Accepted Manuscript

Discrete Optimization

Approximation Algorithms for the Parallel Flow Shop Problem

Xiandong Zhang, Steef van de Velde

PII: S0377-2217(11)00719-3
DOI: 10.1016/j.ejor.2011.08.007
Reference: EOR 10672

To appear in: *European Journal of Operational Research*

Received Date: 9 July 2010
Revised Date: 2 August 2011
Accepted Date: 8 August 2011

Please cite this article as: Zhang, X., van de Velde, S., Approximation Algorithms for the Parallel Flow Shop Problem, *European Journal of Operational Research* (2011), doi: 10.1016/j.ejor.2011.08.007

Approximation Algorithms for the Parallel Flow Shop Problem

Xiandong Zhang* Steef van de Velde†

August 13, 2011

Abstract

We consider the $\mathcal{NP}$-hard problem of scheduling n jobs in m two-stage parallel flow shops so as to minimize the makespan. This problem decomposes into two subproblems: assigning the jobs to parallel flow shops; and scheduling the jobs assigned to the same flow shop by use of Johnson's rule. For $m = 2$, we present a $\frac{3}{2}$-approximation algorithm, and for $m = 3$, we present a $\frac{12}{7}$-approximation algorithm. Both these algorithms run in $O(n \log n)$ time. These are the first approximation algorithms with fixed worst-case performance guarantees for the parallel flow shop problem.

Key Words: scheduling; parallel flow shop; hybrid flow shop; approximation algorithms; worst-case analysis

1 Introduction

Consider the problem of scheduling a set of n independent jobs $\mathcal{J} = \{J_1, \ldots, J_n\}$, in which each job J_j consists of a chain of two operations (O_{1j}, O_{2j}) $(j = 1, \ldots, n)$, in a *hybrid flow shop*, also called a *flexible flow shop*, so as to minimize the length of the schedule, that is, the makespan. A hybrid flow shop is an extension of the classical flow shop, where there are m_1 identical machines M_{i1} $(i = 1, \ldots, m_1)$ in stage 1 and m_2 identical machines M_{i2} $(i = 1, \ldots, m_2)$ in stage 2. The first operation O_{1j} of any job J_j needs first be processed on

*Department of Management Science, School of Management, Fudan University, Shanghai 200433, China. Tel: 86-21-25011156. E-mail: xiandongzhang@fudan.edu.cn

†Rotterdam School of Management, Erasmus University, Post Box 1738, 3000 DR Rotterdam, the Netherlands.

one of the machines in stage 1 during an uninterrupted processing time $p_{1j} \geq 0$, and then the second operation O_{2j} needs to be processed on one of the machines in stage 2 during an uninterrupted processing time $p_{2j} \geq 0$.

The hybrid flow shop problem of minimizing makespan has been well studied (Ruiz and Vazquez-Rodriguez (2010), Ribas et al. (2010) and Naderi et al. (2010)). Obviously, if $m_1 = m_2 = 1$, then the problem is polynomially solvable in $O(n \log n)$ time by Johnson's rule (Johnson (1954)). However, if $m_1 \geq 2$, or by symmetry $m_2 \geq 2$, the problem becomes strongly NP-hard (Hoogeveen et al. (1996)). Many researchers have focused on the special case with a single machine in one stage (Chen (1995), Gupta (1988), Gupta and Tunc (1991), Gupta et al. (1997)). For a review of the literature for the hybrid flow shop problem with a single machine in one stage, see Linn and Zhang (1999) and Wang (2005). For the general case, Chen (1994) and Lee and Vairaktarakis (1994) present $O(n \log n)$-time heuristics with worst-case performance guarantee ratio $2 - 1/\max\{m_1, m_2\}$. If, for any instance of the problem, the makespan of the schedule generated by some heuristic does not exceed ρ times the optimal makespan, where ρ is a constant that is as small as possible, then ρ is the worst-case performance ratio of the heuristic. A heuristic with a worst-case performance ratio of ρ is called referred to as a ρ-approximation algorithm.

A hybrid flow shop is a manufacturing system that offers much flexibility, but as Vairaktarakis and Elhafsi (2000) point out, this superior performance comes at the expense of sophisticated material handling systems, like automated guided vehicles and automated transfer lines. As an alternative to the hybrid flow shop, Vairaktarakis and Elhafsi (2000) introduced the *parallel flowline design*, which is a flexible manufacturing environment with m identical parallel two-stage flow shops $F_1, \ldots, F_m$, each consisting of a series of two machines M_{1i} and M_{2i} ($i = 1, \ldots, m$). Each job needs first to be assigned to one of the flow shops, and once assigned, it will stay there for both operations. See Figure 1 for a hybrid two-stage flow shop, where the arrows indicate the routes that the different jobs may follow, and Figure 2 for a parallel two-stage flow shop. In the remainder, we will refer to a parallel flowline design as a *parallel flow shop*.

The makespan parallel flow shop problem breaks down into two consecutive subproblems; first assigning each job to one of the m flow shops, and then scheduling the jobs in each flow shop so as to minimize the makespan. Whereas this second problem can obviously be solved in polynomial time by Johsnon's rule (Johnson (1954)), the first subproblem makes

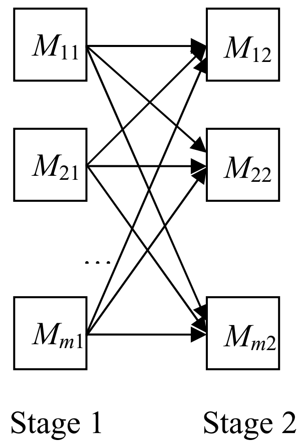

Figure 1: A hybrid two-stage flow shop.

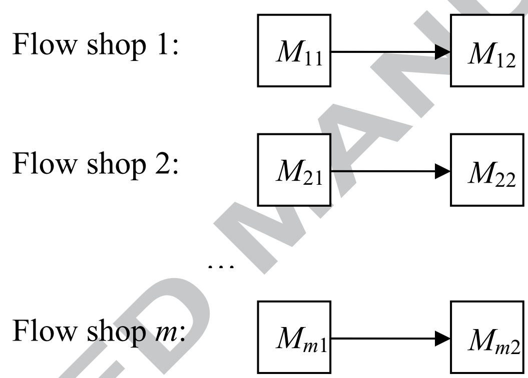

Figure 2: A parallel two-stage flow shop.

the problem $\mathcal{NP}$-hard, as proved by Vairaktarakis and Elhafsi (2000), who also presented an $O(n\sum_{j=1}^{n}(p_{1j}+p_{2j})^3)$ time dynamic programming algorithm for its solution. Qi (2008) gave a faster algorithm, running in $O(n\sum_{j=1}^{n}(p_{1j}+p_{2j})^2)$ time.

Vairaktarakis and Elhafsi (2000) concluded empirically, on the basis of computational experiments with several heuristics for both problems, that the parallel flow shop entails only a minor loss in throughput performance in comparison with the hybrid flow shop; accordingly, it is an attractive alternative to the hybrid flow shop, with its complicated routings. Other heuristics for the parallel flow shop problem have been presented by Cao and Chen (2003) and Al-Salem (2004).

In contrast to the makespan hybrid flow shop problem, no approximation results for the makespan parallel flow shop are known. In this paper, we present a $\frac{3}{2}$-approximation algorithm for the parallel flow shop problem with $m = 2$ in Section 2. For $m = 3$, we present a $\frac{12}{7}$-approximation algorithm in Section 3. These results are the first polynomial-time algorithms with fixed worst-case ratios for the parallel flow shop problem.

Section 4 ends the paper with some conclusions, where we point out that our algorithms and their worst-case performance guarantees also apply to the parallel flow shop problem where each job J_j after the completion of its first operation may be transferred to another flow shop for the processing of its second operation and where such a transfer requires a transportation time $\tau_j \geq 0$. This transportation time effectively introduces a minimum time lag between the completion time of the first operation and the start time of the second operation of a job. Note that if $\tau_j = 0$ for each J_j, then the parallel flow shop problem with transportation times boils down to the hybrid flow shop problem. For the hybrid flow shop problem with $m_1 = m_2 = 2$, our approximation algorithm has the same worst-case performance ratio as the one by Chen (1994) and Lee and Vairaktarakis (1994). At the other extreme, if $\tau_j = \infty$ for each J_j, then transfer between flow shops is effectively prohibited, and we have the original parallel flow shop problem.

2 A $\frac{3}{2}$-approximation algorithm for $m = 2$

In the remainder of the paper, we assume that the job set $\mathcal{J} = \{J_1, \ldots, J_n\}$ has been re-indexed according to Johnson's rule; that is, for any pair of jobs (J_i, J_j) we have that $i < j$ if and only if

$$\min\{p_{1i}, p_{2j}\} \leq \min\{p_{1j}, p_{2i}\}.$$

For any instance of the m parallel two-stage flow shop problem, we refer to the Johnsonian schedule σ as the schedule that is obtained by assigning all the jobs to the first flow shop F_1 and processing them in order of Johnson's rule. $C_{\max}(\mathcal{J})$ denotes the makespan of the Johnsonian schedule for any job set $\mathcal{J} = \{J_1, \ldots, J_n\}$, whereas S_{ij} and C_{ij} denote the start and completion times of the operations O_{ij} in the Johnsonian schedule, respectively, for $i = 1, 2; j = 1, \ldots, n$.

Lemma 1, which goes with no proof, specifies a simple lower bound on the minimum makespan $C^*_{\max}$ for the m parallel two-stage flow shop problem.

Lemma 1 *We have that*

$$C^*_{\max} \geq \max\{\frac{1}{m}\sum_{j=1}^{n} p_{1j}, \frac{1}{m}\sum_{j=1}^{n} p_{2j}, \frac{1}{m}C_{\max}(\mathcal{J}), \max_{1\leq j\leq n}\{p_{1j}+p_{2j}\}\}. \quad (1)$$

Roughly speaking, the core idea for the $\frac{3}{2}$-algorithm is to judiciously cut a Johnsonian schedule σ for $\mathcal{J}$ into two parts. The first part is scheduled on F_1, the second part on F_2. Both parts are scheduled according to Johnson's rule in order to minimize the makespan. The key question of course is where to cut the schedule so as to guarantee the $\frac{3}{2}$ performance ratio.

