Optim Lett (2009) 3:583–592
DOI 10.1007/s11590-009-0137-6

ORIGINAL PAPER

A $(\frac{4}{3})-$approximation algorithm for a special case of the two machine flow shop problem with several availability constraints

Hatem Hadda

Received: 22 April 2009 / Accepted: 24 June 2009 / Published online: 9 July 2009

Abstract In this paper, we deal with a special case of the two-machine flow shop scheduling problem with several availability constraints on the second machine, under the resumable scenario. We develop an improved algorithm with a relative worst-case error bound of 4/3.

Keywords Flow shop scheduling · Availability constraint · Approximation algorithms · Error bound

1 Introduction

While most papers dealing with the scheduling field assume that the machines are available all the time, a growing number of works are considering that the machines are subject to breakdown. Authors differentiate between different models, the *semi-resumable* (*sr*) case that if a job cannot be finished before the next down period of a machine then the job will have to partially restart when the machine has become available again. This model contains two special cases: namely the $resumable(r)$ when the job can be continued without any penalty and the $nonresumable(nr)$ case when the job needs to totally restart.

In this paper, we tackle a special case of the two-machine flow shop scheduling problem with several availability constraints (holes for short) on the second machine. In our case, we consider that the last hole starts before the optimal completion time (C^{joh}) of all the jobs when considering the $F_2||C_{\max}$ problem. The objective is to find a schedule of n given jobs that minimizes the total completion time (i.e. the

H. Hadda (✉)
Unité de Recherche URAII, INSAT-TUNIS, Centre Urbain Nord,
B.P. No 676, Tunis Cedex, Tunisia
e-mail: hatem.hadda@esti.rnu.tn

makespan). Each job J_i is composed by two operations O_{iA} and O_{iB}, which have to be processed on two machines A and B. Each machine can process at most one job at a time. Machine A is assumed to be always available. While machine B is unavailable during q holes and the precise time of each hole is known in advance. Besides, all jobs are supposed to be resumable. Following Kubzin et al. [7], we will use the notation $F_2|h(q_A, q_B), T|C_{\max}$ to denote a two-machine flow shop problem with q_A (respectively q_B) holes on machine $A(B)$ under scenario $T \in \{r, sr, nr\}$. Extending this notation, we will denote our special case $F_2|h(0, q), r, s_q < C^{\text{joh}}|C_{\max}$, where s_q is the start time of the last hole on machine B.

While the well-known two-machine flow shop problem can be solved by Johnson's algorithm [5], Lee [8,9] showed that the problem with an availability constraint is NP-hard, and proposed a $(\frac{3}{2})$-approximation algorithm for the $F_2|h(0, 1), sr|C_{\max}$ problem; and a 2-approximation algorithm for the $F_2|h(1, 0), sr|C_{\max}$. Better approximation algorithms have been proposed for the resumable scenario. In this respect, Breit [1] gives a $(\frac{5}{4})$-approximation algorithm for the $F_2|h(0, 1), r|C_{\max}$ problem, while Cheng and Wang [3] propose a $(\frac{4}{3})$-approximation algorithm for the $F_2|h(1, 0), r|C_{\max}$. Finally, it has been shown that the two-machine flow shop with a single hole admits a PTAS under the semiresumable scenario [7] and a FPTAS under the resumable scenario [10].

Concerning the two-machine flow shop problem with a variable number of holes, it has been shown to be NP-hard in the strong sense [6]. Furthermore, the problem under the resumable scenario, has been proven to be non-approximable within a fixed factor in polynomial time provided that there is at least one hole on the second machine [6,9]. Under the semiresumable and nonresumable scenarios, Breit et al. [2] prove that the single machine problem with two holes is also non-approximable within a fixed factor in polynomial time. This means that the $F_2|h(q, 0), r|C_{\max}$ problem is the only configuration of the two-machine flow shop problem with several holes that may admit a fixed factor approximation algorithm, for which Kubzin et al. [7] propose a $(\frac{3}{2})$-approximation algorithm. In this context, Cheng and Wang [4] propose a $(\frac{5}{3})$-approximation algorithm for the particular configuration of the semiresumable two-machine flow shop problem with consecutive two holes (one on each machine).

