Theoretical Computer Science 407 (2008) 389–399

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Dense open-shop schedules with release times

Rongjun Chen [a,*], Wanzhen Huang [b], Guochun Tang [c]

[a] School of Sciences, Changzhou Institute of Technology, Changzhou, Jiangsu, 213002, China

[b] Department of Mathematical Sciences, Lakehead University, Thunder Bay, Ontario, Canada P7B5E1

[c] Institute of Management Engineering, Shanghai Second Polytechnic University, Shanghai, 201209, China

ARTICLE INFO

Article history:
Received 14 March 2008
Received in revised form 28 June 2008
Accepted 23 July 2008
Communicated by D.-Z. Du

Keywords:
Scheduling
Open-shop
Dense schedule
Performance ratio
Job release time

ABSTRACT

We study open-shop scheduling problems with job release times. The objective is to minimize the makespan. Dense schedules, easy to construct, are often used as approximate solutions. Performance ratios of the makespans from dense schedules and that of the optimal schedule of the problem are used to evaluate the quality of approximate schedules. It is conjectured (proved for a limited number of machines) that this performance ratio is bounded above by $(2-1/m)$ for m-machine open-shop problems without job release times. In this paper, we extend the conjecture to open-shop problems with job release times. The results proved in this paper are: 1. Dense schedule performance ratio is bounded above by 2 for three-machine open-shop problems with job release times; 2. The conjectured performance ratio upper bound of 5/3 is proved for two special cases of three-machine open-shop problems with job release times; 3. A performance ratio upper bound of 7/4 is proved for three-machine problems.

1. Introduction

In a shop scheduling model we are given $m \geq 2$ machines and n jobs. Each job consists of a number of operations, each to be processed on a specified machine for a specific amount of time. Each machine processes no more than one job at a time, and each job is processed on at most one machine at a time. No pre-emption is allowed in processing any operation, i.e. once started, the processing of any operation must not be interrupted before it is completed.

Traditionally, in scheduling theory three basic shop scheduling models are considered: job-shop, flow-shop and open-shop. In the job-shop setting, each job consists of a chain of a number of operations with specified order, each of which should be processed by a specified machine. The flow-shop is a special case of the job-shop where the operation chains of all jobs consist of exactly m operations, one on each of m machines, and the orders are the same for all these jobs. The open-shop differs from the flow-shop in the sense that the operations of jobs can be processed in any order.

For each of the three models, one of the well-known objectives is to minimize the makespan, the time when the last job is completed. Following a standard notation [4], we denote these problems by $J||C_{\max}$, $F||C_{\max}$, and $O||C_{\max}$, respectively. All these three problems are strongly NP-hard [4]. Therefore, it is important to develop efficient polynomial approximation algorithms. Usually, the quality of a polynomial approximation algorithm is measured by its worst-case performance ratio $\sup \frac{C_{\max}(S)}{C_*}$, where $C_{\max}(S)$ is the makespan of a schedule S found by the algorithm, C_* is the corresponding minimum possible makespan, and the supremum is taken over all problem instances.

Recently, it has been discovered that finding a guaranteed good approximate solution for any shop scheduling problem is as difficult as finding an optimal one. It is shown in [2] that, unless $P = NP$, there is no polynomial approximation algorithm for any of these shop scheduling problems with a worst-case performance ratio less than 5/4.

* Corresponding author. Tel.: +86 0519 88596322.
E-mail addresses: chenrjecust@163.com (R. Chen), wzhuang@sleet.lakeheadu.ca (W. Huang), gtang@sh163.net (G. Tang).

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2008.07.030

Nevertheless, there is an interesting simple class of shop schedules that can be constructed by a greedy algorithm. A schedule is said to be dense, if it satisfies the following condition: If any machine M_i is idle at time t, then all jobs, that are processed on M_i after time t must be processed at time t on machines other than M_i. It is observed [5] that the makespan of any dense open-shop schedule is at most twice that of the optimal one. It is conjectured that, for $m \geq 2$, the makespan of any dense open-shop schedule is at most $2 - 1/m$ times the optimal makespan [10]. This conjecture has been proved for up to $m \leq 6$ [8]. It is shown in [1,6] that the bound of $2 - 1/m$ cannot be further reduced.

Up until now, all jobs are assumed to be available at time zero. Consider the open-shop with jobs arriving at different release times, no pre-emptions allowed, and the makespan as the objective to be minimized — that is, $Om|r_j|C_{\max}$. Lawer et al. [3] prove that $O2|r_j|C_{\max}$ is strong NP-hard. In this paper we study dense schedules for the problem with arrival times, which can be easily constructed by the following greedy algorithm: being from the initial time and whenever a machine becomes available, select one of the released and available jobs to process on the machine and avoid unnecessary idleness. Obviously dense schedule is not unique. We conjecture that the makespan of any dense schedule is at most $2 - 1/m$ times that of the optimal one when jobs are released at different times. Chen [7] proves that the performance ratio of any dense schedule is bounded above by $3/2$ for problem $O2|r_j|C_{\max}$. In this paper we study the performance ratio of dense schedules for $O3|r_j|C_{\max}$.

The rest of this paper is organized as follows. After introducing necessary notations and giving preliminaries in Section 2, we prove that the conjecture is true for two special kinds of dense schedules for problem $O3|r_j|C_{\max}$. A weaker bound of 2 is established in Section 3. In Section 4 we improve the bound of 2 to $\frac{7}{4}$ in general and to $\frac{5}{3}$ for its special cases. We conclude the paper in Section 5 and provide the complete proof of Theorem 4 in the Appendix.

