $\sum_{i=1}^{u}(s^1_{\pi_4(i)} + a_{\pi_4(i)}) + \Delta_1 < C^\star$ and

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

If $u = 2$, then we have the following contradiction: $\sum_{i=1}^{n}(s^1_i + a_i) \leq C^\star - \Delta_1 < C^\star - 2C^\star/3 = C^\star/3$ But $C^\star/3 > \sum_{i=1}^{n}(s^1_i + a_i) > (s^1_p + a_p) + (s^1_q + a_q) \geq min\{s^2_p + b_p, s^2_q + b_q\} > C^\star/3$.

If $u = 1$, then see figure 9, we have

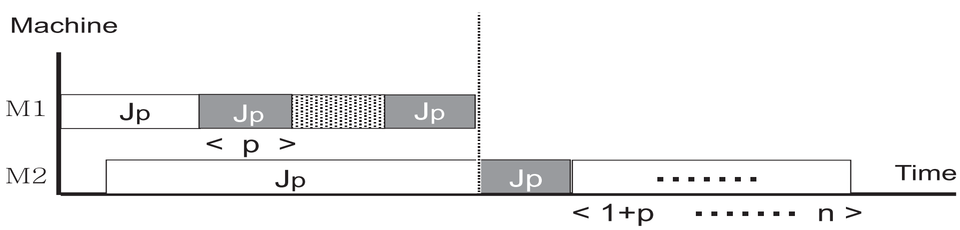

Figure 9: Illustration of π_4 and π_5.

$$\begin{aligned} C_{\max}(\pi_4) &= (s^1_p + a_p) + (b_p + \sum_{i=2}^{n}(s^2_{\pi_4(i)} + b_{\pi_4(i)})) \\ &\leq C^\star/3 + C^\star = 4C^\star/3 \end{aligned}$$

Similarly for π_5. This completes the proof of theorem 1.

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 π_4 in step (5), π_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, [7] contains a (not very realistic) example to show that the worst case error bound is no smaller than $C^\star/2$.

Number of jobs, n	Optimal solution percentage of H1	Average error bound	Largest error bound
6	77%	.025	.045
7	88%	.012	.054
8	85%	.029	.054
9	84%	.019	.044
10	78%	.031	.048
11	84%	.029	.046
12	75%	.032	.050

Table 4: Computational results for heuristic 1

4 COMPUTATIONAL RESULTS

In this paper, we provided the details only for heuristic H1 that applies to the case where the unavailable time is on machine 1. The paper [7] also contains heuristic H2 that applies to the case where the unavailable time is on machine 2. We present our computational results for both H1 and H2 in tables 4 and 5 respectively. Both heuristics have a worst case error bound no larger than $2C^\star/3$. In order to determine how tight this error bound is, we implemented both heuristics using the programming language JAVA. We simulated randomly generated job shops with $n = 6, 7, \ldots, 12$. All jobs' setup times and processing times were random integers between 1 and 10. The unavailable time was determined by choosing a random number, l between .1 and .15 and another random number between, k between .2 and .25. Then, the unavailable time for machine 1 was the interval:

$$\left[\lfloor l \cdot \sum_{i=1}^{n}(s_i^1 + a_i)\rfloor, \lfloor k \cdot \sum_{i=1}^{n}(s_i^1 + a_i)\rfloor\right] \text{ where } \lfloor\ \ \rfloor \text{ is the floor function.}$$

A similar interval was used for machine 2. We experimented with various sized intervals for the unavailable times and the intervals choosen seemed the most reasonable. We calcululated 100 simulations for each value of n. The value of $C^\star$ was determined by considering the makespan for all permutations of the jobs and choosing the best one.

5 CONCLUSIONS

In this paper we studied the two-machine flowshop scheduling problem with anticipatory setup times, resumable setup times and processing times, and an availability constraint imposed on one of the machines. Wang and Chang [7] presented two heuristics for this problem when the availability constraint was imposed on machines 1 and 2 respectively. We presented heuristic H1 including detailed proofs whose details are not found in [7] to show that the worst case error bound of the heuristic was $\frac{2}{3}$. We also presented a simulation study testing the heuristics and determined that:

Number of jobs, n	Optimal solution percentage of H2	Average error bound	Largest error bound
6	80%	.031	.042
7	89%	.012	.054
8	82%	.045	.079
9	85%	.055	.066
10	77%	.026	.047
11	84%	.042	.089
12	72%	.050	.075

Table 5: Computational results for heuristic 2

- heuristics 1 and 2 found the optimal solution an average of 82% and 81% percent of the time;
- the average average error bound for heuristic 1 was .025 while the average largest error bound was .049;
- the average average error bound for heuristic 2 was .037 while the average largest error bound was .065.

The main conclusion is that these heuristics perform much better than their worst case error bound of .666 suggests.

References

[1] Breit J., An improved approximation algorithm for two-machine flow shop scheduling with an availability constraint, Information Processing Letters 2004;90:273–8.

[2] Cheng TCE., Wang G., Two-machine flowshop scheduling with consecutive availability constraints. Information Processing Letters 1999;71:49–54.

[3] Lee C-Y. Minimizing the makespan in the two-machine flowshop scheduling problem with an availability constraint. Operations Research Letters 1997;20:129–39.

[4] Rajedran C., Ziegler H. Heuristics for scheduling in a flow shop with setup processing and removal times separated. Production Planning and Control 1997;8:568–76.

[5] Schmidt G. Scheduling with limited availability. European Journal of Operational Research 2000;121:1–15.

[6] Sule D., Huang K., Sequencing on two and three machines with setup, processing and removal times separated. International Journal of Production Research 1983;21:723–32.

[7] Wang, X., Chang, E., Heuristics for two machine flowshop scheduling with setup times and an unavailability constraint, Computers and Operations Research 2007; 34:152–162.

[8] Yoshida, T., Hitomi, K., Optimal two-stage production scheduling with setup times separated, AIIE Transactions 1979;11:261–263.