Let now $T_1 = \frac{1}{4}C_{\max}(\mathcal{J})$ and $T_2 = \frac{3}{4}C_{\max}(\mathcal{J})$. Initially, we try to cut the Johnsonian schedule σ at time T_2. We have then the following lemma.

Lemma 2 *If there exists no job J_h with $S_{2h} \leq T_2 \leq C_{2h}$, then let $\mathcal{J}^1 = \{J_1, \ldots, J_{k-1}\}$ and $\mathcal{J}^2 = \{J_k, \ldots, J_n\}$ with J_k such that $S_{1k} \leq T_2 \leq C_{1k}$. We then have that*

$$\max\{C_{\max}(\mathcal{J}^1), C_{\max}(\mathcal{J}^2)\} \leq \frac{3}{2}C^*_{\max}.$$

Proof. See Figure 3 for an illustration of how the two job sets are formed if there is no job J_h such that $S_{2h} \leq T_2 \leq C_{2h}$. By visual inspection of Figure 3 and by use of (1), it follows that

$$C_{\max}(\mathcal{J}^1) \leq T_2 = \frac{3}{4}C_{\max}(\mathcal{J}) \leq \frac{3}{2}C^*_{\max}, \text{ and}$$

$$C_{\max}(\mathcal{J}^2) \leq C_{\max}(\mathcal{J}) - T_2 + p_{1k} \leq \frac{1}{4}C_{\max}(\mathcal{J}) + p_{1k} \leq \frac{3}{2}C^*_{\max}.$$ □

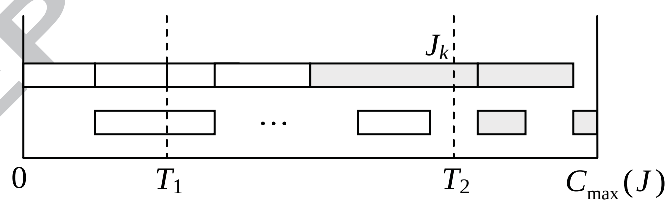

Figure 3: Cutting the Johnsonian schedule as prescribed in Lemma 2.

The implication of Lemma 1 is that if there is no job J_h with $S_{2h} \leq T_2 \leq C_{2h}$, then we have indeed constructed a schedule with makespan no more than $\frac{3}{2}$ times the optimal makespan and we are done. Accordingly, we need to investigate the case where such a job J_h does exist. We then have the following result.

Lemma 3 *If there exists a job J_h with $S_{2h} \le T_2 \le C_{2h}$ and if $S_{1h} \ge T_1$ or $C_{1h} = S_{2h}$, then let $\mathcal{J}^1 = \{J_1, \dots, J_{h-1}\}$ and $\mathcal{J}^2 = \{J_h, \dots, J_n\}$. It then holds that*

$$\max\{C_{\max}(\mathcal{J}^1), C_{\max}(\mathcal{J}^2)\} \le \frac{3}{2} C^*_{\max}.$$

Proof. Refer to Figure 4 for an illustration. Since $S_{2h} \le T_2$, job J_{h-1} is finished before or at T_2. We have therefore that

$$C_{\max}(\mathcal{J}^1) \le T_2 \le \frac{3}{2} C^*_{\max}.$$

If $S_{1h} \ge T_1$, we have that

$$C_{\max}(\mathcal{J}^2) \le C_{\max}(\mathcal{J}) - T_1 = T_2 \le \frac{3}{2} C^*_{\max}.$$

If $C_{1h} = S_{2h}$, then

$$C_{\max}(\mathcal{J}^2) \le p_{1h} + p_{2h} + (C_{\max}(\mathcal{J}) - T_2) \le \frac{3}{2} C^*_{\max}.$$

□

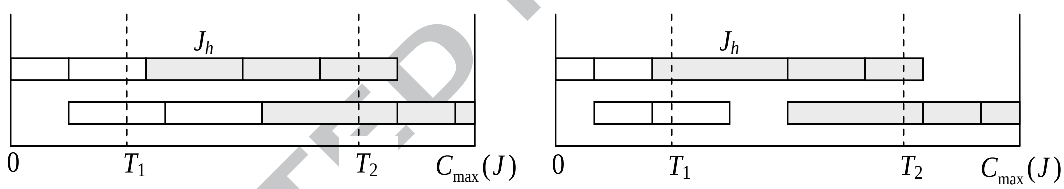

Figure 4: Cutting the Johnsonian schedule as prescribed in Lemma 3.

Lemmata 2 and 3 do not cover the case where there exists a job J_h with $S_{2h} \le T_2 \le C_{2h}$, $S_{1h} < T_1$ and $C_{1h} < S_{2h}$. To analyze this case, we transform the Johnsonian schedule σ into the schedule σ' by delaying all operations as much as possible without changing the makespan. Hence, σ' has makespan $C_{\max}(\mathcal{J})$, has no idle time between any two operations on machine M_2, and all jobs are sequenced in order of Johnson's rule. We refer to σ' as the *delayed* Johnsonian schedule. Let now S'_{ij} and C'_{ij} denote the start and completion times of O_{ij} in σ'.

For σ', we have the following result.

Lemma 4 *If $S'_{1h} \geq T_1$ or $C'_{1h} = S'_{2h}$, then let $\mathcal{J}^1 = \{J_1, \ldots, J_{h-1}\}$, $\mathcal{J}^2 = \{J_h, \ldots, J_n\}$. It then holds that*

$$\max\{C_{\max}(\mathcal{J}^1), C_{\max}(\mathcal{J}^2)\} \leq \frac{3}{2}C^*_{\max}.$$

Proof. In this case, there is a job J_h with $S_{2h} \leq T_2 \leq C_{2h}$, therefore we have

$$C_{\max}(\mathcal{J}^1) = C_{2(h-1)} \leq S_{2h} \leq T_2 \leq \frac{3}{2}C^*_{\max}.$$

If $S'_{1h} \geq T_1 = \frac{1}{4}C_{\max}(\mathcal{J})$, then

$$C_{\max}(\mathcal{J}^2) \leq C_{\max}(\mathcal{J}) - S'_{1h} \leq \frac{3}{4}C_{\max}(\mathcal{J}) = \frac{3}{2}C^*_{\max}.$$

This case is illustrated in Figure 5, which shows both σ and σ'.

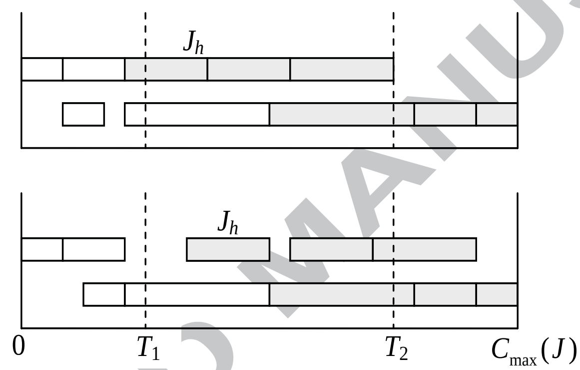

Figure 5: Cutting the delayed Johnsonian schedule as prescribed in Lemma 4 if $S'_{1h} \geq T_1$. The top schedule is the Johnsonian schedule σ, the bottom schedule is the delayed Johnsonian schedule σ'.

If $S'_{1h} < T_1$ and we have $C'_{1h} = S'_{2h}$, then

$$C_{\max}(\mathcal{J}^2) \leq p_{1h} + p_{2h} + (C_{\max}(\mathcal{J}) - C_{2h}) \leq C^*_{\max} + \frac{1}{4}C_{\max}(\mathcal{J}) = \frac{3}{2}C^*_{\max}.$$

This case is illustrated by Figure 6. □

We have dealt now with many different subcases. The only case left to consider is the one with a job J_h with $S_{2h} \leq T_2 \leq C_{2h}$, $S_{1h} < T_1$, $C_{1h} < S_{2h}$, $S'_{1h} < T_1$ and $C'_{1h} < S'_{2h}$. See Figure 7 for an illustration of this case. In what follows, we will focus on this case.

We then have the following lemma.

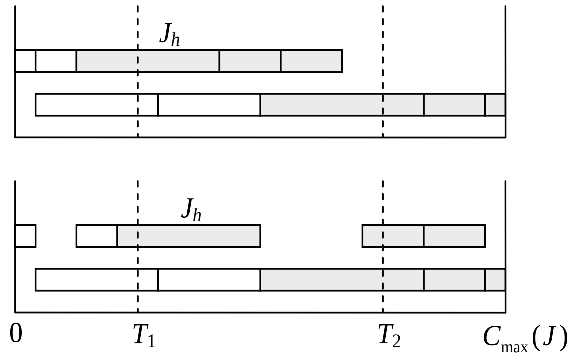

Figure 6: Cutting the delayed Johnsonian schedule as prescribed in Lemma 4 if $S'_{1h} < T_1$. The top schedule is the Johnsonian schedule σ, the bottom schedule is delayed Johnsonian schedule σ'.

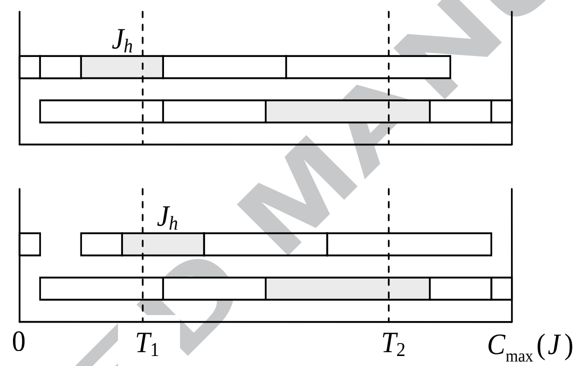

Figure 7: Illustration of a Johnsonian schedule σ (the top schedule) and a delayed Johnsonian schedule σ' (the bottom schedule) for a job J_h with $S_{2h} \leq T_2 \leq C_{2h}$, $S_{1h} < T_1$, $C_{1h} < S_{2h}$, $S'_{1h} < T_1$ and $C'_{1h} < S'_{2h}$.