2. Notation and preliminaries

We now formally describe the open-shop scheduling model. Let $M = \{M_1, M_2, \ldots, M_m\}$ be the set of machines and $N = \{1, 2, \ldots, n\}$ the set of jobs. Each job j is available at time r_j, consists of a set $\{O_{1,j}, \ldots, O_{m,j}\}$ of operations, and operation $O_{i,j}$ has to be processed on machine M_i for $p_{i,j}$ time units. Without loss of generality, we assume that

$$0 = r_1 \leq r_2 \leq \cdots \leq r_n.$$

The workload W_i of machine M_i is the total processing time of all operations assigned to the machine, i.e., $W_i = \sum_{j=1}^{n} p_{i,j}$. The length P_j of job j is the total processing time of all the operations of the job j, i.e.,$P_j = \sum_{i=1}^{m} p_{i,j}$.

A schedule S of the problem can be identified as a set of intervals $\{I(O_{i,j}), i = 1, \ldots, m, j = 1, \ldots, n\}$, where $I(O_{i,j}) = [B_{i,j}, C_{i,j})$, $B_{i,j}$ and $C_{i,j}$ is the start and completion time of operation $O_{i,j}$, respectively, with $C_{i,j} - B_{i,j} = p_{i,j}$. A machine M_i is said to be busy at time t if $t \in \bigcup_{j=1}^{n} [B_{i,j}, C_{i,j})$, otherwise we say that machine M_i is idle at time t. Let $C_{\max}(S)$ and C_* denote the makespan of S and the optimal makespan of the problem, respectively. It is easy to verify the following.

$$C_* \geq \max_{j \in N} (r_j + P_j), \tag{2.1}$$

$$C_* \geq \max_{1 \leq j \leq n} \max_{1 \leq i \leq m} \left\{ r_j + \sum_{k=j}^{n} p_{i,k} \right\}, \tag{2.2}$$

and

$$C_* \geq \max_{1 \leq i \leq m} \left\{ \sum_{k=1}^{n} p_{i,k} \right\} = \max_{1 \leq i \leq m} \{W_i\}, \tag{2.3}$$

which is a special case of (2.2) when $j = 1$.

Definition 2.1. An idle interval $I = [b, e)$ on machine M_i for a given schedule S is called **reasonable** if one of the following conditions holds for job j, $j = 1, 2, \ldots, n$,

(1) Job j has been finished on machine M_i before time b, i.e. $C_{i,j} \leq b$; or

(2) Job j is being processed on a machine other than M_i at any time t in I , i.e. , $I \subseteq \bigcup_{i' \neq i} [B_{i',j}, C_{i',j})$; or

(3) Job j released after time e, i.e. $r_j \geq e$.

A schedule S is **dense** if all idle intervals are reasonable.

Without loss of generality, we may assume that release time $r_j, j = 1, 2, \ldots, n$ does not belong to any idle interval. If it happens, we can simply break the idle interval into smaller intervals at the release times.

For schedule S, we also define

$$L_i(t', t'') = \text{total idle time on machine } M_i \text{ between times } t' \text{ and } t''$$

for $t' \leq t'', i = 1, 2, \ldots, m$.

Obviously,

$$L_i(0, r_j) \leq r_j, \quad i = 1, 2, \ldots, m, j = 1, 2, \ldots, n. \tag{2.4}$$

Let S be a dense schedule for problem $Om|r_j|C_{\max}$. The following three lemmas are straightforward by the definition of a dense schedule.

Lemma 1. *If $O_{i,j}$ is processed in $[B_{i,j}, C_{i,j})$, then*

$$L_i(r_j, B_{i,j}) \le \sum_{\{k: O_{k,j} \in F_{i,j}\}} p_{k,j},$$

where $F_{i,j} = \{O_{k,j} : C_{k,j} \le B_{i,j}\}$ is the set of operations for job j that have been finished before operation $O_{i,j}$ starts.

Lemma 2. *If m machines are not idle simultaneously in a time interval $[a, b)$, then*

$$\sum_{i=1}^{m} L_i(a, b) \le (m-1)(b-a).$$

Lemma 3. *Let $M_{\text{idle}}(t) \ge 1$ be the number of machines that are idle at time t. Then there are at most $m - M_{\text{idle}}(t)$ jobs that are released before t will be processed on these idle machines after t.*

As a special case when $M_{\text{idle}}(t) = m$, i.e. all machines are idle at time t, then jobs released before t must all be completed by time t.

3. Weaker bound for $O3|r_j|C_{\max}$

We now study the performance ratio of a dense schedule for three-machine open-shop problem with job release times. Without loss of generality, we assume that machine M_3 terminates the dense schedule S.

In the first theorem, we prove that the conjecture is true if there is no idle time on M_3 after the last job's release time r_n.

Theorem 1. *Suppose there is no idle time on M_3 after r_n. Then*

$$C_{\max}/C_* \le 2 - \frac{1}{3}. \tag{3.1}$$

Proof. If there is no idle time on machine M_3, then S is optimal, since

$$C_{\max} = L_3(0, C_{\max}) + W_3 = W_3 \le C_*.$$

Otherwise, let $I = [b, e)$ be the last idle interval on machine M_3, then $I \subseteq [r_{k-1}, r_k)$ for some job k, $2 \le k \le n$.

If all jobs in $\{1, 2, \ldots, k-1\}$ are finished before time e, then $e = r_k$. From inequality (2.2) we have

$$C_{\max} = r_k + p_{3,k} + p_{3,k+1} + \cdots + p_{3,n} \le C_*,$$

which implies (3.1). Otherwise, we assume that there are g jobs in $\{1, 2, \ldots, k-1\}$ that still need to be processed on M_3 starting from time e. Because S is dense, these jobs must be on other machines during the time interval I, and therefore, $g \le 2$. We assume these jobs to be $\{j_1, \ldots, j_g\}$, with

$$1 \le j_1 < \cdots < j_g \le k-1.$$

We consider the following two cases.

Case 1. There is a job $j \in \{j_1, \ldots, j_g\}$ such that $p_{3,j} \ge C_*/3$.