The remainder of this paper is organized as follows. Section 2 introduces some notations and definitions. In Sect. 3, we state our approximation algorithm and analyze its worst-case behavior.

2 Notation and basic definitions

We will use the following notation.
$J = \{J_1, \ldots, J_n\}$: the set of jobs.
$\pi = \langle\pi(1), \ldots, \pi(n)\rangle$: a permutation schedule where $\pi(i)$ is the ith job in π.
a_i, b_i: processing times for J_i on A and B respectively.
q : number of holes. Without loss of generality we assume that no two holes overlap.
$s_k,\ t_k$: start and finish time of hole k, $k \in \{1, \ldots, q\}$. We assume that $s_1 < s_2 < \cdots < s_q$.
$h_k = t_k - s_k$: length of hole k.
$S_{ij}(\pi)$ and $C_{ij}(\pi)$: the start and finish time of operation $O_{ij},\ i \in \{1, \ldots, n\},\ j \in$

$\{A, B\}$ in a permutation π. We will drop the reference to the schedule π whenever no confusion can arise.
$C_{\max}(\pi)$: the makespan of π.
$\pi^\star$: an optimal solution.
$C^\star_{\max}$: the optimal makespan.
C^{joh}: the optimal makespan for $F_2||C_{\max}$. It is evident that

$$C^{\text{joh}} \le C^\star_{\max}. \tag{1}$$

We also define $a(Q) = \sum_{J_k \in Q} a_k$, $b(Q) = \sum_{J_k \in Q} b_k$ for a non-empty set Q of jobs.

Furthermore, for a given job $\pi(k)$ in π we define $H_{\pi(k)} = \sum_{h_i \in I} h_i$, where $I = \{h_i \mid s_i > S_{\pi(k)B}\}$, and $H_T = \sum_{1 \le i \le q} h_i$.

We note that it is sufficient to consider permutation schedules for the $F_2|h(0, q), r|C_{\max}$ [7]. Furthermore, we assume that all operations are started as early as possible.

The special case that we consider assumes that $s_q < C^{\text{joh}}$, and necessarily, hole q will affect all schedules. Hence

$$b(J) + H_T \le C^\star_{\max}. \tag{2}$$

We also note that the assumed restriction ($s_q < C^{\text{joh}}$), is reasonable, as in many situations, C^{joh} can be considered as an approximation to the scheduling horizon.

In order to calculate the makespan of a given schedule π, we have to search for the job $\pi(u)$ which starts the last busy period on machine B. One of the following two conditions must be realized (see Fig. 1).

Condition1 $C_{\pi(u)A} = S_{\pi(u)B}$. Hence

$$C_{\max}(\pi) = C_{\pi(u)A} + \sum_{i=u}^{n} b_{\pi(i)} + H_{\pi(u)}. \tag{3}$$

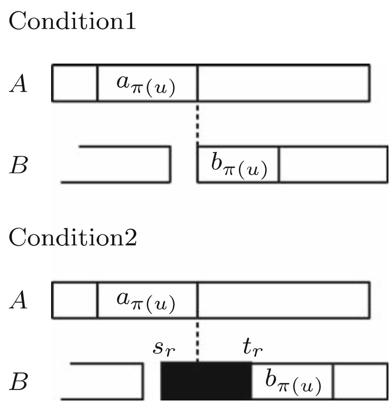


Fig. 1 Two shapes of schedule π

Condition2 There exists a hole r such that $s_r \leq C_{\pi(u)A} < t_r$. Hence

$$C_{\max}(\pi) = t_r + \sum_{i=u}^{n} b_{\pi(i)} + H_{\pi(u)}. \tag{4}$$

While the general $F_2|h(0,q), r|C_{\max}$ problem is not polynomially approximable within a fixed factor [9], the next result shows that it is not the case for our special case.

Theorem 1 *If π is a solution to $F_2|h(0,q), r, s_q < C^{\text{joh}}|C_{\max}$, then $C_{\max}(\pi)/C^{\star}_{\max} \leq 2$.*

Proof Let $\pi(u)$ be the job which starts the last busy period on machine B. If $\pi(u)$ follows (condition1), then using (2) and (3) we have $C_{\max}(\pi) = C_{\pi(u)A} + \sum_{i=u}^{n} b_{\pi(i)} + H_{\pi(u)} \leq a(J) + b(J) + H_T \leq 2C^{\star}_{\max}$.

