Applied Mathematics, 2019, 10, 947-955
https://www.scirp.org/journal/am
ISSN Online: 2152-7393
ISSN Print: 2152-7385

LPT Algorithm for Jobs with Similar Sizes on Three Machines

Yajie Ma[1], Rongheng Li[1*], Yunxia Zhou[2]

[1]Key Laboratory of Computing and Stochastic Mathematics (Ministry of Education), School of Mathematics and Statistics, Hunan Normal University, Changsha, China
[2]College of Information Science and Engineering, Hunan Normal University, Changsha, China
Email: *lirongheng@hunnu.edu.cn

How to cite this paper: Ma, Y.J., Li, R.H. and Zhou, Y.X. (2019) *LPT* Algorithm for Jobs with Similar Sizes on Three Machines. *Applied Mathematics*, **10**, 947-955.
https://doi.org/10.4236/am.2019.1011066

Received: October 18, 2019
Accepted: November 8, 2019
Published: November 11, 2019

Abstract

In this paper, *LPT* (largest processing time) algorithm is considered for scheduling jobs with similar sizes on three machines. The objective function is to minimize the maximum completion time of all machines. The worst case performance ratio of the *LPT* algorithm is given as a piecewise linear function of r if job sizes fall in $[1,r]$. Our result is better than the existing result. Furthermore, the ratio given here is the best. That means our result cannot be improved any more.

Keywords

LPT Algorithm, Parallel Machine, Performance Ratio, Schedule

1. Introduction

The scheduling problem on m parallel identical machines is defined as follows: Given a job set $L=\{J_1,J_2,\cdots,J_n\}$ of n jobs where job J_j has non-negative processing time p_j, assign the jobs on m machines $\{M_1,M_2,\cdots,M_m\}$ so as to minimize the maximum completion times of the jobs on each machine. Since scheduling problem was proposed by Graham [1], it has been studied extensively in many varieties and from many viewpoints. In classic scheduling problem, there is no constraint on the size of jobs. However, in practice, the size of job can neither be too large nor too small. This motivates researchers to study scheduling problems in which the sizes of all jobs fall in $[1,r]$ with $r\geq 1$. Researches of such model can be found in [2]-[9] to name a few.

LPT (Largest Processing Time) algorithm is a famous algorithm proposed by Graham [10]. For a given job list $L=\{J_1,J_2,\cdots,J_n\}$ of n jobs, *LPT* algorithm firstly sorts the jobs with a non-increasing order of their sizes. Then *LPT* algorithm assigns the jobs one by one according to the non-increasing order and always assigns the current job to the machine with least load. The worst case performance ratio of *LPT* is $\frac{4}{3}-\frac{1}{3m}$. H.Kellerer [9] gave the following result.

$$R(m,LPT,\mu)$$
$$\le \begin{cases} \frac{m(2k-2)-\mu(k-2)}{km} & \text{for } \mu \in \left[\frac{1+m(k-3)}{k-2}, \frac{m(k^2-2k-2)+k}{(k-2)(k+1)}\right] \\ \frac{(k+2)m-1}{(k+1)m} & \text{for } \mu \in \left[\frac{m(k^2-2k-2)+k}{(k-2)(k+1)}, \frac{1+m(k-2)}{k-1}\right] \end{cases}$$

where $k \ge 3, k \in N$, $\frac{p_1}{p_n} = 2-\frac{\mu}{m}$, ($1 \le \mu < m, \mu \in R$). For $\mu \le (m+3)/4 (k=3)$, the bound of $(4m-\mu)/3m$ is tight. We use $p_j \in [1,r]$ with $r \ge 1$ instead of $\frac{p_1}{p_n} = 2-\frac{\mu}{m}$. Then the tight interval for r is $\left[\frac{3}{2}, \frac{5}{3}\right]$ in Kellerer's result. In this paper, we will give the tight bound as a piecewise linear function of r for $m=3$ and all $r \ge 1$.

2. Theorem and Its Proof

Before the analysis, we give some symbols used later on.

1) $p_{i(j)}$ represents the size of the j-th job assigned on machine M_i by *LPT* algorithm.

2) $S_i = \{ j \ne n \mid J_j \text{ is assigned on machine } M_i \text{ by } LPT \text{ algorithm} \}$.