On M_3 which terminates the schedule, we have

$$\begin{aligned} C_{\max} &= W_3 + L_3(0, r_j) + L_3(r_j, e) \\ &\le C_* + r_j + p_{1,j} + p_{2,j} \quad \text{(by Lemma 1)} \\ &= C_* + r_j + P_j - p_{3,j} \\ &\le (2 - 1/3)C_* \quad \text{(by (2.1))}, \end{aligned}$$

which proves (3.1).

Case 2. For any

$$j \in \{j_1, \ldots, j_g\}, \qquad p_{3,j} < C_*/3. \tag{3.2}$$

We get

$$\begin{aligned} C_{\max} &\le r_k + p_{3,j_1} + \cdots + p_{3,j_g} + p_{3,k} + p_{3,k+1} + \cdots + p_{3,n} \\ &\le C_* + p_{3,j_1} + \cdots + p_{3,j_g} \quad \text{(by (2.2))} \\ &\le C_* + (g/3)C_*, \quad \text{(by (3.2))} \end{aligned}$$

which, together with $g \le 2$, implies (3.1). □

In the next theorem, we consider a dense schedule with idle times on machine M_3 after the last job's release time.

Theorem 2. *Suppose there exist idle times on machine M_3 after time r_n. If there is a job j processed on M_3 after the last idle time with $p_{3,j} \ge C_*/3$, then (3.1) holds.*

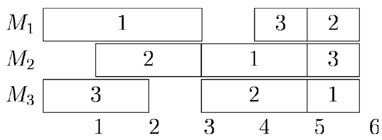

Fig. 1. The optimal schedule of the 3-job instance with makespan of 6.

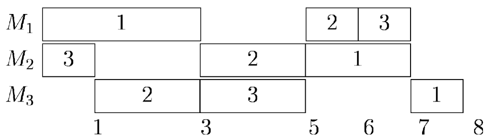

Fig. 2. A dense schedule of the 3-job instance with makespan of 8.

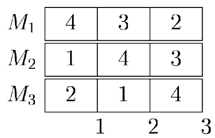

Fig. 3. The optimal schedule of the 4-job instance with makespan of 3.

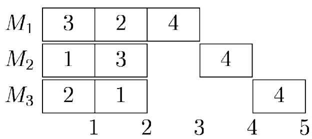

Fig. 4. A dense schedule of the 4-job instance with makespan of 5.

Proof. The proof is similar to that of Case 2 in Theorem 1. □

In Theorem 3, we can see that a weaker bound of 2 holds for three-machine open-shop with job release times.

Theorem 3. *The makespan of any dense schedule for problem $O3|r_j|C_{\max}$ is less than 2 times that of the optimal schedule.*

Proof. Let $j \in N$ terminate the schedule and its last operation processed on M_3, i.e. $C_{\max} = C_{3,j}$. We have

$$\begin{aligned} C_{\max} &\le r_j + W_3 + L_3(r_j, C_{\max}) \\ &\le r_j + W_3 + p_{1,j} + p_{2,j} \quad \text{(by Lemma 1)} \\ &< r_j + W_3 + P_j \le C_* + C_* = 2C_*, \end{aligned}$$

by inequalities (2.1) and (2.3). □

The following instance shows the bound of 2 is not tight.

Instance. Consider three machines and three jobs with the processing times specified by vectors P_1, P_2, P_3, respectively,

$$P_1 = (3, 2, 1), \qquad P_2 = (1, 2, 2), \qquad P_3 = (1, 1, 2),$$

and the release times are $r_1 = r_3 = 0, r_2 = 1$. For the problem, an optimal schedule and a dense schedule are given in Figs. 1 and 2, respectively. We have $C_{\max}/C_* = \frac{4}{3} < 2$.

If we consider four jobs with the processing times specified by vectors P_1, P_2, P_3, P_4, respectively,

$$P_1 = (0, 1, 1), \qquad P_2 = (1, 0, 1), \qquad P_3 = (1, 1, 0), \qquad P_4 = (1, 1, 1),$$

and the release times are $r_1 = r_2 = r_3 = 0$. For this problem, an optimal schedule and a dense schedule are given in Figs. 3 and 4, respectively. We have $C_{\max}/C_* = \frac{5}{3} < 2$. □

Remark: The results of Theorems 1–3 can be easily generalized to m machines.

4. Improved bound for $O3|r_j|C_{\max}$

In this section we further consider the three-machine open-shop problems with job release times. Assume that machine M_3 terminates the schedule. When $m = 3$, we have $2 - 1/m = 5/3$. The next theorem indicates that the ratio is 7/4, the difference of which from 5/3 is only 0.08333.

Theorem 4. *The makespan of any dense schedule for the three-machine open-shop with job release times is at most 7/4 times that of an optimal schedule.*

Proof. Detailed proof will be given in the Appendix because of its length. □

Now we give two classes of three-machine problems that the performance ratio can actually achieve the bound of 5/3.

The first case is when the processing times are independent of machines. The problem is denoted by $O3|r_j, p_{i,j} = p_j|C_{\max}$ which is proved to be NP-hard [11].

Theorem 5. *For any dense schedule of problem* $O3|r_j, p_{i,j} = p_j|C_{\max}$*, there holds that* $C_{\max}/C_* \le 5/3$.

Proof. We only need to consider Case 2 in the proof of Theorem 4. Let $I = [b, e)$ be the last idle interval on machine M_3. Because of the density of S, j_1 and j_2 have been processed right before e on M_2 and M_1, respectively. Due to the condition of Case 2, any two machines have no common idle between $[r_{j_2}, e)$ and M_1 and M_2 have no common idle during $[r_{j_1}, r_{j_2})$.