Lemma 5 *If there is a job J_h with $S_{2h} \leq T_2 \leq C_{2h}$, $S_{1h} < T_1$, $C_{1h} < S_{2h}$, $S'_{1h} < T_1$ and $C'_{1h} < S'_{2h}$, then machine M_2 is completely busy during the period $[T_1, T_2]$ in schedule σ and machine M_1 is completely busy during the period $[T_1, T_2]$ in schedule σ'.*

Proof. If in schedule σ machine M_2 would not have been busy during the interval $[T_1, T_2]$, then operation O_{2h} could have been started earlier. Similarly, if M_1 would not have been busy during the interval $[T_1, T_2]$ in schedule σ', then operation O_{1h} could have been started later. □

We now separate all n jobs into two subsets $\mathcal{S}^1$ and $\mathcal{S}^2$ with $\mathcal{S}^1 = \{J_j | p_{1j} \leq p_{2j}, j = 1, \ldots, n\}$ and $\mathcal{S}^2 = \{J_j | p_{1j} > p_{2j}, j = 1, \ldots, n\}$. Since all jobs have been indexed in order of Johnson's rule, we can represent these two sets alternatively as $\mathcal{S}^1 = \{J_1, \ldots, J_u\}$ and $\mathcal{S}^2 = \{J_v, \ldots, J_n\}$ with $v = u + 1$. We branch into two cases: $\sum_{j=v}^{n} p_{1j} \geq T_1$; and $\sum_{j=1}^{u} p_{2j} \geq T_1$. Since these two cases are symmetrical, we analyze only the case with $\sum_{j=v}^{n} p_{1j} \geq T_1$.

In this case, we need to find a job J_e with $e \geq v$ such that $\sum_{j=v}^{e-1} p_{1j} < T_1 \leq \sum_{j=v}^{e} p_{1j}$ and a job J_d with $d < v$ such that $\sum_{j=d+1}^{e-1} p_{2j} < T_1 \leq \sum_{j=d}^{e-1} p_{2j}$. If $v = e$, we let $\sum_{j=v}^{e-1} p_{1j} = 0$. If $d = e - 1$, we let $\sum_{j=d+1}^{e-1} p_{2j} = 0$.

Lemma 6 *J_e and J_d exist.*

PROOF. Since $\sum_{j=v}^{n} p_{1j} \geq T_1$, job J_e must exist. To show that J_d exists, too, we branch into two cases. Since machine M_2 is busy in the period $[T_1, T_2]$ and $S_{1h} \leq T_2 \leq C_{2h}$, we have $\sum_{j=1}^{h} p_{2j} \geq T_2 - T_1 > T_1$. If $J_h \in \mathcal{S}^1$, then $v > h$, and we have that $\sum_{j=1}^{v-1} p_{2j} \geq \sum_{j=1}^{h} p_{2j} > T_1$. Hence, job J_d exists. If $J_h \in \mathcal{S}^2$, then $v \leq h$. And since $\sum_{j=v}^{e} p_{1j} \geq T_1$ and $\sum_{j=1}^{h-1} p_{1j} < T_1$ (because $S_{1h} < T_1$), we have that $e \geq h$. Since $C_{1h} < S_{2h}$, we have $\sum_{j=1}^{h-1} p_{2j} > p_{1h} > p_{2h}$. Together with $\sum_{j=1}^{h} p_{2j} \geq T_2 - T_1 = 2T_1$, we get $\sum_{j=1}^{h-1} p_{2j} > T_1$. Therefore, job J_d exists in this case also. For an illustration, see Figure 8. □

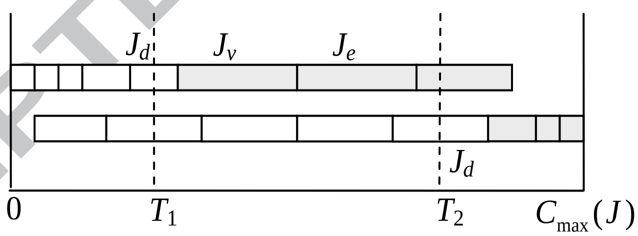

Figure 8: Illustration of the jobs J_u,J_v,J_d,J_e, with $J_u = J_d = J_h$, as they occur in Lemma 6.

We now divide the case $\sum_{j=v}^{n} p_{1j} \geq T_1$ further into 5 different subcases and deal with these subcases in Lemmata 7 to 11.

Lemma 7 *If $\sum_{j=v}^{e} p_{2j} \geq T_1$, let $\mathcal{J}^1 = \{J_v, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. Then*

$$\max\{C_{\max}(\mathcal{J}^1), C_{\max}(\mathcal{J}^2)\} \leq \frac{3}{2} C^*_{\max}.$$

PROOF. In this case, we have $\sum_{j=v}^{e-1} p_{1j} < T_1 \leq \sum_{j=v}^{e} p_{1j}$, $\sum_{j=v}^{e} p_{2j} \geq T_1$, $\mathcal{J}^1 = \{J_v, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. This can be illustrated by Figure 9.

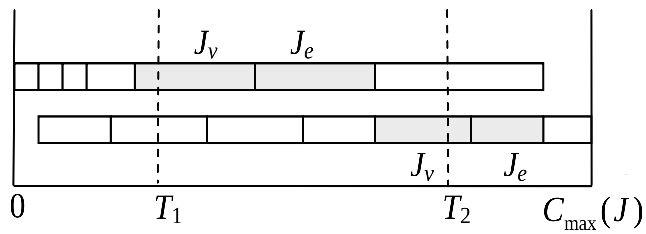

Figure 9: Cutting the Johnsonian schedule as prescribed in Lemma 7.

Let J_w ($v \leq w \leq e$) be the job for which $C_{\max}(\mathcal{J}^1) = \sum_{j=v}^{w} p_{1j} + \sum_{j=w}^{e} p_{2j}$. This implies that

$$\sum_{j=v}^{w} p_{1j} + \sum_{j=w}^{e} p_{2j} = \max_k\{\sum_{j=v}^{k} p_{1j} + \sum_{j=k}^{e} p_{2j}\},$$

and we refer to J_w as the *critical job* of schedule σ. Since $\mathcal{J}^1 \subset \mathcal{S}^2 = \{J_j | p_{1j} > p_{2j}\}$, we must have that $p_{2e} \leq p_{2w} < p_{1w}$ and $\sum_{j=v}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} \leq \sum_{j=v}^{w-1} p_{1j} + \sum_{j=w}^{e-1} p_{2j} < \sum_{j=v}^{e-1} p_{1j} < T_1$. It then holds that

$$C_{\max}(\mathcal{J}^1) = \sum_{j=v}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} + p_{1w} + p_{2w} < T_1 + C^*_{\max} \leq \frac{3}{2} C^*_{\max}.$$

Let σ^2 be the minimum makespan schedule for the jobs in $\mathcal{J}^2$, obtained by scheduling the jobs in order of Johnson's rule. For σ^2, let S''_{ij} denote the start time and C''_{ij} the completion time of operation O_{ij} ($i = 1, 2; j = 1, \ldots, v-1, e+1, \ldots, n$). We have $S''_{ij} = S_{ij}$, $C''_{ij} = C_{ij}$, for $j = 1, \ldots, u$; and $S''_{ij} \leq S_{ij} - T_1$, $C''_{ij} \leq C_{ij} - T_1$, for $j = e+1, \ldots, n$, since job set $\mathcal{J}^1 = \{J_v, \ldots, J_e\}$ is not included in $\mathcal{J}^2$ and $\sum_{j=v}^{e} p_{1j} \geq \sum_{j=v}^{e} p_{2j} \geq T_1$. We have

$$C_{\max}(\mathcal{J}^2) = C''_{2n} \leq C_{\max}(\mathcal{J}) - T_1 = \frac{3}{2} C^*_{\max}.$$

□

Lemma 8 *If $\sum_{j=d}^{v-1} p_{1j} \geq T_1$, then let $\mathcal{J}^1 = \{J_d, \ldots, J_{v-1}\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. We then have that*

$$\max\{C_{\max}(\mathcal{J}^1), C_{\max}(\mathcal{J}^2)\} \leq \frac{3}{2} C^*_{\max}.$$

Proof. This case is illustrated in Figure 10.

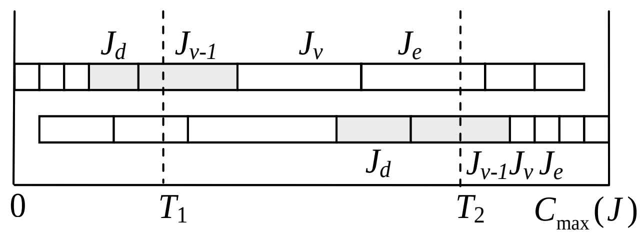

Figure 10: Cutting the Johnsonian schedule as prescribed in Lemma 8.

Since $p_{1j} \leq p_{2j}$ for $j = d, \ldots, v-1$, we have $\sum_{j=d}^{v-1} p_{2j} \geq \sum_{j=d}^{v-1} p_{1j} \geq T_1$. By definition of job J_d, we get $\sum_{j=d+1}^{v-1} p_{2j} < T_1$. The case is then symmetric to the case specified in Lemma 7. □

In the remaining analysis, we therefore assume that $\sum_{j=d}^{v-1} p_{1j} < T_1$.

Lemma 9 *Assume $\sum_{j=d}^{v} p_{1j} \geq T_1$ and $\sum_{j=d}^{v} p_{2j} \geq T_1$. If $v < e$, then let $\mathcal{J}^1 = \{J_d, \ldots, J_v\}$ and $\mathcal{J}^2 = \{J_1, \ldots, J_{d-1}, J_{v+1}, \ldots, J_n\}$. If $v = e$, find a job J_k with $\sum_{j=k+1}^{e} p_{2j} < T_1 \leq \sum_{j=k}^{e} p_{2j}$ and $d \leq k < e$, and let $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. It then holds that*

$$\max\{C_{\max}(\mathcal{J}^1), C_{\max}(\mathcal{J}^2)\} \leq \frac{3}{2} C^*_{\max}.$$

Proof. First consider the case $v < e$, illustrated by Figure 11.