3) $S_i^* = \{ j \mid J_j \text{ is assigned on machine } M_i \text{ by optimal algorithm} \}$.

4) $C_{\max}^{LPT}, C_{\max}^{OPT}$ represent the makespan of optimal algorithm and *LPT* algorithm, respectively.

In the following of this paper, for a given job list $L = \{J_1, J_2, \cdots, J_n\}$, we always assume $p_1 \ge p_2 \ge \cdots \ge p_n$ and $1 \le p_j \le r, j = 1,2,\cdots,n$.

Lemma 1 If $r < 2$, then in *LPT* schedule, the difference of the numbers of jobs on any two machines is at most 1.

Proof: If it is not true, suppose it is the first time that there are k jobs assigned on M_h and there are $k+2$ jobs assigned on M_i in the *LPT* schedule, then we have

$$\sum_{s=1}^{k} p_{h(s)} \ge \sum_{s=1}^{k+1} p_{i(s)}.$$

Hence we get

$$p_{h(1)} \ge \sum_{s=1}^{k-1} \left(p_{i(s)} - p_{h(s+1)} \right) + p_{i(k)} + p_{i(k+1)}$$

By the assumption that this is the first time of appearing the case, we have $p_{i(s)} \ge p_{h(s+1)}$ for $s = 1,2,3,\cdots,k-1$. That means

$$p_{h(1)} \geq p_{i(k)} + p_{i(k+1)} \geq 2.$$

This is a contradiction to $r < 2$.

By Lemma 1, we can conclude that $p_{i(s)} \geq p_{h(s+1)}$ for any $1 \leq i, h \leq m$ in the LPT schedule.

Theorem 2. For any job list $L = \{J_1, J_2, \cdots, J_n\}$ and $m = 3$, we have

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq \begin{cases} \frac{11}{9}; & r \in \left[\frac{5}{3}, +\infty\right), \\ \frac{(2k+1)r+k+2}{3k+3}; & r \in \left[\frac{(2k+3)(3k+2)}{(2k+1)(3k+4)}, \frac{6k+5}{6k+3}\right), \\ \frac{9k+14}{3(3k+4)}; & r \in \left[\frac{3k+4}{3(k+1)}, \frac{(2k+3)(3k+2)}{(2k+1)(3k+4)}\right), \\ \frac{2(k+1)r+k+2}{3k+4}; & r \in \left[1+\frac{3k+4}{9(k+1)(k+2)}, \frac{3k+4}{3(k+1)}\right), \\ \frac{9k+20}{9(k+2)}; & r \in \left[\frac{6(k+1)+5}{3[2(k+1)+1]}, 1+\frac{3k+4}{9(k+1)(k+2)}\right), \\ 1; & r = 1, \end{cases}$$

where k is non-negative integer. Furthermore the bound is tight.

Proof: Case 1: $r \geq \frac{5}{3}$. For any $m \geq 1$, Graham [10] proved that

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq \frac{4}{3} - \frac{1}{3m}$$

and the quality can hold. We get the conclusion by letting $m = 3$.

For $r < \frac{5}{3}$, if the theorem is not true, then there is a job list $L = \{J_1, J_2, \cdots, J_n\}$ with minimal n such that L violates the theorem. We call such a job list as a minimal counter example. For a minimal counter example, it is easy to show that the last job J_n is finished at last. Hence we have

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq \frac{L_1 + L_2 + L_3 + 3p_n}{3C_{\max}^{OPT}(L)} \leq \frac{\sum_{j=1}^{n} p_j + 2p_n}{3C_{\max}^{OPT}(L)} \leq 1 + \frac{2p_n}{3C_{\max}^{OPT}(L)}, \tag{1}$$

Case 2: $\frac{(2k+3)(3k+2)}{(2k+1)(3k+4)} \leq r < \frac{6k+5}{3(2k+1)}$.

In this case, we should prove

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq \frac{(2k+1)r+k+2}{3k+3}. \tag{2}$$

If (2) is not true, by (1) we get

$$\frac{(2k+1)r+k+2}{3k+3} < \frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq 1 + \frac{2p_n}{3C_{\max}^{OPT}(L)}.$$

That means

$$C_{\max}^{OPT}(L) < \frac{2(k+1)p_n}{(2k+1)(r-1)} \le (3k+4)p_n.$$

Hence we get $n \le 9k+9$.