Otherwise ($\pi(u)$ follows (condition2)), using (2) and (4), we obtain $C_{\max}(\pi) = t_r + \sum_{i=u}^{n} b_{\pi(i)} + H_{\pi(u)} \leq C^{\star}_{\max} + b(J) + H_T \leq 2C^{\star}_{\max}$. □

We now recall the following rule for the two-machine flow shop problem.

Ratio Rule (RR) : J_i precedes J_j if $b_i/a_i > b_j/a_j$. If $b_i/a_i = b_j/a_j$, we break the tie arbitrarily.

3 An improved heuristic

We introduce now a $(\frac{4}{3})-$approximation algorithm for the $F_2|h(0,q), r, s_q < C^{\text{joh}}|C_{\max}$ problem.

Algorithm H:

(i) Let J_x and J_y be two jobs with the largest and the second largest processing time on A, respectively, i.e. $\min\{a_x, a_y\} \geq a_i$ for $i = 1, \ldots, n, i \neq x, i \neq y$, and $a_x \geq a_y$.

(ii) Sequence the jobs of J according to RR. Call the resulting schedule π_0. And let $C_H = C_{\max}(\pi_0)$.

(iii) For $p = 1, \ldots, n$, sequence the jobs in the same sequence as that in step (ii) except that J_x is scheduled in the position p. Call the corresponding schedule π_p and let $C_H = \min\{C_H, C_{\max}(\pi_p)\}$. (Note that $\pi_p(p) = J_x$).

(iv) Sequence arbitrarily the jobs of $J\backslash\{J_x, J_y\}$. Then sequence J_x and J_y as the last two jobs such that the completion time of the last one is minimized. Call the resulting schedule π_{n+1} and let $C_H = \min\{C_H, C_{\max}(\pi_{n+1})\}$.

It is clear that Algorithm H can be executed in $O(n \log n)$ time.

Before giving the worst-case error bound of algorithm H, we establish the following two lemmas which will be used in the subsequent analysis.

Lemma 1 *For schedule $\pi_p (0 \leq p \leq n)$, let $J_y = \pi_p(u)$ be the job which starts the last busy period on machine B in π_p. Given an optimal solution $\pi^{\star}$, let $J_z = \pi^{\star}(u')$ be a job in $\pi^{\star}$ such that $S_{zA}(\pi^{\star}) \leq S_{yA}(\pi_p) \leq C_{zA}(\pi^{\star})$.*

(i) *For schedule* $\pi_p(1 \leq p \leq n)$*, assume that* $J_x = \pi_p(p) = \pi^\star(p')$*. If* $(p \geq u$ *and* $p' \geq u')$ *or* $(p < u$ *and* $p' < u')$ *then*

$$\sum_{i=u}^{n} b_{\pi_p(i)} + H_{\pi_p(u)} \leq \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')}.$$

(ii) *For schedule* π_0 *we have*

$$\sum_{i=u}^{n} b_{\pi_0(i)} + H_{\pi_0(u)} \leq \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')}.$$

Proof (i) We note $E = \{\pi_p(i) \,|\, u \leq i \leq n\}$, $F = \{\pi^\star(i) \,|\, u' \leq i \leq n\}$ and $G = E \cap F$.

By definition of job J_z, we have $a(E) \leq a(F)$, then

$$a(E \backslash G) \leq a(F \backslash G). \tag{5}$$

It is clear that whether $(p \geq u$ and $p' \geq u')$ or $(p < u$ and $p' < u')$, we have $J_x \notin E \backslash G$ and $J_x \notin F \backslash G$. As the jobs in π_p, except J_x, are scheduled according to RR, then we have $b_y/a_y \geq b_i/a_i\ \forall J_i \in E \backslash G$, and $b_y/a_y \leq b_i/a_i\ \forall J_i \in F \backslash G$. Hence, and using (5), we derive that

$$\begin{aligned}
\sum_{J_i \in E \backslash G} b_i &= \sum_{J_i \in E \backslash G} a_i \left(\frac{b_i}{a_i}\right) \\
&\leq \sum_{J_i \in E \backslash G} a_i \left(\frac{b_y}{a_y}\right) \\
&\leq \sum_{J_i \in F \backslash G} a_i \left(\frac{b_y}{a_y}\right) \\
&\leq \sum_{J_i \in F \backslash G} a_i \left(\frac{b_i}{a_i}\right) = \sum_{J_i \in F \backslash G} b_i.
\end{aligned}$$