First let us consider the subcase in which $p_{1,j_1} \le p_{1,j_2}$. Suppose $L_2(r_{j_2}, e) = 0$. We have

$$C_{\max} \le C_* + L_1(0, r_{j_2}) + L_1(r_{j_2}, e) + p_{3,j_1} + p_{3,j_2}, \tag{4.1}$$

$$C_{\max} \le C_* + L_2(0, r_{j_2}) + p_{3,j_1} + p_{3,j_2}, \tag{4.2}$$

$$C_{\max} \le C_* + L_3(0, r_{j_2}) + L_3(r_{j_2}, e). \tag{4.3}$$

Summing up (4.1)–(4.3) and taking into account the following two inequalities:

$$L_1(0, r_{j_2}) + L_2(0, r_{j_2}) + L_3(0, r_{j_2}) \le 2r_{j_2}$$

and

$$L_1(r_{j_2}, e) + L_3(r_{j_2}, e) \le p_{1,j_2} + p_{2,j_2},$$

we get

$$\begin{aligned} 3C_{\max} &\le 3C_* + 2r_{j_2} + p_{j_2} + p_{3,j_2} + 2p_{3,j_1} \\ &\le 3C_* + 2(r_{j_2} + p_{j_2}) \le 5C_*, \end{aligned}$$

which proves the theorem.

Next we assume $L_2(r_{j_2}, e) > 0$. According to the density, $L_1(r_{j_2}, e) = 0$. We can prove Theorem 5 similarly by exchanging M_1 and M_2.

Now consider the other subcase in which $p_{1,j_1} > p_{1,j_2}$. Suppose that M_1 and M_3 have common idle during $[r_{j_1}, r_{j_2})$ and let $I_1 = [b_1, e_1)$ be the last common idle interval. We have $I_1 \subseteq [r_{l-1}, r_l)$, where $r_{j_1} < r_l \le r_{j_2}$. After e_1, there is at most one job j_1 released before r_l is processed on M_3. According to the density, we have $L_1(r_{j_2}, e) = L_2(r_{j_2}, e) = 0$ and the following two inequalities

$$\begin{aligned} C_{\max} &\le r_l + \sum_{k=l}^{n} p_{1k} + L_1(r_l, r_{j_2}) + p_{3,j_1} + p_{3,j_2} \\ &\le C_* + L_1(r_l, r_{j_2}) + p_{3,j_1} + p_{3,j_2}, \end{aligned} \tag{4.4}$$

$$\begin{aligned} C_{\max} &= W_3 + L_3(0, r_{j_1}) + L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e) \\ &\le C_* + r_{j_1} + L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e). \end{aligned} \tag{4.5}$$

Summing up (4.4) and (4.5), together with the fact that

$$L_1(r_l, r_{j_2}) + L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e) \le p_{1,j_1} + p_{2,j_1},$$

we have

$$2C_{\max} \le 2C_* + r_{j_1} + p_{j_1} + p_{3,j_2} \le 10/3C_*,$$

which proves Theorem 5.

Next we assume M_1 and M_3 have no common idle during $[r_{j_1}, r_{j_2})$. At this time, if M_2 and M_3 have common idle during $[r_{j_1}, r_{j_2})$ and let $I_1 = [b_1, e_1)$ be the last common idle interval, then we have $I_1 \subseteq [r_{l-1}, r_l)$, where $r_l \le r_{j_2}$. Let μ be the length of the processing time for O_{1,j_1} before r_l (if any). From the density, it follows that

$$L_1(r_{j_2}, e) + L_3(r_{j_2}, e) \le p_{1,j_2} + p_{2,j_2},$$

$$L_2(r_{j_1}, r_l) + L_3(r_{j_1}, r_l) \le r_l - r_{j_1} + \mu \tag{4.6}$$

and

$$L_1(r_l, r_{j_2}) + L_2(r_l, r_{j_2}) + L_3(r_l, r_{j_2}) \le r_{j_2} - r_l. \tag{4.7}$$

In addition, after e_1 both M_2 and M_3 process one job j_1 released before r_l. Thus the idle length of M_2 after r_l is less than $p_{1,j_1} + p_{2,j_1} - \mu$. We obtain

$$C_{\max} \le C_* + L_1(0, r_{j_1}) + L_1(r_l, r_{j_2}) + L_1(r_{j_2}, e) + p_{3,j_1} + p_{3,j_2},$$

$$C_{\max} \le C_* + L_2(0, r_{j_1}) + L_2(r_{j_1}, r_l) + p_{1,j_1} + p_{2,j_1} - \mu,$$

and

$$C_{\max} \leq C_* + L_3(0, r_{j_1}) + L_3(r_{j_1}, r_l) + L_3(r_l, r_{j_2}) + L_3(r_{j_2}, e).$$

Summing up the inequalities above, together with Lemma 2, (4.6) and (4.7), we have

$$\begin{aligned} 3C_{\max} &\leq 3C_* + 2r_{j_1} + p_{j_1} + p_{3,j_2} + p_{1,j_1} + p_{2,j_1} \\ &\leq 3C_* + 2(r_{j_1} + p_{j_1}) \leq 5C_*, \end{aligned}$$

which proves the bound of 5/3. Otherwise, if M_2 and M_3 have no idle during $[r_{j_1}, r_{j_2})$, we can prove Theorem 5 similarly to the subcase where $p_{1,j_1} \leq p_{1,j_2}$. □

The second special class of problems is that the number of operations is no more than 2 for all jobs, which is also NP-hard [9].

Theorem 6. *For problem $O3|r_j|C_{\max}$, where the number of operations is no more than 2 for all jobs, there holds that $C_{\max}/C_* \leq 5/3$.*

Proof. Similarly to the proof of Theorem 5, we only need to prove for Case 2. According to the density, j_1 and j_2 have been processed right before e on M_2 and M_1, respectively. We have $L_2(r_{j_1}, r_{j_2}) = L_2(r_{j_2}, e) = 0$. There are two subcases we have to consider.