Case 2.1: $n = 9k+9$.

In this case we have $|S_1| = |S_2| = 3k+3$, $|S_3| = 3k+2$, $|S_1^*| = |S_2^*| = |S_3^*| = 3k+3$. It is easy to see that there exists $i \in \{1,2,3\}$ satisfying $|S_i^* \cap S_3| \ge k+1$. Without loss of generality, suppose $|S_1^* \cap S_3| \ge k+1$, then $|S_3 \setminus S_1^*| \le 2k+1$. Therefore we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le \frac{\sum_{j\in S_3} p_j + p_n}{\sum_{j\in S_1^*} p_j} = \frac{\sum_{j\in S_3\cap S_1^*} p_j + \sum_{j\in S_3\setminus S_1^*} p_j + 1}{\sum_{j\in S_3\cap S_1^*} p_j + \sum_{j\in S_1^*\setminus S_3} p_j}$$

$$\le \frac{|S_3 \cap S_1^*| + |S_3 \setminus S_1^*| r + 1}{|S_3 \cap S_1^*| + |S_1^* \setminus S_3|} = \frac{|S_3| + |S_3 \setminus S_1^*|(r-1) + 1}{|S_1^*|}$$

$$\le \frac{(2k+1)r + k + 2}{3k+3}.$$

Similarly we can get (2) for the case of $n = 9k+7, 9k+8$.

Case 2.2: $n = 3k'+1, k' \le 3k+1$.

Let $|S_1| = |S_2| = |S_3| = k'$, $|S_1^*| = |S_2^*| = k'$, $|S_3^*| = k'+1$, or $|S_1^*| = k'-1$, $|S_2^*| = |S_3^*| = k'+1$. At any case $|S_3^*| = k'+1$ holds and there exists $i \in \{1,2,3\}$ satisfying $|S_i \cap S_3^*| \ge \frac{k'+1}{3}$. W.l.o.g, suppose $|S_1 \cap S_3^*| \ge \frac{k'+1}{3}$, then $|S_1 \setminus S_3^*| \le \frac{2k'-1}{3}$. Hence we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le \frac{\sum_{j\in S_1} p_j + p_n}{\sum_{j\in S_3^*} p_j} = \frac{\sum_{j\in S_1\cap S_3^*} p_j + \sum_{j\in S_1\setminus S_3^*} p_j + 1}{\sum_{j\in S_1\cap S_3^*} p_j + \sum_{j\in S_3^*\setminus S_1} p_j}$$

$$\le \frac{|S_1 \cap S_3^*| + |S_1 \setminus S_3^*| r + 1}{|S_1 \cap S_3^*| + |S_3^* \setminus S_1|} = \frac{|S_1 \setminus S_3^*|(r-1) + |S_1| + 1}{|S_3^*|}$$

$$= \frac{(2k'-1)r + k' + 4}{2k'-1+k'+4} \le \frac{(2k'+1)r + k'+1+4}{2k'+1+k'+1+4}$$

$$= \frac{(2k'+1)r + k' + 5}{2k'+1+k'+5} \le \frac{(6k+3)r + 3k + 6}{9k+9}$$

$$= \frac{(2k+1)r + k + 2}{3k+3},$$

where the last two inequalities result from the facts that function $y = \frac{(2x-1)r + x + 4}{3x+3}$ and $y = \frac{(x+1)r + x + 5}{3x+6}$ are increasing function of *x*. Similarly we can get (2) for the case of $n = 3k'+i, k' \le 3k+1, i = 2,3$.

Case 3: $\frac{3k+4}{3(k+1)} \le r < \frac{(2k+3)(3k+2)}{(2k+1)(3k+4)}$.

In this case, if the claim is not true, then by (1) there is a the minimal counter example L satisfying

$$\frac{9k+14}{3(3k+4)} < \frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le 1 + \frac{2p_n}{3C_{\max}^{OPT}(L)}.$$

Hence we get

$$C_{\max}^{OPT}(L) < (3k+4)p_n.$$

That means $n \le 9k+9$. By the proof of Case 2 we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le \frac{(2k+1)r+k+2}{3k+3} \le \frac{9k+14}{3(3k+4)}.$$

Case 4: $1 + \frac{3k+4}{9(k+1)(k+2)} \le r < \frac{3k+4}{3(k+1)}$.