Hence $b(E \backslash G) \leq b(F \backslash G)$ and so

$$\sum_{i=u}^{n} b_{\pi_p(i)} \leq \sum_{i=u'}^{n} b_{\pi^\star(i)}. \tag{6}$$

As J_y starts a busy period on machine B, then (6) implies that $S_{zB}(\pi^\star) \leq S_{yB}(\pi_p)$, for otherwise $\pi^\star$ is not optimal. Hence $H_{\pi_p(u)} \leq H_{\pi^\star(u')}$ and we obtain

$$\sum_{i=u}^{n} b_{\pi_p(i)} + H_{\pi_p(u)} \leq \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')}.$$

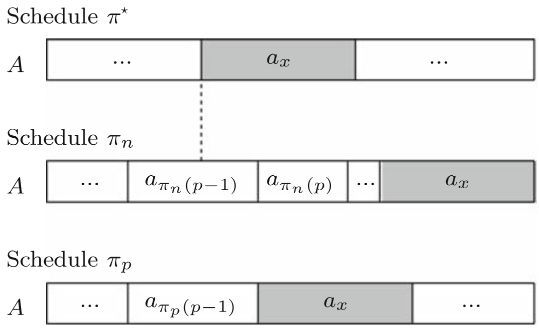


Fig. 2 Schedule π_p satisfying Lemma 2

(ii) Given that all jobs in π_0 are scheduled according to RR, using a similar argument as in (i), it should be easy to show the result. □

Lemma 2 *One of the following two conditions must be realized.*

- J_x *is the first job in* $\pi^\star$.
- *There exists a schedule* $\pi_p, 2 \le p \le n$, *such that* $S_{\pi_p(p-1)A}(\pi_p) < S_{xA}(\pi^\star) \le S_{xA}(\pi_p)$.

Proof Suppose that J_x is not the first job in $\pi^\star$. We consider the schedule π_n (where J_x is scheduled last). Let $\pi_n(p)$ be the first job of π_n such that $S_{\pi_n(p)A}(\pi_n) \ge S_{xA}(\pi^\star)$ (see Fig. 2). It is then easy to see that schedule π_p obtained by scheduling J_x in position p is such that $S_{\pi_p(p-1)A}(\pi_p) < S_{xA}(\pi^\star) \le S_{xA}(\pi_p)$ (see Fig. 2). □

The worst-case performance of algorithm H is given by Theorem 2.

Theorem 2 *For the* $F_2|h(0,q), r, s_q < C^{\text{joh}}|C_{\max}$ *problem, the relative worst-case error bound of algorithm* H *is given by* $C_H/C^\star_{\max} \le 4/3$.

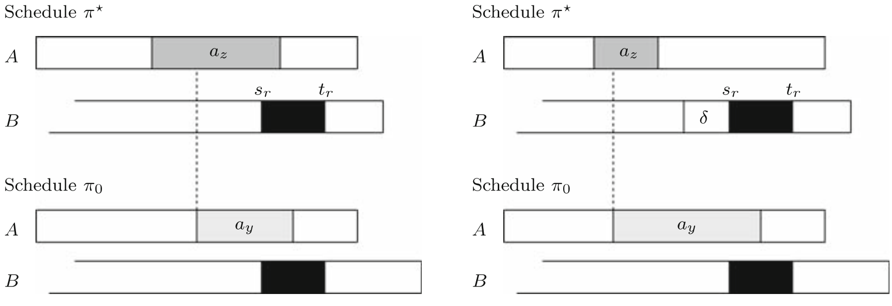
Proof Let $Q = \{J_i \in J | a_i \ge C^\star_{\max}/3\}$. Although it is NP-hard to calculate $C^\star_{\max}$, it is obvious that $|Q| \le 3$. Furthermore, the unique configuration that satisfies $|Q| = 3$, is to have 3 jobs such that $a_1 = a_2 = a_3 = C^\star_{\max}/3$, $b_3 = 0$ and the rest of the jobs such that $a_i = 0$. Considering that this case is obvious, in the remainder of the proof we assume that $|Q| \le 2$.