Case 2.1. M_2 and M_3 have no common idle. This yields

$$C_{\max} \leq C_* + L_2(0, r_{j_1}) + p_{3,j_1} + p_{3,j_2}, \tag{4.8}$$

and

$$C_{\max} \leq C_* + L_3(0, r_{j_1}) + L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e). \tag{4.9}$$

Summing up (4.8) and (4.9), together with

$$L_2(0, r_{j_1}) + L_3(0, r_{j_1}) \leq r_{j_1}$$

and

$$L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e) \leq p_{2,j_1},$$

we have

$$2C_{\max} \leq 2C_* + r_{j_1} + p_{j_1} + p_{3,j_2} \leq 10/3C_*.$$

Case 2.2. M_2 and M_3 have common idle. Let $I_1 = [b_1, e_1)$ be the last common idle. We have $I_1 \subseteq [r_{l-1}, r_l)$, where $r_l \leq r_{j_1}$. After e_1, there is at least one machine of M_2 and M_3 that does not process any job released before r_l, say M_2. We obtain that,

$$\begin{aligned} C_{\max} &\leq r_l + \sum_{k=l}^{n} p_{2,k} + L_2(r_l, r_{j_1}) + p_{3,j_1} + p_{3,j_2} \\ &\leq C_* + L_2(r_l, r_{j_1}) + p_{3,j_1} + p_{3,j_2}, \end{aligned} \tag{4.10}$$

and

$$C_{\max} \leq r_l + L_3(r_l, r_{j_1}) + L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e) + C_*. \tag{4.11}$$

Summing up (4.10) and (4.11), together with

$$r_l + L_3(r_l, r_{j_1}) + L_2(r_l, r_{j_1}) \leq r_{j_1}$$

and

$$L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e) \leq p_{2,j_1},$$

we get

$$2C_{\max} \leq 2C_* + r_{j_1} + p_{j_1} + p_{3,j_2} \leq 10/3C_*,$$

which completes the proof. □

5. Conclusion

In this paper, we study dense schedules for three-machine open-shop problems with job release times. We prove that the makespan of a dense schedule is at most $\frac{5}{3}$ times the optimum schedule if the dense schedule S satisfies either of the following conditions:

1. There is no idle time on machine M_3 after the last job's release time; or
2. There is a job j processed after the last idle interval on M_3 with the processing time $p_{3,j} \geq C_*/3$.

We also prove the bound of 2 is an upper bound for the worst-case performance ratio for problem $O3|r_j|C_{\max}$. An improved bound of 7/4 is proved in general, and a bound of 5/3 is proved for special cases where the processing times are independent of machines or the number of operations is no more than 2.

Acknowledgments

The first and the third authors were supported by the National Natural Science Foundation of China under grant number 20710015. The second author was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

The first author would also like to acknowledge Dr. Wenci Yu, late professor at East China University of Science and Technology, for his valuable advice during the initial stage of this research work.

Appendix. Proof of Theorem 4

Theorem 4. *The makespan of any dense schedule for the three-machine open-shop with job release times is at most* 7/4 *times that of an optimal schedule.*

Proof. Without loss of generality, assume machine M_3 terminates the schedule. If there is no idle time on machine M_3 after r_n, or there exist idle times on M_3 after r_n and a job j processed after the last idle interval on M_3 with $p_{3,j} \geq C_*/3$, then the theorem is proved due to Theorem 1 or 2, respectively.

Next we consider the situation in which there exist idle times on machine M_3 after r_n and all jobs j processed after the last idle time on M_3 satisfy $p_{3,j} < C_*/3$.

We use mathematical induction to prove the theorem for number of jobs n.

For $n = 1$, the inequality is trivial because $C_{\max} = P_1 = C_*$. We now assume that the theorem is true for up to $n - 1$ jobs. We need to prove the theorem for n jobs.

If all three machines are idle at a certain time, let $I = [b, e)$ be the last such idle interval, then $I \subseteq [r_{k-1}, r_k)$ for some $k \in \{2, 3, \ldots, n\}$. Then by Lemma 3, all jobs in $\{1, 2, 3, \ldots, k-1\}$ are finished before time b, and other jobs are not released until $e = r_k$. We now consider a new problem with only the last $n - k + 1$ ($<n$) jobs, with the same processing times and modified release times as $r'_j = r_j - r_k$, for all $j = k, k + 1, \ldots, n$. Let C'_* be the optimal makespan of the new problem with $n - k + 1$ jobs. It is obvious that

$$C_* = C'_* + r_k.$$

For any dense schedule of the original problem, the processing of the last $n - k + 1$ jobs forms a dense schedule for the new problem, and obviously we have $C_{\max} = C'_{\max} + r_k$. Since the new problem has less than n jobs, by the induction we have $C'_{\max}/C'_* < 7/4$. Therefore

$$C_{\max}/C_* = (C'_{\max} + r_k)/(C'_* + r_k) \leq C'_{\max}/C'_* < 7/4,$$

since $C'_{\max} \geq C'_*$.

Now we assume that three machines are never idle at the same time. Let $I = [b, e)$ be the last idle interval on M_3 after the last job's release time. i.e. $b \geq r_n$. The remaining proof is to exhaustively analyze all possible cases. For each possible case, we will give an upper bound for the performance ratio. At the end, we will find that the largest upper bound for the three-machine case is 7/4.

Since $I \in [r_n, C_{\max})$ and the schedule is dense, we know that there are at most two jobs on M_3 after time e. Based on the number of jobs after time e on machine M_3, we consider Case 1 and Case 2 as follows.

Case 1. There is only one job, say job j, processed after the last idle interval $I = [b, e)$ on machine M_3. Because the schedule is dense, job j must be just completed from one of the other machines at time e, say M_2. i.e. $[B_{2,j}, C_{2,j}) = [B_{2,j}, e)$.