If $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{v} p_{1j} + p_{2v} = \sum_{j=d}^{v-1} p_{1j} + p_{1v} + p_{2v}$, we have $C_{\max}(\mathcal{J}^1) < T_1 + C^*_{\max} < \frac{3}{2} C^*_{\max}$. If $C_{\max}(\mathcal{J}^1) = p_{1d} + \sum_{j=d}^{v} p_{2j} = p_{1d} + p_{2d} + \sum_{j=d+1}^{v} p_{2j}$, we have $C_{\max}(\mathcal{J}^1) < C^*_{\max} + T_1 \leq \frac{3}{2} C^*_{\max}$. If $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{w} p_{1j} + \sum_{j=w}^{v} p_{2j}$ and $d < w < v$, where J_w is the critical job, we have $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{w} p_{1j} + \sum_{j=w}^{v} p_{2j} < T_1 + T_1 \leq C^*_{\max}$, since $\sum_{j=d}^{v-1} p_{1j} < T_1$ and $\sum_{j=d+1}^{v} p_{2j} < T_1$. The proof that $C_{\max}(\mathcal{J}^2) \leq \frac{3}{2} C^*_{\max}$ is similar to the proof of Lemma 7.

Now consider the case $v = e$, which is illustrated by Figure 12.

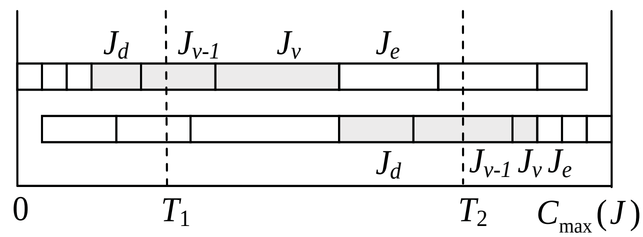

Figure 11: Cutting the Johnsonian schedule as prescribed in Lemma 9 if $v < e$.

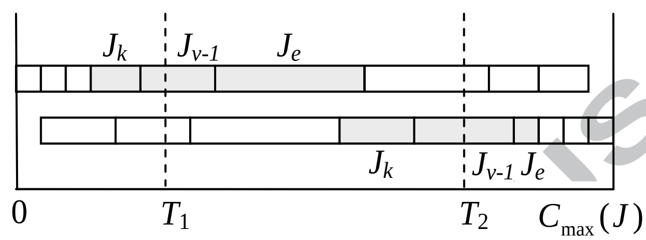

Figure 12: Splitting of the Johnsonian schedule according to Lemma 9. ($v \geq e$)

Since $\sum_{j=d}^{e-1} p_{2j} \geq T_1$, job J_k exists. In this case, we have $\sum_{j=k}^{e-1} p_{1j} < T_1$, which follows from $\sum_{j=d}^{v-1} p_{1j} < T_1$ and $d \leq k < v = e$. Therefore, the proof is analogous to the one for $v < e$. □

In Lemma 9, we consider only the situation that $\sum_{j=d}^{v} p_{1j} \geq T_1$ and $\sum_{j=d}^{v} p_{2j} \geq T_1$. If $\sum_{j=d}^{v} p_{1j} \geq T_1$ and $\sum_{j=d}^{v} p_{2j} < T_1$, it must be that $v \leq e-2$. Otherwise, if $v = e$ or $v = e-1$, we would have that $\sum_{j=d}^{v} p_{2j} \geq T_1$. If the subcase in Lemma 9 is not satisfied, we have Lemmata 10 and 11 to solve remaining cases.

Lemma 10 *If $\sum_{j=d}^{e-1} p_{1j} \geq T_1$, let $\mathcal{J}^1 = \{J_d, \ldots, J_{e-1}\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. It then holds that*

$$\max\{C_{\max}(\mathcal{J}^1), C_{\max}(\mathcal{J}^2)\} \leq \frac{3}{2} C^*_{\max}.$$

Proof. If $v = e$ or $v = e-1$, the result is correct due to Lemma 8 and Lemma 9. Hence, we need to consider only the case $v \leq e-2$, which is illustrated by Figure 13.

Consider $C_{\max}(\mathcal{J}^1)$. Let J_w be the critical job in the minimum makespan schedule for $\mathcal{J}^1$. If $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{w} p_{1j} + \sum_{j=w}^{e-1} p_{2j}$ and $d \leq w < v$, we must have $p_{1d} \leq p_{1w} \leq p_{2w}$ and

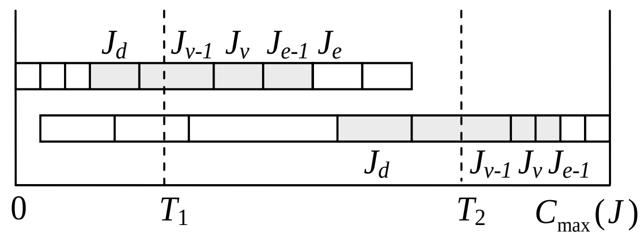

Figure 13: Cutting the Johnsonian schedule as prescribed in Lemma 10.

$\sum_{j=d}^{w-1} p_{1j} + \sum_{j=w+1}^{e-1} p_{2j} \leq \sum_{j=d+1}^{e-1} p_{2j} < T_1$. Then, $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{w-1} p_{1j} + \sum_{j=w+1}^{e-1} p_{2j} + p_{1w} + p_{2w} < T_1 + C^*_{\max} = \frac{3}{2}C^*_{\max}$.

If $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{w} p_{1j} + \sum_{j=w}^{e-1} p_{2j}$ and $v \leq w \leq e-1$, we have $\sum_{j=w+1}^{e-1} p_{2j} - \sum_{j=w+1}^{e-1} p_{1j} \leq 0$, since $\{J_w, \ldots, J_{e-1}\} \subset \mathcal{S}^2$. This implies that

$$\begin{aligned} C_{\max}(\mathcal{J}^1) &= \sum_{j=d}^{w} p_{1j} + \sum_{j=w}^{e-1} p_{2j} \\ &= \sum_{j=d}^{v-1} p_{1j} + \sum_{j=v}^{e-1} p_{1j} + p_{2w} + \sum_{j=w+1}^{e-1} p_{2j} - \sum_{j=w+1}^{e-1} p_{1j} \\ &\leq \sum_{j=d}^{v-1} p_{1j} + \sum_{j=v}^{e-1} p_{1j} + p_{2w}. \end{aligned}$$

If $\sum_{j=d}^{v-1} p_{1j} + p_{2w} \geq T_1$, we have $\sum_{j=d}^{v} p_{1j} \geq T_1$ and $\sum_{j=d}^{v} p_{2j} \geq T_1$, since $p_{2w} \leq p_{2v} < p_{1v}$ and $\sum_{j=d}^{v-1} p_{1j} \leq \sum_{j=d}^{v-1} p_{2j}$. We have solved this case in Lemma 9. If $\sum_{j=d}^{v-1} p_{1j} + p_{2w} < T_1$, we have that

$$C_{\max}(\mathcal{J}^1) \leq \sum_{j=d}^{v-1} p_{1j} + p_{2w} + \sum_{j=v}^{e-1} p_{1j} < T_1 + T_1 < C^*_{\max}.$$

Since we have $\sum_{j=d}^{e-1} p_{1j} \geq T_1$ and $\sum_{j=d}^{e-1} p_{2j} \geq T_1$ by definition, the proof of set $\mathcal{J}^2$ is analogous to that of Lemma 7. □

Lemma 11 *If $\sum_{j=d}^{e-1} p_{1j} < T_1$, find a job J_k with $d \leq k < v$ such that $\sum_{k+1}^{e} p_{2j} < T_1 \leq \sum_{k}^{e} p_{2j}$, and define $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. It then holds that*

$$\max\{C_{\max}(\mathcal{J}^1), C_{\max}(\mathcal{J}^2)\} \leq \frac{3}{2}C^*_{\max}.$$

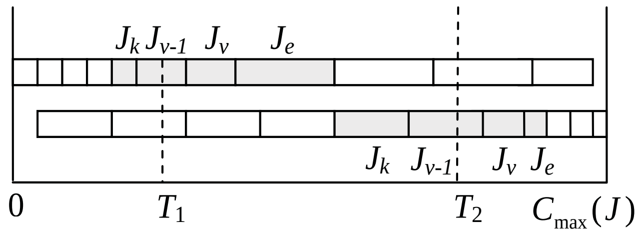

Figure 14: Cutting the Johnsonian schedule as indicated in Lemma 11.

Proof. For a visualization of this case, see Figure 14.

Since $\sum_{j=d}^{e-1} p_{2j} \geq T_1$, job J_k exists. If $C_{\max}(\mathcal{J}^1) = \sum_{j=k}^{w} p_{1j} + \sum_{j=w}^{e} p_{2j}$ and $k \leq w < v$, we must have $p_{1k} \leq p_{1w} \leq p_{2w}$ and $\sum_{j=k}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} \leq \sum_{j=k+1}^{e} p_{2j} < T_1$. Then, $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} + p_{1w} + p_{2w} < T_1 + C^*_{\max} = \frac{3}{2}C^*_{\max}$.

If $C_{\max}(\mathcal{J}^1) = \sum_{j=k}^{w} p_{1j} + \sum_{j=w}^{e} p_{2j}$ and $v \leq w \leq e$, we must have $p_{2e} \leq p_{2w} < p_{1w}$ and $\sum_{j=k}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} \leq \sum_{j=k}^{e-1} p_{1j} \leq \sum_{j=d}^{e-1} p_{1j} < T_1$. Then, $C_{\max}(\mathcal{J}^1) = \sum_{j=k}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} + p_{1w} + p_{2w} < T_1 + C^*_{\max} = \frac{3}{2}C^*_{\max}$.