We are going to consider the following 3 cases.

Case 1 $|Q| = 0$.

We consider the schedule π_0. Let $J_y = \pi_0(u)$ be the job which starts the last busy period on machine B. And let $J_z = \pi^\star(u')$ be a job in $\pi^\star$ such that $S_{zA}(\pi^\star) \le S_{yA}(\pi_0) \le C_{zA}(\pi^\star)$. From Lemma 1(ii) we have

$$\sum_{i=u}^{n} b_{\pi_0(i)} + H_{\pi_0(u)} \le \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')}. \tag{7}$$



(a) Case 1-1 **(b)** Case 1-2

Fig. 3 Schedules π_0 and $\pi^\star$

If J_y satisfies (condition1), then form (3) and (7) we obtain

$$\begin{aligned} C_{\max}(\pi_0) &= C_{yA}(\pi_0) + \sum_{i=u}^{n} b_{\pi_0(i)} + H_{\pi_0(u)} \\ &\leq S_{yA}(\pi_0) + a_y + \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')} \\ &\leq C_{zA}(\pi^\star) + a_y + \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')}. \end{aligned} \tag{8}$$

Given the position of J_z in $\pi^\star$, we have $C_{zA}(\pi^\star) + \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')} \leq C^\star_{\max}$. Hence, using (8) we obtain $C_{\max}(\pi_0) \leq C^\star_{\max} + a_y$. As by assumption $a_y < C^\star_{\max}/3$, then $C_{\max}(\pi_0) \leq 4C^\star_{\max}/3$.

If J_y satisfies (condition2), two cases are to be considered (see Fig. 3)

Case 1-1 $S_{zB}(\pi^\star) \geq t_r$. In this case it is easy to verify that $t_r + \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')} \leq C^\star_{\max}$. Hence, and using (4) and (7), we obtain $C_{\max}(\pi_0) = t_r + \sum_{i=u}^{n} b_{\pi_0(i)} + H_{\pi_0(u)} \leq C^\star_{\max}$.

Case 1-2 $S_{zB}(\pi^\star) < s_r$. We note $\delta = s_r - S_{zB}(\pi^\star)$. By construction we have $\delta \leq a_y \leq C^\star_{\max}/3$, and $t_r + \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')} - \delta \leq C^\star_{\max}$. Hence and using (4) and (7) we have

$$\begin{aligned} C_{\max}(\pi_0) &= t_r + \sum_{i=u}^{n} b_{\pi_0(i)} + H_{\pi_0(u)} \\ &\leq t_r + \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')} - \delta + \delta \\ &\leq C^\star_{\max} + \delta \\ &\leq 4C^\star_{\max}/3. \end{aligned}$$

Case 2 $|Q| = 1$ (i.e. $Q = \{J_x\}$).

We note $J_x = \pi^\star(p') = \pi_p(p)$. We consider step (iii) of the algorithm. According to Lemma 2, one of the following two cases must be realized.

Case 2-1 There exists a schedule $\pi_p, 2 \le p \le n$, such that $S_{\pi_p(p-1)A}(\pi_p) < S_{xA}(\pi^\star) \le S_{xA}(\pi_p)$ (see Fig. 2).

Case 2-2 J_x is the first job in $\pi^\star$.

We consider separately each case.

Case 2-1 Let $J_y = \pi_p(u)$ be the job which starts the last busy period on machine B in schedule π_p.

If $J_y \ne J_x$, let $J_z = \pi^\star(u')$ be a job in $\pi^\star$ such that $S_{zA}(\pi^\star) \le S_{yA}(\pi_p) \le C_{zA}(\pi^\star)$.

It should be clear that if $p \ge u$ then $p' \ge u'$, and if $p < u$ then $p' < u'$. Thus, from Lemma 1(i) we obtain $\sum_{i=u}^{n} b_{\pi_p(i)} + H_{\pi_p(u)} \le \sum_{i=u'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(u')}$. Using exactly the same argument as in case 1, we derive $C_{\max}(\pi_p) \le C^\star_{\max} + a_y \le 4C^\star_{\max}/3$.