Case 1.1. $C_{1,j} = B_{2,j}$. Because of the density, we know

$$L_i(r_j, B_{1,j}) = 0, \quad i = 1, 2, 3.$$

On each of the three machines, we get

$$C_{\max} \leq C_* + L_1(0, r_j) + p_{2,j} + p_{3,j},$$
$$C_{\max} \leq C_* + L_2(0, r_j) + p_{1,j} + p_{3,j},$$
$$C_{\max} \leq C_* + L_3(0, r_j) + p_{1,j} + p_{2,j}.$$

Summing up the three inequalities above, noting the fact that there is no common idle time on all three machines which implies

$$L_1(0, r_j) + L_2(0, r_j) + L_3(0, r_j) \leq 2r_j,$$

we have

$$3C_{\max} \leq 3C_* + 2r_j + 2P_j \leq 5C_*,$$

or

$$C_{\max}/C_* \leq 5/3.$$

Case 1.2. $C_{1,j} < B_{2,j}$. In this case, we have $B_{2,j} \le b$. It is easy to see that

$$L_i(r_j, B_{1,j}) = 0, \quad i = 1, 2, 3,$$

and

$$L_i(C_{1,j}, B_{2,j}) = 0, \quad i = 2, 3.$$

The following three situations should be considered.

Case 1.2.1. M_2 and M_3 have no common idle time up to $C_{1,j}$.

In this case,

$$L_2(0, C_{1,j}) + L_3(0, C_{1,j}) \le r_j + p_{1,j}.$$

We also have

$$C_{\max} \le C_* + L_2(0, C_{1,j}) + p_{3,j}$$

and

$$C_{\max} \le C_* + L_3(0, C_{1,j}) + p_{2,j}.$$

Summing up the two, we get

$$2C_{\max} \le 2C_* + r_j + p_{1,j} + p_{2,j} + p_{3,j} \le 3C_*,$$

or

$$C_{\max}/C_* \le 3/2.$$

Case 1.2.2. M_2 and M_3 have common idle time before $C_{1,j}$ and the last common idle interval $I_1 = [b_1, e_1) \subseteq [B_{1,j}, C_{1,j})$. Also we assume $I_1 \subseteq [r_{l-1}, r_l)$ for some $l > j$. After I_1, machines M_2 and M_3 can process at most one job released before I_1 (Lemma 3), which can only be job j. Therefore $e_1 = r_l$. Also we know that all jobs released before r_l, except job j, are finished from M_2 and M_3 by then. On machines M_2 and M_3, we have

$$\begin{aligned} C_{\max} &\le C_* + L_2(0, r_j) + L_2(r_j, r_l) + L_2(r_l, C_{1,j}) + p_{3,j} \\ &\le C_* + r_j + L_2(r_j, r_l) + L_2(r_l, C_{1,j}) + p_{3,j}, \\ C_{\max} &\le r_l + \sum_{k=l}^{n} p_{3,k} + L_3(r_l, C_{1,j}) + L_3(C_{1,j}, e) + p_{3,j} \\ &\le C_* + L_3(r_l, C_{1,j}) + p_{2,j} + p_{3,j}. \end{aligned}$$

Note that

$$L_2(r_j, r_l) + L_2(r_l, C_{1,j}) + L_3(r_l, C_{1,j}) \le p_{1,j}.$$

Summing up the inequalities above we have

$$2C_{\max} \le 2C_* + r_j + p_{1,j} + p_{2,j} + 2p_{3,j} \le 10/3C_*,$$

or

$$C_{\max}/C_* \le 5/3.$$

Case 1.2.3. M_2 and M_3 have common idle time before $C_{1,j}$ and the last common idle interval $I_1 \subseteq [0, r_j)$. Thus M_3 and M_2 have no common idle after r_j and we have

$$L_2(r_j, e) + L_3(r_j, e) \le p_{1,j} + p_{2,j}.$$

Again we assume $I_1 \subseteq [r_{l-1}, r_l)$ for some $l \le j$. From Lemma 3, we know that after time e_1, M_2 and M_3 cannot process more than one job released before r_l. Next we need to consider the following three subcases.

Case 1.2.3.1. By time r_l, machine M_2 has finished all jobs released before r_l. On M_2 and M_3, we have

$$\begin{aligned} C_{\max} &\le r_l + \sum_{k=l}^{n} p_{2,k} + L_2(r_l, r_j) + L_2(r_j, e) + p_{3,j} \\ &\le C_* + L_2(r_l, r_j) + L_2(r_j, e) + p_{3,j}, \\ C_{\max} &\le C_* + L_3(0, r_l) + L_3(r_l, r_j) + L_3(r_j, e) \\ &\le C_* + r_l + L_3(r_l, r_j) + L_3(r_j, e). \end{aligned}$$

Summing up the two inequalities above we have

$$\begin{aligned} 2C_{\max} &\le 2C_* + r_l + (L_2(r_l, r_j) + L_3(r_l, r_j)) + L_2(r_j, e) + L_3(r_j, e) + p_{3,j} \\ &\le 2C_* + r_l + (r_j - r_l) + p_{1,j} + p_{2,j} + p_{3,j} \le 3C_*, \end{aligned}$$

or

$$C_{\max}/C_* \le 3/2.$$

Case 1.2.3.2. By time r_l, machine M_3 has finished all jobs released before r_l. The proof is similar to Case 1.2.3.1.