Since we have $\sum_{j=k}^{e} p_{1j} \geq \sum_{j=v}^{e} p_{1j} \geq T_1$ and $\sum_{j=k}^{e} p_{2j} \geq T_1$, the proof of set $\mathcal{J}^2$ is analogous to that of Lemma 7. □

We are now done with the analysis of the case for which $\sum_{j=v}^{n} p_{1j} \geq T_1$, and for which there exists a job J_h with $S_{2h} \leq T_2 \leq C_{2h}$, $S_{1h} < T_1$, $C_{1h} < S_{2h}$, $S'_{1h} < T_1$ and $C'_{1h} < S'_{2h}$. If $\sum_{j=1}^{u} p_{2j} \geq T_1$, the case is symmetrical to the case $\sum_{j=v}^{n} p_{1j} \geq T_1$, and we can cut the Johnsonian schedule similarly.

Lemma 12 *There is no case with both* $\sum_{j=v}^{n} p_{1j} < T_1$ *and* $\sum_{j=1}^{u} p_{2j} < T_1$.

Proof. If $\sum_{j=v}^{n} p_{1j} < T_1$ and $\sum_{j=1}^{u} p_{2j} < T_1$, we get $\sum_{j=v}^{n} p_{2j} < T_1$ and $\sum_{j=1}^{u} p_{1j} < T_1$. Then we must have that $\sum_{j=v}^{n} p_{1j} + \sum_{j=1}^{u} p_{2j} + \sum_{j=v}^{n} p_{2j} + \sum_{j=1}^{u} p_{1j} < C_{\max}(\mathcal{J})$, which is a contradiction. □

Using Lemmata 2-12, we have proved that we can split any set $\mathcal{J}$ into two disjoint subsets $\mathcal{J}^1$ and $\mathcal{J}^2$ and guarantee that the minimum makespan schedule for either subset has makespan no larger than $\frac{3}{2}C^*_{\max}$. The full details of the algorithm, referred to as Algorithm $SPLT1$, can be found as following.

Algorithm 1 *SPLT1*

Step 1. (Initialization) Re-index the job set $\mathcal{J}$ according to the Johnson's rule.

Let $S_{11} = 0$, $C_{11} = S_{11} + p_{11}$, $S_{21} = C_{11}$, $C_{21} = S_{21} + p_{21}$.

For $j = 2$ to n, do the following:

$S_{1j} = C_{1(j-1)}$, $C_{1j} = S_{1j} + p_{1j}$, $S_{2j} = \max\{C_{1j}, C_{2(j-1)}\}$, $C_{2j} = S_{2j} + p_{2j}$.

Let $C_{\max}(\mathcal{J}) = C_{2n}$, $T_1 = \frac{1}{4}C_{\max}(\mathcal{J})$, $T_2 = \frac{3}{4}C_{\max}(\mathcal{J})$.

Step 2. Find the job J_h with $S_{2h} \leq T_2 \leq C_{2h}$. If job J_h does not exists, find the job J_k with $S_{1k} \leq T_2 \leq C_{1k}$, and let $\mathcal{J}^1 = \{J_1, \ldots, J_{k-1}\}$, and $\mathcal{J}^2 = \{J_k, \ldots, J_n\}$, stop; otherwise, go to Step 3 with J_h.

*Step 3.*If $S_{1h} \geq T_1$ or $C_{1h} = S_{2h}$, let $\mathcal{J}^1 = \{J_1, \ldots, J_{h-1}\}$, and $\mathcal{J}^2 = \{J_h, \ldots, J_n\}$, stop; otherwise, go to Step 4 with J_h.

Step 4. Let $C'_{1n} = S_{2n}$ and $S'_{1n} = C'_{1n} - p_{1n}$.

For $j = (n-1)$ to 1, perform the following computations:

$C'_{1j} = \min\{S'_{1(j+1)}, S_{2j}\}$ and $S'_{1j} = C'_{1j} - p_{1j}$, where S'_{1j} and C'_{1j} are the latest possible start and completion time of job J_j in machine M_1.

Step 5. If $S'_{1h} \geq T_1$ or $C'_{1h} = S'_{2h}$, let $\mathcal{J}^1 = \{J_1, \ldots, J_{h-1}\}$, $\mathcal{J}^2 = \{J_h, \ldots, J_n\}$, and stop; otherwise, go to Step 6.

Step 6. In schedule σ, find the job J_u with $p_{1u} \leq p_{2u}$ and $p_{1(u+1)} > p_{2(u+1)}$, and let $v = u + 1$. Therefore, in schedule σ, we have $p_{1j} \leq p_{2j}$ for $j = 1, \ldots, u$ and $p_{1j} > p_{2j}$ for $j = v, \ldots, n$. Then, we branch into the two cases.

Case 1. $\sum_{j=v}^{n} p_{1j} \geq T_1$. Find a job J_e with $e \geq v$ such that $\sum_{j=v}^{e-1} p_{1j} < T_1 \leq \sum_{j=v}^{e} p_{1j}$ and a job J_d with $d < v$ such that $\sum_{j=d+1}^{e-1} p_{2j} < T_1 \leq \sum_{j=d}^{e-1} p_{2j}$. We branch into five subcases.

Subcase 1.1 $\sum_{j=v}^{e} p_{2j} \geq T_1$. Let $\mathcal{J}^1 = \{J_v, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. Stop.

Subcase 1.2 $\sum_{j=d}^{v-1} p_{1j} \geq T_1$. Let $\mathcal{J}^1 = \{J_d, \ldots, J_{v-1}\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. Stop.

Subcase 1.3 $\sum_{j=d}^{v} p_{1j} \geq T_1$ and $\sum_{j=d}^{v} p_{2j} \geq T_1$. If $v < e$, let $\mathcal{J}^1 = \{J_d, \ldots, J_v\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. If $v = e$, find a job J_k with $\sum_{j=k+1}^{e} p_{2j} < T_1 \leq \sum_{j=k}^{e} p_{2j}$ and $d \leq k < e$. Let $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. Stop.

Subcase 1.4 $\sum_{j=d}^{e-1} p_{1j} \geq T_1$. Let $\mathcal{J}^1 = \{J_d, \ldots, J_{e-1}\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. Stop.

Subcase 1.5 $\sum_{j=d}^{e-1} p_{1j} < T_1$. Find a job J_k with $d \leq k < v$ such that $\sum_{k+1}^{e} p_{2j} < T_1 \leq \sum_{k}^{e} p_{2j}$, $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. Stop.

Case 2. $\sum_{j=1}^{u} p_{2j} \geq T_1$. Find a job J_d with $d \leq u$ such that $\sum_{j=d+1}^{u} p_{2j} < T_1 \leq \sum_{j=d}^{u} p_{2j}$ and a job J_e with $e > u$ such that $\sum_{j=d+1}^{e-1} p_{1j} < T_1 \leq \sum_{j=d+1}^{e} p_{1j}$. We branch into five subcases.

Subcase 2.1 $\sum_{j=d}^{u} p_{1j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_d, \ldots, J_u\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.2 $\sum_{j=u+1}^{e} p_{2j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_{u+1}, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.3 $\sum_{j=u}^{e} p_{1j} \geq T_1$ *and* $\sum_{j=u}^{e} p_{2j} \geq T_1$. *If* $d < u$, *let* $\mathcal{J}^1 = \{J_u, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *If* $d = u$, *find a job* J_k *with* $\sum_{j=d}^{k-1} p_{1j} < T_1 \leq \sum_{j=d}^{k} p_{1j}$ *and* $d < k \leq e$. *Let* $\mathcal{J}^1 = \{J_d, \ldots, J_k\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.4 $\sum_{j=d+1}^{e} p_{2j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_{d+1}, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.5 $\sum_{j=d+1}^{e} p_{2j} < T_1$. *Find a job* J_k *with* $u < k \leq e$ *such that* $\sum_{d}^{k-1} p_{2j} < T_1 \leq \sum_{d}^{k} p_{2j}$, $\mathcal{J}^1 = \{J_d, \ldots, J_k\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Theorem 1 *Algorithm* $SPLT1$ *is a* $\frac{3}{2}$*-approximation for minimizing makespan on two parallel two-stage flow shops.* □

In Step 1 of the algorithm $SPLT1$, the re-indexing process runs in $O(n \log n)$ time. In all the remaining steps, finding a job with particular conditions needs $O(n)$ time by checking jobs one by one. Therefore, the overall time complexity of the algorithm is $O(n \log n)$, which implies a fast algorithm.

3 A $\frac{12}{7}$-approximation algorithm for $m = 3$

For $m = 3$, we essentially design a similar approach as for Algorithm $SPLT1$; we start by cutting the Johnsonian schedule σ into two parts. We will do this in such a way that the makespan of the first part is bounded from above by $\frac{4}{7} C_{\max}(\mathcal{J}) \leq \frac{12}{7} C^*_{\max}$ and the makespan of the second part is bounded from above by $\frac{16}{21} C_{\max}(\mathcal{J}) \leq \frac{16}{7} C^*_{\max}$; remember from Lemma 1 that $C_{\max}(\mathcal{J}) \leq 3C^*_{\max}$ if $m = 3$. We then use algorithm $SPLT1$ to cut the second part into two further parts and guarantee that both these further parts can be scheduled with a makespan smaller than $\frac{12}{7} C^*_{\max}$.

As before, let the Johnsonian schedule be σ, and let S_{ij} and C_{ij} be the earliest start and completion times of operations O_{ij} for $i = 1, 2$ and $j = 1, \ldots, n$. We set $T_1 = \frac{5}{21} C_{\max}(\mathcal{J})$, $T_2 = \frac{16}{21} C_{\max}(\mathcal{J})$.

Algorithm 2 $SPLT2$

Step 1. (Initialization) Re-index the job set $\mathcal{J}$ *according to the Johnson's rule.*

Let $S_{11} = 0$, $C_{11} = S_{11} + p_{11}$, $S_{21} = C_{11}$, $C_{21} = S_{21} + p_{21}$.

For $j = 2$ *to* n, *perform the following computations:*

$S_{1j} = C_{1(j-1)}$, $C_{1j} = S_{1j} + p_{1j}$, $S_{2j} = \max\{C_{1j}, C_{2(j-1)}\}$, $C_{2j} = S_{2j} + p_{2j}$.