Otherwise, $J_y = J_x = \pi_p(p)$. Let $J_z = J_x = \pi^\star(p')$. We have $S_{zA}(\pi^\star) \le S_{yA}(\pi_p) \le C_{zA}(\pi^\star)$ then from Lemma 1(i) we obtain

$$\sum_{i=p}^{n} b_{\pi_p(i)} + H_{\pi_p(p)} \le \sum_{i=p'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(p')}. \tag{9}$$

If J_x follows (condition1) then using (3) and (9) we obtain

$$\begin{aligned} C_{\max}(\pi_p) &= C_{xA}(\pi_p) + \sum_{i=p}^{n} b_{\pi_p(i)} + H_{\pi_p(p)} \\ &\le S_{\pi_p(p-1)A}(\pi_p) + a_{\pi_p(p-1)} + a_x + \sum_{i=p'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(p')} \\ &\le S_{xA}(\pi^\star) + a_{\pi_p(p-1)} + a_x + \sum_{i=p'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(p')}. \end{aligned} \tag{10}$$

Given the position of J_x in $\pi^\star$, we have $S_{xA}(\pi^\star) + a_x + \sum_{i=p'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(p')} \le C^\star_{\max}$. Hence, using (10) we obtain

$$C_{\max}(\pi_p) \le C^\star_{\max} + a_{\pi_p(p-1)} \le 4C^\star_{\max}/3.$$

If J_x follows (condition2), as in case 1, we consider the following two cases:

Case 2-1-1 $S_{xB}(\pi^\star) \ge t_r$. In this case it is easy to verify that as in case 1-1, $t_r + \sum_{i=p'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(p')} \le C^\star_{\max}$. Hence, and using (4) and (9), we obtain $C_{\max}(\pi_p) = t_r + \sum_{i=p}^{n} b_{\pi_p(i)} + H_{\pi_p(p)} \le C^\star_{\max}$.

Case 2-1-2 $S_{xB}(\pi^\star) < s_r$. We again consider $\delta = s_r - S_{xB}(\pi^\star)$. We have

$$\begin{aligned}\delta &= s_r - S_{xB}(\pi^\star)\\ &\le C_{xA}(\pi_p) - C_{xA}(\pi^\star)\\ &\le S_{xA}(\pi_p) - S_{xA}(\pi^\star)\\ &\le a_{\pi_p(p-1)} \le C^\star_{\max}/3,\end{aligned}$$

and $t_r + \sum_{i=p'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(p')} - \delta \le C^\star_{\max}$. Hence and using (4) and (9) we have

$$\begin{aligned}C_{\max}(\pi_p) &= t_r + \sum_{i=p}^{n} b_{\pi_p(i)} + H_{\pi_p(p)}\\ &\le t_r + \sum_{i=p'}^{n} b_{\pi^\star(i)} + H_{\pi^\star(p')} - \delta + \delta\\ &\le C^\star_{\max} + \delta \le 4C^\star_{\max}/3.\end{aligned}$$

Case 2-2 We consider π_1 where J_x is scheduled first as in $\pi^\star$. Let $J_y = \pi_1(u)$ be the job which starts the last busy period on machine B in schedule π_1.

If $J_y \ne J_x$ then using Lemma 1(i), it can be shown that $C_{\max}(\pi_1) \le C^\star_{\max} + a_{\pi_1(u)} \le 4C^\star_{\max}/3$. Otherwise we have $C_{\max}(\pi_1) = a_x + b(J) + H_{\pi_1(1)} = a_x + b(J) + H_{\pi^\star(1)} \le C^\star_{\max}$.

Case 3 $|Q| = 2$.

In this case, we have

$$a(J\backslash Q) = a(J) - a(Q) \le C^\star_{\max}/3. \tag{11}$$

We consider the schedule π_{n+1}. Let $J_y = \pi_{n+1}(u)$ be the job which starts the last busy period on machine B.

If $J_y \notin Q$, then it should be clear that either J_y follows (condition1) or (condition2) we have $C_{\max}(\pi_{n+1}) \le a(J\backslash Q) + b(J) + H_T$. Hence, using (2) and (11) we obtain $C_{\max}(\pi_{n+1}) \le 4C^\star_{\max}/3$.