Case 1.2.3.3. After I_1, both M_2 and M_3 have to process jobs released before r_l. According to the density, there is only one such job, say job k, with $r_k \le r_{l-1}$. Let $I_i = [b_i, e_i)$ be the other common idle intervals of M_2 and M_3 before I_1, and λ_i be the length of I_i, $i = 2, 3, \ldots, r$. We have

$$L_2(0, r_j) + L_3(0, r_j) \le r_j + \sum_{i=1}^{r} \lambda_i.$$

If $\sum_{i=1}^{r} \lambda_i \le C_*/3$, we get

$$C_{\max} \le C_* + L_2(0, r_j) + L_2(r_j, e) + p_{3,j},$$
$$C_{\max} \le C_* + L_3(0, r_j) + L_3(r_j, e),$$

and the sum becomes

$$2C_{\max} \le 2C_* + r_j + \sum_{i=1}^{r} \lambda_i + p_{3,j} \le 10/3C_*,$$

or

$$C_{\max}/C_* \le 5/3.$$

If $\sum_{i=1}^{r} \lambda_i > C_*/3$, we have

$$r_k + p_{1,k} \ge \sum_{i=1}^{r} \lambda_i > C_*/3.$$

Because $r_k + P_k \le C_*$, it follows that

$$p_{2,k} + p_{3,k} \le 2/3C_*.$$

We obtain on machines M_2 and M_3,

$$\begin{aligned} C_{\max} &\le r_l + \sum_{h=l}^{n} p_{2,h} + p_{2,k} + L_2(r_l, r_j) + L_2(r_j, e) + p_{3,j} \\ &\le C_* + p_{2,k} + L_2(r_l, r_j) + L_2(r_j, e) + p_{3,j}, \\ C_{\max} &\le r_l + \sum_{h=l}^{n} p_{3,h} + p_{3,k} + L_3(r_l, r_j) + L_3(r_j, e) \\ &\le C_* + p_{3,k} + L_3(r_l, r_j) + L_3(r_j, e), \end{aligned}$$

with the sum

$$\begin{aligned} 2C_{\max} &\le 2C_* + (r_j - r_l) + p_{2,k} + p_{3,k} + P_j \\ &\le 2C_* + r_j + P_j - \sum_{i=1}^{r} \lambda_i + p_{2,k} + p_{3,k} \\ &\le 3C_* - 1/3C_* + 2/3C_* \le 10/3C_*, \end{aligned}$$

or

$$C_{\max}/C_* \le 5/3.$$

Case 2. There are two jobs processed on machine M_3 after time e, say jobs j_1 and j_2. Without loss of generality, we assume that job j_2 is processed immediately after time e and followed by job j_1. In this case, we assume jobs j_1 and j_2 are processed on one of the two other machines right before e, say j_2 is processed on M_1 and j_1 is processed on M_2. Also assume $r_{j_1} \le r_{j_2}$ (If $r_{j_1} \ge r_{j_2}$, the proof is similar). It is easy to see that any two of the three machines have no common idle during $[r_{j_2}, e)$. Also M_1 and M_2 have no common idle during $[r_{j_1}, r_{j_2})$.

Next we need to consider the following three subcases.

Case 2.1. M_1 and M_3 have common idle in $[r_{j_1}, r_{j_2})$. Let $I_1 = [b_1, e_1)$ be the last common idle interval, then $I_1 \subseteq [r_{l-1}, r_l)$, where $r_{j_1} < r_l \le r_{j_2}$. According to the density, M_2 must process j_1 during the entire interval of $[b_1, e)$. In this case, we also have

$$L_2(r_{j_2}, e) = L_1(r_{j_2}, e) = 0.$$

On machines M_1 and M_3, we have

$$\begin{aligned} C_{\max} &\le r_l + \sum_{k=l}^{n} p_{1,k} + L_1(r_l, r_{j_2}) + p_{3,j_2} + p_{3,j_1} \\ &\le C_* + L_1(r_l, r_{j_2}) + p_{3,j_2} + p_{3,j_1}, \\ C_{\max} &= C_* + L_3(0, r_{j_1}) + L_3(r_{j_1}, e) \\ &\le C_* + r_{j_1} + L_3(r_{j_1}, e). \end{aligned}$$

Summing up the two inequalities above, together with the fact that

$$L_1(r_l, r_{j_2}) + L_3(r_{j_1}, e) \le p_{1,j_1} + p_{2,j_1},$$

we get

$$2C_{\max} \le 2C_* + r_{j_1} + p_{3,j_1} + p_{3,j_2} \le 10/3C_*,$$

or

$$C_{\max}/C_* \le 5/3.$$

Case 2.2. M_2 and M_3 are idle simultaneously in $[r_{j_1}, r_{j_2})$. Let $I_1 = [b_1, e_1)$ be the last common idle, then $I_1 \subseteq [r_{l-1}, r_l)$, where $r_{j_1} < r_l \le r_{j_2}$. After e_1 both M_3 and M_2 process at most one job released before r_l. Evidently the job is j_1 and processed during I_1 by M_1.

M_1 has no idle in $[r_{j_1}, r_l)$ and any two machines have no common idle in (r_l, e). We have

$$L_1(r_{j_1}, r_{j_2}) + L_3(r_{j_1}, r_{j_2}) \le r_{j_2} - r_{j_1}. \tag{A.1}$$

It follows from the density that

$$L_1(r_{j_2}, e) + L_3(r_{j_2}, e) \le p_{1,j_2} + p_{2,j_2}. \tag{A.2}$$

$$L_2(r_{j_1}, r_{j_2}) + L_2(r_{j_2}, e) + L_3(r_l, r_{j_2}) + L_3(r_{j_2}, e) \le p_{1,j_1} + p_{2,j_1}. \tag{A.3}$$

In addition we have:

$$\begin{aligned} C_{\max} &\le L_1(0, r_{j_1}) + L_1(r_{j_1}, r_{j_2}) + L_1(r_{j_2}, e) + p_{3,j_1} + p_{3,j_2} + C_*. \\ C_{\max} &\le L_2(0, r_{j_1}) + L_2(r_{j_1}, r_{j_2}) + L_2(r_{j_2}, e) + p_{3,j_1} + p_{3,j_2} + C_*. \\ C_{\max} &\le L_3(0, r_{j_1}) + L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e) + C_*. \\ C_{\max} &\le r_l + \sum_{k=l}^{n} p_{3,k} + L_3(r_l, r_{j_2}) + L_3(r_{j_2}, e) + p_{3,j_1} \\ &\le L_3(r_l, r_{j_2}) + L_3(r_{j_2}, e) + p_{3,j_1} + C_*. \end{aligned}$$