Let $C_{\max}(\mathcal{J}) = C_{2n}$, *and* $T_1 = \frac{5}{21}C_{\max}(\mathcal{J})$, $T_2 = \frac{16}{21}C_{\max}(\mathcal{J})$.

Step 2. Find a job J_h *with* $S_{1h} \leq T_1 \leq C_{1h}$. *If job* J_h *does not exist, find a job* J_k *with* $S_{2k} \leq T_1 \leq C_{2k}$. *Let* $\mathcal{J}^1 = \{J_1, \ldots, J_k\}$, *and* $\mathcal{J}^2 = \{J_{k+1}, \ldots, J_n\}$. *Stop; otherwise, go to Step 3 with job* J_h.

Step 3. For job J_h, *if* $C_{2h} \leq \frac{4}{7}C_{max}$ *or* $C_{1h} = S_{2h}$, *let* $\mathcal{J}^1 = \{J_1, \ldots, J_h\}$, *and* $\mathcal{J}^2 = \{J_{h+1}, \ldots, J_n\}$. *Stop; otherwise, go to Step 4.*

Step 4. Let $C'_{1n} = S_{2n}$ *and* $S'_{1n} = C'_{1n} - p_{1n}$.

For $j = (n-1)$ *to* 1, *perform the following computations:*

$C'_{1j} = \min\{S'_{1(j+1)}, S_{2j}\}$ *and* $S'_{1j} = C'_{1j} - p_{1j}$, *where* S'_{1j} *and* C'_{1j} *are the latest possible start and completion time of job* J_j *in machine* M_1.

Step 5. Find a job J_t *with* $S'_{2t} \leq T_2 < C'_{2t}$. *If job* J_t *does not exists, we have solved this case in Step 3. If* $S'_{1t} \geq \frac{3}{7}C_{\max}(\mathcal{J})$ *or* $C'_{1t} = S'_{2t}$, *let* $\mathcal{J}^1 = \{J_t, \ldots, J_n\}$, *and* $\mathcal{J}^2 = \{J_1, \ldots, J_t\}$. *Stop; otherwise, go to Step 6.*

Step 6. In schedule σ, *find the job* J_u *with* $p_{1u} \leq p_{2u}$ *and* $p_{1(u+1)} > p_{2(u+1)}$, *and let* $v = u + 1$. *Therefore, in schedule* σ, *we have* $p_{1j} \leq p_{2j}$ *for* $j = 1, \ldots, u$; *and* $p_{1j} > p_{2j}$ *for* $j = v, \ldots, n$. *Then, we branch into the two cases.*

Case 1. $\sum_{j=v}^{n} p_{1j} \geq T_1$. *Find a job* J_e *with* $e \geq v$ *such that* $\sum_{j=v}^{e-1} p_{1j} < T_1 \leq \sum_{j=v}^{e} p_{1j}$ *and a job* J_d *with* $d < v$ *such that* $\sum_{j=d+1}^{e-1} p_{2j} < T_1 \leq \sum_{j=d}^{e-1} p_{2j}$. *We branch into six subcases.*

Subcase 1.1 $\sum_{j=v}^{e} p_{2j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_v, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 1.2 $\sum_{j=1}^{e-1} p_{2j} < T_1$. *Find a job* J_k *with* $\sum_{j=k+1}^{e} p_{2j} < T_1 \leq \sum_{j=k}^{e} p_{2j}$ *and* $1 \leq k < e$. *Let* $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 1.3 $\sum_{j=d}^{v-1} p_{1j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_d, \ldots, J_{v-1}\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 1.4 $\sum_{j=d}^{v} p_{1j} \geq T_1$ *and* $\sum_{j=d}^{v} p_{2j} \geq T_1$. *If* $v < e$, *let* $\mathcal{J}^1 = \{J_d, \ldots, J_v\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *If* $v = e$, *find a job* J_k *with* $\sum_{j=k+1}^{e} p_{2j} < T_1 \leq \sum_{j=k}^{e} p_{2j}$ *and* $d \leq k < e$. *Let* $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 1.5 $\sum_{j=d}^{e-1} p_{1j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_d, \ldots, J_{e-1}\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 1.6 $\sum_{j=d}^{e-1} p_{1j} < T_1$. *Find a job* J_k *with* $d \leq k < v$ *such that* $\sum_{k+1}^{e} p_{2j} < T_1 \leq \sum_{k}^{e} p_{2j}$, $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Case 2. $\sum_{j=1}^{u} p_{2j} \geq T_1$. *Find a job* J_d *with* $d \leq u$ *such that* $\sum_{j=d+1}^{u} p_{2j} < T_1 \leq \sum_{j=d}^{u} p_{2j}$ *and a job* J_e *with* $e > u$ *such that* $\sum_{j=d+1}^{e-1} p_{1j} < T_1 \leq \sum_{j=d+1}^{e} p_{1j}$. *We branch into six subcases.*

Subcase 2.1 $\sum_{j=d}^{u} p_{1j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_d, \ldots, J_u\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.2 $\sum_{j=d+1}^{n} p_{1j} < T_1$. *Find a job* J_k *with* $\sum_{j=d}^{k-1} p_{1j} < T_1 \leq \sum_{j=d}^{k} p_{1j}$ *and* $u < k \leq n$. *Let* $\mathcal{J}^1 = \{J_d, \ldots, J_k\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.3 $\sum_{j=u+1}^{e} p_{2j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_{u+1}, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.4 $\sum_{j=u}^{e} p_{1j} \geq T_1$ *and* $\sum_{j=u}^{e} p_{2j} \geq T_1$. *If* $d < u$, *let* $\mathcal{J}^1 = \{J_u, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *If* $d = u$, *find a job* J_k *with* $\sum_{j=d}^{k-1} p_{1j} < T_1 \leq \sum_{j=d}^{k} p_{1j}$ *and* $d < k \leq e$. *Let* $\mathcal{J}^1 = \{J_d, \ldots, J_k\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.5 $\sum_{j=d+1}^{e} p_{2j} \geq T_1$. *Let* $\mathcal{J}^1 = \{J_{d+1}, \ldots, J_e\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Subcase 2.6 $\sum_{j=d+1}^{e} p_{2j} < T_1$. *Find a job* J_k *with* $u < k \leq e$ *such that* $\sum_{d}^{k-1} p_{2j} < T_1 \leq \sum_{d}^{k} p_{2j}$, $\mathcal{J}^1 = \{J_d, \ldots, J_k\}$ *and* $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. *Stop.*

Algorithm $SPLT2$ gives two job sets $\mathcal{J}^1$ and $\mathcal{J}^2$, with $C_{\max}(\mathcal{J}^1) \leq \frac{12}{7} C^*_{\max}$ and $C_{\max}(\mathcal{J}^2) \leq \frac{16}{7} C^*_{\max}$. We can then apply Algorithm $SPLT1$ to the job set $\mathcal{J}^2$, which gives two further job sets for which have makespan bounded by $\frac{12}{7} C^*_{\max}$. We have therefore the following result.

Theorem 2 *Algorithm* $SPLT2$ *is a* $\frac{12}{7}$*-approximation for the problem of minimizing makespan in three parallel two-stage flow shops.* □

The detailed proof of Theorem 2 is shown in Appendix A. In Step 1 of the algorithm $SPLT2$, the re-indexing process runs in $O(n \log n)$ time. In the remaining steps, finding a job with particular conditions needs $O(n)$ time by checking jobs one by one. Therefore, the overall time complexity of the algorithm is again $O(n \log n)$.

4 Conclusions

We have developed approximation algorithms with worst-case performance guarantees for scheduling jobs in a flexible manufacturing environment with two and three two-stage parallel flow shops. The key idea is to judiciously cut the Johnsonian schedule in two and three parts, respectively, and schedule each part in a different flow shop.

Our results apply also to the makespan parallel flow shop problem with transportation times, in which the operations of the same job can be performed in different flow shops and where transporting job J_j from one flow shop to another requires a transportation time $\tau_j \geq 0$ ($j = 1, \ldots, n$). This is so, since in our algorithms transfer of jobs does not take place.

If $\tau_j = 0$ for each j, then the parallel flow shop problem with transportation times reduces to the hybrid flow shop problem, and our approximation algorithm has the same worst-case performance guarantee as the algorithms by Chen (1994) and Lee and Vairaktarakis (1994) when m=2.

Acknowledgement

The authors thank the anonymous reviewers and the editor for their valuable comments on improving an earlier version. This research was supported in part by projects of National Natural Science Foundation of China (No. 70832002 and No. 10971034).

References

R. Ruiz, J. Vazquez-Rodriguez, The hybrid flow shop scheduling problem, European Journal of Operational Research 205 (1) (2010) 1–18.

I. Ribas, R. Leisten, J. Framiñan, Review and classification of hybrid flow shop scheduling problems from a production system and a solutions procedure perspective, Computers & Operations Research 37 (8) (2010) 1439–1454.

B. Naderi, R. Ruiz, M. Zandieh, Algorithms for a realistic variant of flowshop scheduling, Computers & Operations Research 37 (2) (2010) 236–246.

S. Johnson, Optimal two-and three-stage production schedules with setup times included, Naval Research Logistics Quarterly 1 (1).

J. Hoogeveen, J. Lenstra, B. Veltman, Preemptive scheduling in a two-stage multiprocessor flow shop is NP-hard, European Journal of Operational Research 89 (1) (1996) 172–175.

B. Chen, Analysis of classes of heuristics for scheduling a two-stage flow shop with parallel machines at one stage, Journal of the Operational Research Society (1995) 234–244.

J. Gupta, Two-stage, hybrid flowshop scheduling problem, Journal of the Operational Research Society (1988) 359–364.

J. Gupta, E. Tunc, Schedules for a two-stage hybrid flowshop with parallel machines at the second stage, International Journal of Production Research 29 (7) (1991) 1489–1502.

J. Gupta, A. Hariri, C. Potts, Scheduling a two-stage hybrid flow shop with parallel machines at the first stage, Annals of Operations Research 69 (1997) 171–191.