Otherwise $J_y \in Q$. As the jobs of Q are scheduled such as to minimize the total completion time, then the completion time of jobs in Q considered alone should be no greater than $C^\star_{\max}$. Hence we have $C_{\max}(\pi_{n+1}) \le a(J\backslash Q) + C^\star_{\max} \le 4C^\star_{\max}/3$.

We then conclude that

$$C_H = \min_{0\le p\le n+1}\{C_{\max}(\pi_p)\} \le 3C^\star_{\max}/2$$

The proof is complete. □

Although we are not able to show that the bound is tight, the following instance shows that the bound cannot be better than 5/4. We consider the following problem instance with $n = z + 2$ and $q = 1$. Let $a_1 = a_2 = a_3 = z, b_1 = b_2 = z, b_3 = 1$, $a_i = b_i = 1$ for $4 \leq i \leq n, s_1 = z, t_1 = 2z - 1$ where $z > 2$. Without loss of generality, we set $J_x = J_2$ and $J_y = J_3$.

It is clear that step (ii) may result in schedule $\pi_0 = \langle J_1, J_4, \ldots, J_n, J_2, J_3 \rangle$ with $C_{\max}(\pi_0) = 5z - 1$.

For $p = 1, \ldots, n$, step (iii) may result in schedules $\pi_1 = \langle J_2, J_1, J_4, J_5, \ldots, J_n, J_3 \rangle$, $\pi_2 = \langle J_1, J_2, J_4, J_5, \ldots, J_n, J_3 \rangle$, $\pi_3 = \langle J_1, J_4, J_2, J_5, \ldots, J_n, J_3 \rangle, \ldots$ and $\pi_n = \langle J_1, J_4, J_5, \ldots, J_n, J_3, J_2 \rangle$ with $C_{\max}(\pi_p) = 5z - 1$ for $p = 1, \ldots, n$.

Step (iv) may result in schedule $\pi_{n+1} = \pi_0$ with $C_{\max}(\pi_{n+1}) = 5z - 1$. Hence $C_H = 5z - 1$. However the optimal solution is $\pi^\star = \langle J_4, \ldots, J_n, J_1, J_2, J_3 \rangle$ with $C^\star_{\max} = 4z$. We see that $C_H / C^\star_{\max}$ goes to 5/4 as z approaches to infinity.

References

1. Breit, J.: An improved approximation algorithm for two-machine flow shop scheduling with an availability constraint. Inf. Process. Lett. **90**, 273–278 (2004)
2. Breit, J., Schmidt, G., Strusevich, V.A.: Non-preemptive two-machine open shop scheduling with non-availability constraints. Math. Methods Oper. Res. **57**, 217–234 (2003)
3. Cheng, T.C.E., Wang, G.: An improved heuristic for two-machine flowshop scheduling with an availability constraint. Oper. Res. Lett. **26**, 223–229 (2000)
4. Cheng, T.C.E., Wang, G.: Two-machine flow shop scheduling with consecutive availability constraints. Inf. Process. Lett. **71**, 49–54 (1999)
5. Johnson, S.M.: Optimal two- and three-stage production schedules with setup times included. Res. Log. Q. **1**, 61–68 (1954)
6. Kubiak, W., Blazewicz, J., Formanowicz, P., Breit, J., Schmidt, G.: Two-machine flow shops with limited machine availability. Eur. J. Oper. Res. **136**, 528–540 (2002)
7. Kubzin, M.A., Potts, C.N., Strusevich, V.A.: Approximation results for flow shop scheduling problems with machine availability constraints. Comput. Oper. Res. **36**, 379–390 (2009)
8. Lee, C.Y.: Minimizing the makespan in the two-machine flow shop scheduling problem with an availability constraint. Oper. Res. Lett. **20**, 129–139 (1997)
9. Lee, C.Y.: Two-machine flowshop scheduling with availability constraints. Eur. J. Oper. Res. **114**, 420–429 (1999)
10. Ng, C.T., Kovalyov, M.Y.: An FPTAS for scheduling a two-machine flowshop with one unavailability interval. Naval Res. Log. **51**, 307–315 (2004)