Summing up the four inequalities above, together with Lemma 2 and (A.1)–(A.3), we have

$$4C_{\max} \le 4C_* + r_{j_1} + p_{j_1} + r_{j_2} + p_{j_2} + 2p_{3,j_1} + p_{3,j_2} \le 7C_*.$$

Case 2.3. At most one machine idle at any time in $[r_{j_1}, r_{j_2})$. In this case we have

$$L_1(r_{j_1}, r_{j_2}) + L_2(r_{j_1}, r_{j_2}) + L_3(r_{j_1}, r_{j_2}) \le r_{j_2} - r_{j_1}.$$

Also by Lemma 3, the number of idle machines during interval (r_{j_1}, e) is at most one. We need to consider the following two subcases:

Case 2.3.1. M_2 has no idle in $[r_{j_2}, e)$. We have the following inequalities on three machines,

$$\begin{aligned} C_{\max} &\le C_* + L_1(0, r_{j_1}) + L_1(r_{j_1}, r_{j_2}) + L_1(r_{j_2}, e) + p_{3,j_1} + p_{3,j_2}, \\ C_{\max} &\le C_* + L_2(0, r_{j_1}) + L_1(r_{j_1}, r_{j_2}) + p_{3,j_1} + p_{3,j_2}, \\ C_{\max} &\le C_* + L_3(0, r_{j_1}) + L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e), \end{aligned}$$

and also by Lemma 1,

$$C_{\max} \le C_* + L_3(0, r_{j_1}) + p_{1,j_1} + p_{2,j_1}.$$

Summing up the four inequalities above, we have

$$\begin{aligned} 4C_{\max} &\le 4C_* + [L_1(0, r_{j_1}) + L_2(0, r_{j_1}) + L_3(0, r_{j_1})] + [L_1(r_{j_1}, r_{j_2}) + L_1(r_{j_1}, r_{j_2}) + L_3(r_{j_1}, r_{j_2})] \\ &\quad + L_3(0, r_{j_1}) + L_1(r_{j_2}, e) + L_3(r_{j_2}, e) + p_{3,j_1} + p_{3,j_2} + p_{3,j_1} + p_{3,j_2} + p_{1,j_1} + p_{2,j_1} \\ &\le 4C_* + r_{j_1} + p_{j_1} + r_{j_2} + p_{j_2} + r_{j_1} + p_{3,j_1} + p_{3,j_2} \\ &\le 7C_* \end{aligned}$$

or

$$C_{\max}/C_* \le 7/4.$$

Case 2.3.2. M_2 has idle interval in $[r_{j_2}, e)$. Because of the density, we have

$$L_1(r_{j_1}, e) = 0,$$

and

$$L_2(r_{j_1}, e) + L_3(r_{j_1}, e) \le p_{1,j_1} + p_{2,j_1}.$$

Consider,

$$C_{\max} \le C_* + L_1(0, r_{j_1}) + p_{3,j_1} + p_{3,j_2},$$
$$C_{\max} \le C_* + L_2(0, r_{j_1}) + L_2(r_{j_1}, r_{j_2}) + L_2(r_{j_2}, e) + p_{3,j_1} + p_{3,j_2},$$
$$C_{\max} \le C_* + L_3(0, r_{j_1}) + L_3(r_{j_1}, r_{j_2}) + L_3(r_{j_2}, e),$$

and also on machine M_3,

$$C_{\max} \le C_* + L_3(0, r_{j_2}) + L_3(r_{j_2}, e) \le C_* + r_{j_2} + p_{1,j_2} + p_{2,j_2}.$$

Summing up the four inequalities above, we get

$$4C_{\max} \le 4C_* + r_{j_1} + p_{j_1} + r_{j_2} + p_{j_2} + r_{j_1} + p_{3,j_1} + p_{3,j_2} \le 7C_*.$$

Combining all the above results, we reach the conclusion of Theorem 4. □

References

[1] B. Chen, V.A. Strusevich, Approximation algorithms for three machine open-shop scheduling, ORSA Journal on Computing 5 (1993) 321–326.
[2] D.P. Williamson, L.A. Hall, J.A. Hoogeveen, C.A.J. Hurkens, J.K. Lenstra, S.V. Sevastianov, D.B. Shmoys, Short shop schedules, Operation Research 45 (1997) 288–294.
[3] E.L. Lawer, J.K. Lenstra, A.H.G. Rinnooy Kan, Minimizing maximum lateness in a two-machine open shop [J], Math. Oper. Res. 6 (1981) 153–158; Math. Open. Res. 7 (1982) 635 (erratum).
[4] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, D.B. Shmoys, Sequencing and scheduling: Algorithms and complexity, in: S.C. Graves, et al. (Eds.), in: Handbooks in OR. & MS, vol. 4, Elsevier Science Publishers B. V., Amsterdam, 1993, pp. 445–522.
[5] I. Bárány, T. Fiala, Nearly optimum solution of multimachine scheduling problems, Szigma Mathematika Közgadasági Folyoírat 15 (1982) 177–191 (in Hungarian).
[6] J.M. Wein, Algorithms for scheduling and network problems, Ph.D. Thesis, MIT, Cambridge, MA, 1991.
[7] R. Chen, Dense schedules for open-shop with jobs release dates, OR Transactions 7 (2003) 73–77.
[8] R. Chen, W. Yu, Analysis of operation chain's properties of dense schedules for open-shop, Journal of East China University of Science and Technology 29 (2003) 522–526.
[9] T. Geonzalez, S. Sahni, Open-shop scheduling to minimize finish time, Journal of the Association for Computing Machinery 23 (1976) 665–679.
[10] V.A. Aksjonov, A polynomial-time algorithm for an approximate solution of a scheduling problem, Upravlyaemye Sistemy 28 (1988) 8–11 (in Russian).
[11] R.J. Wittrock, An adaptable scheduling algorithms for flexible flow lines, Operations Research 36 (1985) 445–412.