R. Linn, W. Zhang, Hybrid flow shop scheduling: a survey, Computers & industrial engineering 37 (1-2) (1999) 57–61.

H. Wang, Flexible flow shop scheduling: optimum, heuristics and artificial intelligence solutions, Expert Systems 22 (2) (2005) 78–85.

B. Chen, Scheduling multiprocessor flow shops, In New Advances in Optimization and Approximation.(D,-Z. Du and J. Sun, Eds.) (1994) 1–8.

C. Lee, G. Vairaktarakis, Minimizing makespan in hybrid flowshops, Operations Research Letters 16 (3) (1994) 149–158.

G. Vairaktarakis, M. Elhafsi, The use of flowlines to simplify routing complexity in two-stage flowshops, IIE Transactions 32 (8) (2000) 687–699.

X. Qi, New Results for Scheduling Two Flowlines, Working paper of Hong Kong University of Science and Technology .

D. Cao, M. Chen, Parallel flowshop scheduling using Tabu search, International Journal of Production Research 41 (13) (2003) 3059–3073.

A. Al-Salem, A Heuristic to Minimize Makespan in Proportional Parallel Flow Shops, International Journal of Computing & Information Sciences 2 (2) (2004) 98.

Appendix A: Proof of Theorem 2

Lemma 13 *If there exists no job J_h with $S_{1h} \leq T_1 \leq C_{1h}$, then let $\mathcal{J}^1 = \{J_1, \ldots, J_k\}$ and $\mathcal{J}^2 = \{J_{k+1}, \ldots, J_n\}$ with J_k such that $S_{2k} \leq T_1 \leq C_{2k}$. We then have that*

$$C_{\max}(\mathcal{J}^1) \leq \frac{12}{7} C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \leq \frac{16}{7} C^*_{\max}.$$

Proof. Since there is no job J_h with $S_{1h} \leq T_1 \leq C_{1h}$, machine M_1 is idle after T_1. Furthermore, there must exist a job J_k with $S_{2k} \leq T_1 \leq C_{2k}$, otherwise machine M_2 would

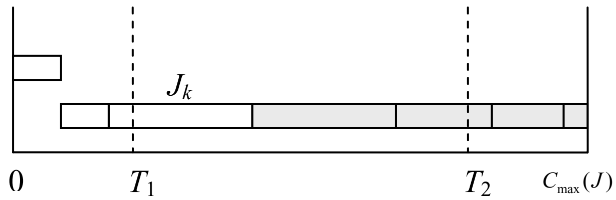

Figure 15: Cutting the Johnsonian schedule as prescribed in Lemma 13.

be idle after T_1, too. We then let $\mathcal{J}^1 = \{J_1, \ldots, J_k\}$, and $\mathcal{J}^2 = \{J_{k+1}, \ldots, J_n\}$. This case is illustrated by Figure 15.

Since $S_{2k} \le T_1$, we have $C_{\max}(\mathcal{J}^1) = S_{2k} + p_{2k} \le T_1 + C^*_{\max} = \frac{5}{21}C_{\max}(\mathcal{J}) + C^*_{\max} \le \frac{12}{7}C^*_{\max}$. And due to $C_{2k} \ge T_1$, we get $C_{\max}(\mathcal{J}^2) \le C_{\max}(\mathcal{J}) - C_{2k} \le \frac{16}{21}C_{\max}(\mathcal{J})$. □

Lemma 14 *If there is a job J_h with $S_{1h} \le T_1 \le C_{1h}$ and $C_{2h} \le \frac{4}{7}C_{\max}(\mathcal{J})$ or $C_{1h} = S_{2h}$, let $\mathcal{J}^1 = \{J_1, \ldots, J_h\}$, and $\mathcal{J}^2 = \{J_{h+1}, \ldots, J_n\}$. We then have that*

$$C_{\max}(\mathcal{J}^1) \le \frac{12}{7}C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \le \frac{16}{7}C^*_{\max}.$$

□

Proof. This case is visualized in Figure 16. The proof is similar to the one of Lemma 3. □

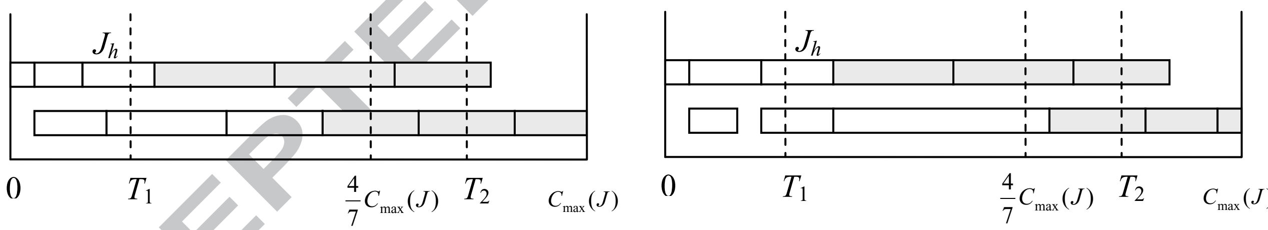

Figure 16: Cutting the Johnsonian schedule as indicated in Lemma 14.

Suppose now there is a job J_h with $S_{1h} \le T_1 \le C_{1h}$ for which $C_{2h} > \frac{4}{7}C_{\max}(\mathcal{J})$ and $C_{1h} < S_{2h}$. Then machine M_2 must be busy in the period $[T_1, \frac{4}{7}C_{\max}(\mathcal{J})]$, i.e. $\sum_{j=1}^{n} p_{2j} \ge \frac{4}{7}C_{\max}(\mathcal{J}) - \frac{5}{21}C_{\max}(\mathcal{J}) = \frac{1}{3}C_{\max}(\mathcal{J}) > T_1$. We now delay all operations O_{ij} in σ as much as possible within the makespan $C_{\max}(\mathcal{J})$. Let S'_{ij} and C'_{ij} denote the modified start and completion times of O_{ij} and let σ' denote the modified schedule.

Lemma 15 *In schedule σ', find a job J_t with $S'_{2t} \leq T_2 \leq C'_{2t}$. If $S'_{1t} \geq \frac{3}{7}C_{\max}(\mathcal{J})$ or $C'_{1t} = S'_{2t}$, let $\mathcal{J}^1 = \{J_t, \ldots, J_n\}$, and $\mathcal{J}^2 = \{J_1, \ldots, J_{t-1}\}$. We then have that*

$$C_{\max}(\mathcal{J}^1) \leq \frac{12}{7}C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \leq \frac{16}{7}C^*_{\max}.$$

Proof. Because there is a job J_h with $S_{1h} \leq T_1 \leq C_{1h}$ for which $C_{2h} > \frac{4}{7}C_{\max}(\mathcal{J})$ and $C_{1h} < S_{2h}$, we have $\sum_{j=1}^{n} p_{2j} > T_1$. Job J_t does exist. This case is visualized in Figure 17.

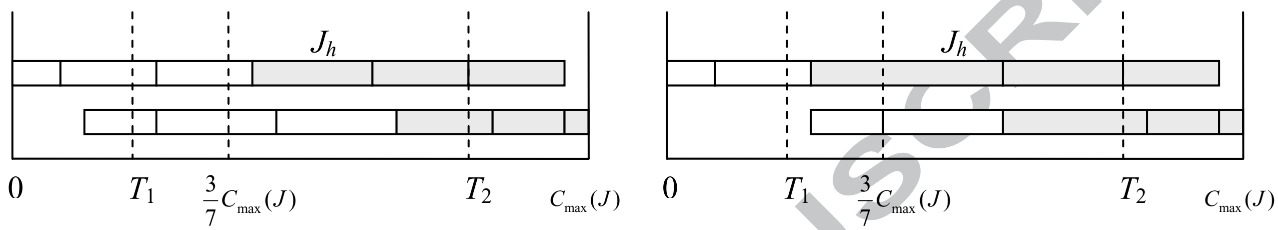

Figure 17: Cutting the Johnsonian schedule as indicated in Lemma 15.

Since $S'_{2t} \leq T_2$, we have

$$C_{\max}(\mathcal{J}^2) = C_{2(t-1)} \leq S'_{2t} \leq T_2 \leq \frac{16}{21}C_{\max}(\mathcal{J}).$$

If $S'_{1t} \geq \frac{3}{7}C_{\max}(\mathcal{J})$, then $C_{\max}(\mathcal{J}^1) \leq C_{\max}(\mathcal{J}) - S'_{1t} \leq \frac{4}{7}C_{\max}(\mathcal{J}) = \frac{12}{7}C^*_{\max}$. If $S'_{1t} < \frac{3}{7}C_{\max}(\mathcal{J})$, then we have $C'_{1t} = S'_{2t}$, and hence $C_{\max}(\mathcal{J}^1) \leq p_{1t} + p_{2t} + (C_{\max}(\mathcal{J}) - C_{2t}) \leq C^*_{\max} + \frac{5}{21}C_{\max}(\mathcal{J}) = \frac{12}{7}C^*_{\max}$. □

Lemma 13 to Lemma 15 have solved many different cases of this problem. The one remaining case is where there exists a job J_t with $S'_{2t} \leq T_2 \leq C'_{2t}$, $S'_{1t} < \frac{3}{7}C_{\max}(\mathcal{J})$, $C'_{1t} < S'_{2t}$, and a job J_h with $S_{1h} \leq T_1 \leq C_{1h}$, $C_{2h} > \frac{4}{7}C_{\max}(\mathcal{J})$ and $C_{1h} < S_{2h}$. This case is illustrated in Figure 18.

In this remaining case, machine M_2 must be busy in the period $[T_1, \frac{4}{7}C_{\max}(\mathcal{J})]$ in schedule σ, for otherwise, operation O_{2h} could have been started earlier; in schedule σ', machine M_1 is busy in the period $[\frac{3}{7}C_{\max}(\mathcal{J}), T_2]$, for otherwise, operation O_{1t} could have been started later.