In this case, if the claim is not true, then by (1) there is a the minimal counter example L satisfying

$$\frac{2(k+1)r+k+2}{3k+4} < \frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le 1 + \frac{2p_n}{3C_{\max}^{OPT}(L)}$$

Hence we get

$$C_{\max}^{OPT}(L) < \frac{3k+4}{3(k+1)(r-1)} \le (3k+6)p_n.$$

That means $n \le 9k+15$.

Case 4.1: $n = 9k+15$.

In this case we have $|S_1| = |S_2| = 3k+5$, $|S_3| = 3k+4$, $|S_1^*| = |S_2^*| = |S_3^*| = 3k+5$. It is easy to see that there exists $i \in \{1,2,3\}$ satisfying $|S_3 \cap S_i^*| \ge k+2$. W.l.o.g, suppose $|S_3 \cap S_3^*| \ge k+2$. That means $|S_3 \setminus S_3^*| \le 2k+2$. Hence we get

$$\begin{aligned}
\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} &\le \frac{\sum_{j\in S_3} p_j + 1}{\sum_{j\in S_3^*} p_j} = \frac{\sum_{j\in S_3\cap S_3^*} p_j + \sum_{j\in S_3\setminus S_3^*} p_j + 1}{\sum_{j\in S_3^*\cap S_3} p_j + \sum_{j\in S_3^*\setminus S_3} p_j} \\
&\le \frac{|S_3 \cap S_3^*| + |S_3 \setminus S_3^*| r + 1}{|S_3^*|} = \frac{|S_3 \setminus S_3^*|(r-1) + |S_3| + 1}{|S_3^*|} \\
&\le \frac{(2k+2)(r-1)+3k+4+1}{3k+5} \le \frac{(2k+2)r+k+3}{3k+5} \\
&\le \frac{(2k+2)r+k+2}{3k+4},
\end{aligned}$$

where the last inequality results from the fact that $y = \frac{x+b}{x+1}(b \ge 1)$ is a decreasing function of *x*.

Similarly we can show that the claim is true for the cases of $n = 9k+i, i = 13,14$.

Case 4.2: $n = 3k'+3, k' \le 3k+3$.

In this case, let $|S_1| = |S_2| = k'+1$, $|S_3| = k'$. If There exists $i \in \{1,2,3\}$ satisfying $|S_3 \cap S_i^*| \ge \frac{k'+1}{3}$, w.l.o.g, suppose $|S_3 \cap S_1^*| \ge \frac{k'+1}{3}$, then we have

$\left|S_3 \setminus S_1^*\right| \le \frac{2k'-1}{3}$. Therefore we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le \frac{\sum_{j\in S_3} p_j + p_n}{\sum_{j\in S_1^*} p_j} = \frac{\sum_{j\in S_3\cap S_1^*} p_j + \sum_{j\in S_3\setminus S_1^*} p_j + p_n}{\sum_{j\in S_1^*\cap S_3} p_j + \sum_{j\in S_1^*\setminus S_3} p_j}$$

$$\le \frac{\left|S_3\cap S_1^*\right| + \left|S_3\setminus S_1^*\right| r + 1}{\left|S_3\cap S_1^*\right| + \left|S_1^*\setminus S_3\right|} = \frac{\left|S_3\setminus S_1^*\right|(r-1) + \left|S_3\right| + 1}{\left|S_1^*\right|}$$

$$\le \frac{\frac{2k'-1}{3}(r-1) + k' + 1}{k'+1} \le \frac{6kr + 5r + 3k + 7}{3(3k+4)}$$

$$\le \frac{(2k+2)r + k + 2}{3k+4}.$$

If $\left|S_1^*\right| = \left|S_2^*\right| = \left|S_3^*\right| = k'+1$, it is easy to see that there exists $i \in \{1,2,3\}$ satisfying $\left|S_3 \cap S_i^*\right| \ge \frac{k'+1}{3}$. Then we get the claim is true by above discussion.

If $\left|S_1^*\right| \ge k'+2, \left|S_3 \cap S_1^*\right| \ge \frac{k'+1}{3}$, we also get the claim is true by above discussion.