In what follows, we deal with the remaining case with jobs J_h and J_t only. We split the n jobs into two subsets $\mathcal{S}^1 = \{J_1, \ldots, J_u\} = \{J_j | p_{1j} \leq p_{2j}, j = 1, \ldots, n\}$ and $\mathcal{S}^2 = \{J_v, \ldots, J_n\} = \{J_j | p_{1j} > p_{2j}, j = 1, \ldots, n\}$. We then branch into two cases: the case

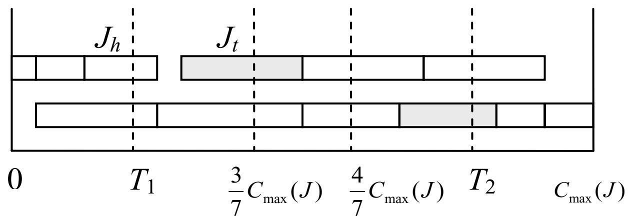

Figure 18: The remaining case with jobs J_h and J_t.

$\sum_{j=v}^{n} p_{1j} \geq T_1$, and the case $\sum_{j=1}^{u} p_{2j} \geq T_1$. Since they are symmetrical, we analyze the first case only.

Since $\sum_{j=v}^{n} p_{1j} \geq T_1$, we can find a job J_e with $e \geq v$ such that $\sum_{j=v}^{e-1} p_{1j} < T_1 \leq \sum_{j=v}^{e} p_{1j}$. We have the following Lemma.

Lemma 16 *If $\sum_{j=v}^{e} p_{2j} \geq T_1$, then let $\mathcal{J}^1 = \{J_v, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. Then*

$$C_{\max}(\mathcal{J}^1) \leq \frac{12}{7} C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \leq \frac{16}{7} C^*_{\max}.$$

Proof. This case is illustrated by Figure 19.

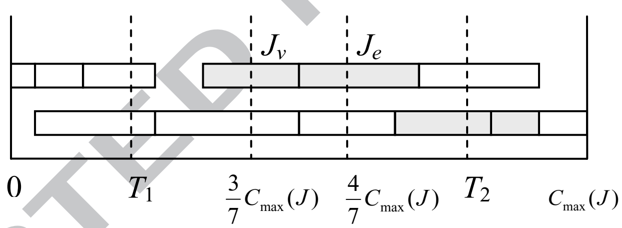

Figure 19: Cutting the Johnsonian schedule as indicated in Lemma 16.

Let $C_{\max}(\mathcal{J}^1) = \sum_{j=v}^{w} p_{1j} + \sum_{j=w}^{e} p_{2j}$ and $v \leq w \leq e$. We must have $p_{2e} \leq p_{2w} < p_{1w}$ and $\sum_{j=v}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} < \sum_{j=v}^{e-1} p_{1j} < T_1$. Then, $C_{\max}(\mathcal{J}^1) = \sum_{j=v}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} + p_{1w} + p_{2w} < T_1 + C^*_{\max} = \frac{12}{7} C^*_{\max}$. The proof for $C_{\max}(\mathcal{J})$ is analogous to the proof of Lemma 7. □

If the condition in Lemma 16 is not satisfied, we need to find a job J_d with $d < v$ such that $\sum_{j=d-1}^{e-1} p_{2j} < T_1 \leq \sum_{j=d}^{e-1} p_{2j}$. If there is no such job J_d, we have the following result.

Lemma 17 *If there is no job J_d with $d < v$ such that $\sum_{j=d-1}^{e-1} p_{2j} < T_1 \leq \sum_{j=d}^{e-1} p_{2j}$, we find a job J_k with $\sum_{j=k+1}^{e} p_{2j} < T_1 \leq \sum_{j=k}^{e} p_{2j}$ and $1 \leq k < e$, and we let $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ and $\mathcal{J}^2 = \{J \backslash \mathcal{J}^1\}$. We then have that*

$$C_{\max}(\mathcal{J}^1) \leq \frac{12}{7} C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \leq \frac{16}{7} C^*_{\max}.$$

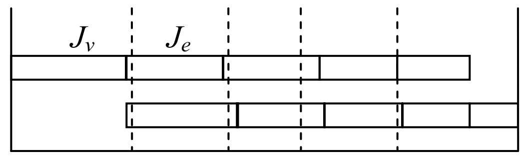

Figure 20: Cutting the Johnsonian schedule as indicated in Lemma 17.

Proof. This case is visualized in Figure 20, where $k = v = 1$. In this case, we have $e \geq h$, since $\sum_{j=v}^{e} p_{1j} \geq T_1$ and $\sum_{j=1}^{h-1} p_{1j} \leq T_1$. Furthermore, we have $k < v$, for otherwise we would have $\sum_{j=v}^{e} p_{2j} \geq T_1$, which already has been covered by Lemma 16. With $C_{2h} > \frac{4}{7} C_{\max}(\mathcal{J})$ and machine M_2 being busy in the period $[\frac{5}{21} C_{\max}(\mathcal{J}), \frac{4}{7} C_{\max}(\mathcal{J})]$, we have $\sum_{j=1}^{e} p_{2j} > T_1$. Therefore job J_k exists. Since $\sum_{j=1}^{h} p_{2j} > T_1$ and $\sum_{j=1}^{e-1} p_{2j} < T_1$, we have $e-1 < h$. Since also $e \geq h$, we must have that $e = h$. Then we have $\sum_{j=k}^{e-1} p_{1j} \leq \sum_{j=1}^{h-1} p_{1j} < T_1$. If $C_{\max}(\mathcal{J}^1) = \sum_{j=k}^{w} p_{1j} + \sum_{j=w}^{e} p_{2j}$ and $v \leq w \leq e$, we must have $p_{2e} \leq p_{2w} < p_{1w}$ and $\sum_{j=k}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} \leq \sum_{j=k}^{e-1} p_{1j} < T_1$. Then, $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} + p_{1w} + p_{2w} < T_1 + C^*_{\max} = \frac{12}{7} C^*_{\max}$. If $C_{\max}(\mathcal{J}^1) = \sum_{j=k}^{w} p_{1j} + \sum_{j=w}^{e} p_{2j}$ and $k \leq w < v$, we must have $p_{1k} \leq p_{1w} \leq p_{2w}$ and $\sum_{j=k}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} \leq \sum_{j=k+1}^{e} p_{2j} < T_1$. Then, $C_{\max}(\mathcal{J}^1) = \sum_{j=d}^{w-1} p_{1j} + \sum_{j=w+1}^{e} p_{2j} + p_{1w} + p_{2w} < T_1 + C^*_{\max} = \frac{12}{7} C^*_{\max}$. Because of $k < v$, we also have $\sum_{j=k}^{e} p_{1j} \geq T_1$. Since $\sum_{j=k}^{e} p_{2j} \geq T_1$, the proof of $C_{\max}(\mathcal{J}^2)$ is analogous to Lemma 7. □

If there exists a job J_e with $e \geq v$ such that $\sum_{j=v}^{e-1} p_{1j} < T_1 \leq \sum_{j=v}^{e} p_{1j}$ and a job J_d with $d < v$ such that $\sum_{j=d-1}^{e-1} p_{2j} < T_1 \leq \sum_{j=d}^{e-1} p_{2j}$, we have the following Lemmata 18 - 21. Their proofs are similar to those of Lemma 8 - 11.

Lemma 18 *If $\sum_{j=d}^{v-1} p_{1j} \geq T_1$, let $\mathcal{J}^1 = \{J_d, \ldots, J_{v-1}\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. We then have that*

$$C_{\max}(\mathcal{J}^1) \leq \frac{12}{7} C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \leq \frac{16}{7} C^*_{\max}.$$

□

Lemma 19 *If $\sum_{j=d}^{v} p_{1j} \geq T_1$ and $\sum_{j=d}^{v} p_{2j} \geq T_1$, we have two cases. If $v < e$, let $\mathcal{J}^1 = \{J_d, \ldots, J_v\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. If $v = e$, find a job J_k with $\sum_{j=k+1}^{e} p_{2j} < T_1 \leq \sum_{j=k}^{e} p_{2j}$ and $d \leq k < e$. Let $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. We then have that*

$$C_{\max}(\mathcal{J}^1) \leq \frac{12}{7} C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \leq \frac{16}{7} C^*_{\max}.$$

□

Lemma 20 *In case of $\sum_{j=d}^{e-1} p_{1j} \geq T_1$, let $\mathcal{J}^1 = \{J_d, \ldots, J_{e-1}\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. We then have that*

$$C_{\max}(\mathcal{J}^1) \leq \frac{12}{7} C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \leq \frac{16}{7} C^*_{\max}.$$

□

Lemma 21 *In case of $\sum_{j=d}^{e-1} p_{1j} < T_1$, find a job J_k with $d \leq k < v$ such that $\sum_{k+1}^{e} p_{2j} < T_1 \leq \sum_{k}^{e} p_{2j}$, $\mathcal{J}^1 = \{J_k, \ldots, J_e\}$ and $\mathcal{J}^2 = \{\mathcal{J} \backslash \mathcal{J}^1\}$. We then have that*

$$C_{\max}(\mathcal{J}^1) \leq \frac{12}{7} C^*_{\max} \text{ and } C_{\max}(\mathcal{J}^2) \leq \frac{16}{7} C^*_{\max}.$$

□

Using Lemmata 16 - 21, we have solved the case $\sum_{j=v}^{n} p_{1j} \geq T_1$. The algorithm for the case $\sum_{j=1}^{u} p_{2j} \geq T_1$ is symmetrical. For the makespan parallel flow shop problem with $m = 3$, Lemma 12 still holds.

We have now developed an approximation algorithm, referred to as Algorithm $SPLT2$, for the parallel flow shop problem with $m = 3$ with worst-case performance guarantee $\frac{12}{7}$.

Highlights

>We consider the problem of scheduling n jobs in m two-stage parallel flow shops. >For m=2, we present a 3/2-approximation algorithm so as to minimize the makespan. >For m=3, we present a 12/7-approximation algorithm.> Both these algorithms run in O(nlogn) time. >These are the first approximation algorithms with fixed worst-case guarantees.

ACCEPTED MANUSCRIPT