If $\left|S_1^*\right| \ge k'+2, \left|S_3 \cap S_1^*\right| < \frac{k'+1}{3}$, then there exists $i \in \{1,2\}$ satisfying $\left|S_i \cap S_1^*\right| \ge \frac{k'+3}{3}$, w.l.o.g, suppose $\left|S_1 \cap S_1^*\right| \ge \frac{k'+3}{3}$, then $\left|S_1 \setminus S_1^*\right| \le \frac{2k'}{3}$. Hence we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le \frac{\sum_{j\in S_1} p_j + 1}{\sum_{j\in S_1^*} p_j} = \frac{\sum_{j\in S_1\cap S_1^*} p_j + \sum_{j\in S_1\setminus S_1^*} p_j + 1}{\sum_{j\in S_1^*\cap S_1} p_j + \sum_{j\in S_1^*\setminus S_1} p_j}$$

$$\le \frac{\left|S_1\cap S_1^*\right| + \left|S_1\setminus S_1^*\right| r + 1}{\left|S_1\cap S_1^*\right| + \left|S_1^*\setminus S_1\right|} = \frac{\left|S_1\setminus S_1^*\right|(r-1) + \left|S_1\right| + 1}{\left|S_1^*\right|}$$

$$\le \frac{\frac{2k'}{3}(r-1) + k' + 1 + 1}{k'+2} \le \frac{(6k+6)r + 3k + 9}{3(3k+5)}$$

$$\le \frac{(2k+2)r + k + 2}{3k+4},$$

where the last inequality results from the fact that $y = \frac{x+b}{x+1}(b \ge 1)$ is a decreasing function of *x*.

Case 5: $\frac{6(k+1)+5}{3[2(k+1)+1]} \le r < 1 + \frac{3k+4}{9(k+1)(k+2)}$.

In this case we should prove

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le \frac{9k+20}{9(k+2)}.$$

If it is not true, then by (1) the minimal counter example satisfies

$$\frac{9k+20}{9(k+2)} < \frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \le \frac{\sum_{i=1}^{n} p_i + 2p_n}{3C_{\max}^{OPT}(L)}$$

Hence we get

$$C_{\max}^{OPT}(L) < (3k+6)p_n.$$

That means $n \le 9k+15$. By the proof of Case 4 we get

$$C_{\max}^{OPT}(L) \le \frac{2(k+1)r+k+2}{3k+4} \le \frac{9k+14}{3(3k+4)}.$$

By the above discussions of Case 1-5, the inequality in Theorem 2 is proved.

Now we prove the tightness of the inequalities. For $\frac{(2k+3)(3k+2)}{(2k+1)(3k+4)} \le r < \frac{6k+5}{3(2k+1)}$, let $L = \{J_1, J_2, \cdots, J_{9k+7}\}$ with $p_j = r, j = 1,2,\cdots,6k+2$, $p_{6k+3} = p_{6k+4} = r' = \frac{r+1}{2}$, $p_j = 1, j = 6k+5,\cdots,9k+7$. Then $L_1 = (2k+1)r+k+2$, $L_2 = (2k+1)r+k+1$, $L_3 = 2kr+k+2r'$, $L_1^* = 3k+3$, $L_2^* = L_3^* = (3k+1)r+r'$. Hence we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} = \frac{(2k+1)r+k+2}{3k+3}. \tag{3}$$

For $\frac{3k+4}{3(k+1)} \le r < \frac{(2k+3)(3k+2)}{(2k+1)(3k+4)}$. let $L = \{J_1, J_2, \cdots, J_{9k+10}\}$ with $p_j = r, j = 1,\cdots,6$; $p_j = r' = \frac{3k+4-3r}{3k}, j = 7,\cdots,6k+6$, $p_j = 1, j = 6k+7,\cdots,9k+10$. Then we have $L_1 = 2r+2kr'+k+2$, $L_2 = L_3 = 2r+2kr'+k+1$, $L_1^* = 3k+4$, $L_2^* = L_3^* = 3r+3kr'$. Hence we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} = \frac{9k+14}{3(3k+4)}. \tag{4}$$

For $1+\frac{3k+4}{9(k+1)(k+2)} \le r < \frac{3k+4}{3(k+1)}$, let $L = \{J_1, J_2, \cdots, J_{9k+10}\}$ with $p_j = r, j = 1,2,\cdots,6k+6$, $p_j = 1, j = 6k+7,\cdots,9k+10$. Then we have $L_1 = 2(k+1)r+k+2$, $L_2 = L_3 = 2(k+1)r+k+1$, $L_1^* = 3k+4$, $L_2^* = L_3^* = 3(k+1)r$. Hence we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} = \frac{2(k+1)r+k+2}{3k+4}. \tag{5}$$

For $\frac{6(k+1)+5}{3(2(k+1)+1)} \le r < 1+\frac{3k+4}{9(k+1)(k+2)}$, let $L = \{J_1, J_2, \cdots, J_{9k+16}\}$ with $p_1 = p_2 = r$, $p_3 = p_4 = r_1 = \frac{r+1}{2}$, $p_j = r_2 = \frac{3k+6-r-r_1}{3k+3}, j = 5,6k+10$, $p_j = 1, j = 6k+11,\cdots,9k+16$. Then we have $L_1 = r+2(k+1)r_2+k+3$, $L_2 = r+2(k+1)r_2+k+2$, $L_3 = 2r_1+2(k+1)r_2+k+1$, $L_1^* = L_2^* = r+r_1+3(k+1)r_2$, $L_3^* = 3k+6$. Hence we get

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} = \frac{9k+20}{9k+18}. \tag{6}$$

Hence Theorem 2 is proved.

3. Conclusion and Future Work

In this paper, *LPT* algorithm is considered for the scheduling jobs with similar sizes in $[1, r]$ on three machines. The objective function is to minimize the maximum completion time of all machines. The worst performance ratio of the *LPT* algorithm is given as a piecewise linear function of *r*. The result is better than the existing result and cannot be improved any more. It is interesting to consider general number of machines and give tight bound of worst case performance.

Acknowledgements

The authors would like to express their thanks to the National Natural Science Foundation of China for financially supporting under Grant No. 11471110 and the Foundation Grant of Education Department of Hunan (No. 16A126).

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Graham, R.L. (1969) Bounds on Multiprocessing Timing Anomalies. *SIAM Journal on Applied Mathematics*, **17**, 416-429. https://doi.org/10.1137/0117039

[2] Cheng, T.C.E., Kellerer, H. and Kotov, V. (2012) Algorithms Better than LPT for Semi-Online Scheduling with Decreasing Processing Times. *Operations Research Letters*, **40**, 349-352. https://doi.org/10.1016/j.orl.2012.05.009

[3] He, Y.G. (2005) Semi-Online Scheduling Jobs with Tightly-Grouped Processing Times on Three Identical Machines. *Discrete Applied Mathematics*, **150**, 140-159. https://doi.org/10.1016/j.dam.2004.12.005

[4] He, Y. and Zhang, G. (1999) Semi On-Line Scheduling on Two Identical Machines. *Computing*, **62**, 179-187. https://doi.org/10.1007/s006070050020

[5] Kellerer, H., Kotov, V., Speranza, M.G. and Tuza, Z. (1997) Semi On-Line Algorithms for the Partition Problem. *Operations Research Letters*, **21**, 235-242. https://doi.org/10.1016/S0167-6377(98)00005-4

[6] Li, R.H. and Huang, H.C. (2007) List Scheduling for Jobs with Arbitrary Release Times and Similar Lengths. *Journal of Scheduling*, **10**, 365-373. https://doi.org/10.1007/s10951-007-0042-8

[7] Lin, L. and Tan, Z. (2014) Inefficiency of Nash Equilibrium for Scheduling Games with Constrained Jobs: A Parametric Analysis. *Theoretical Computer Science*, **521**, 123-134. https://doi.org/10.1016/j.tcs.2013.11.012

[8] He, Y. and Dosa, G. (2005) Semi-Online Scheduling Jobs with Tightly-Grouped Processing Times on Three Identical Machines. *Discrete Applied Mathematics*, **150**, 140-159. https://doi.org/10.1016/j.dam.2004.12.005

[9] Kellerer, H. (1991) Bounds for Non-Preemptive Scheduling Jobs with Similar Processing Times on Multiprocessor Systems Using LPT-Algorithm. *Computing*,

46, 183-191. https://doi.org/10.1007/BF02238297

[10] Graham, R.L. (1976) Bounds on the Performance of Scheduling Algorithms, Computer and Job-Shop Scheduling Theory. John Wiley Sons, New York, 165-227.