

Performance Analysis of *LPT* Algorithm for Jobs with Similar Sizes on Two Machines

Feng Wang¹, Rongheng Li^{1*}, Yunxia Zhou²

¹Key Laboratory of Computing and Stochastic Mathematics (Ministry of Education), College of Mathematics and Statistics, Hunan Normal University, Changsha Hunan

²College of Information Science and Engineering, Hunan Normal University, Changsha Hunan
Email: *lirongheng@hunnu.edu.cn

Received: Jun. 30th, 2019; accepted: Jul. 11th, 2019; published: Jul. 18th, 2019

Abstract

In this paper, *LPT* algorithm is considered for the scheduling problem in which the processing times of jobs are similar on two 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 if job list $L = \{J_1, J_2, \dots, J_n\}$ satisfies $p_j \in [1, r] (r \geq 1)$. Our result improves the best existing result. Furthermore, the ratio given here is the best. That means our result cannot be improved any more.

Keywords

Scheduling Problem, Identical Machine, *LPT* Algorithm, Worst Performance Ratio

工件具有相似加工时长时两台机上*LPT*算法的性能分析

王 凤¹, 李荣珩^{1*}, 周云霞²

¹湖南师范大学数学与统计学院, 计算与随机数学教育部重点实验室, 湖南 长沙

²湖南师范大学信息科学与工程学院, 湖南 长沙
Email: *lirongheng@hunnu.edu.cn

收稿日期: 2019年6月30日; 录用日期: 2019年7月11日; 发布日期: 2019年7月18日

*通讯作者。

文章引用: 王凤, 李荣珩, 周云霞. 工件具有相似加工时长时两台机上 *LPT* 算法的性能分析[J]. 计算机科学与应用, 2019, 9(7): 1309-1316. DOI: 10.12677/csa.2019.97147

摘要

本文研究了工件具有相似加工时长时2台同类型平行机上LPT算法的最坏性能比。目标函数是使所有机器的最大完工时间达到最小。若工件序列 $L = \{J_1, J_2, \dots, J_n\}$ 中的工件满足 $p_j \in [1, r]$ ($r \geq 1$)，证明了LPT算法的最坏性能比为 r 的分段线性函数，此结论改进了已有的最好结论，并且是不能再改进的最好结论。

关键词

排序问题, 平行机, LPT算法, 最坏性能比

Copyright © 2019 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

排序问题是组合优化的重要分支之一，在生产管理、计算机信息技术处理等方面具有广泛的应用背景。自从Graham [1]在1969年首次提出排序问题以来，这个问题一直是最活跃的研究领域之一。经典排序问题是将 n 个独立工件 J_1, J_2, \dots, J_n 分配到 m 台具有同样性能的机器 M_1, M_2, \dots, M_m 上加工，每一个工件只需在其中一台机器上加工一次就能完工，每台机器每次只能加工一个工件。工件 J_1, J_2, \dots, J_n 之间没有先后的依存关系，且 J_i 的加工时间为 p_i ，目标是使所有工件在最早时间内完成全部加工，也就是使得所有机器中最大完工时间达到最小。由于排序问题是NP-困难问题，我们目前只能研究多项式时间近似算法。LPT (Largest Processing time)算法是Graham [2]提出的著名近似算法。该算法首先将工件排序使其满足 $p_1 \geq p_2 \geq \dots \geq p_n$ ，然后依次安排每个工件，每次总是把尚未投入加工的工件中编号最小者(即加工时长最大的)送到最早空闲的机器上进行加工。LPT算法的最坏性能比是 $\frac{4}{3} - \frac{1}{3m}$ 。

在经典排序问题的算法性能比分析中是假设工件的加工时长是任意的，但实际情况中，工件的加工时长不会太小，也不会太大。为了更符合实际，也为了改进算法的性能比，促使研究工作者对工件加工时长做出适当假设。工件加工时长具有相似加工长度(即假设工件加工时长在区间 $[1, r]$ 内，其中 $r \geq 1$)是常用假设之一[3]-[8]，也有假设工件最大时长已知的[9]及假设所有工件加工时长总和是已知的[6]。

H. Kellerer [10]给出了在 m 台同类型平行机上当工件具有相似的加工时长时LPT算法的性能比的上界为

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

其中 $k \geq 3$, k 为整数, $\frac{p_1}{p_n} = 2 - \frac{\mu}{m}$, ($1 \leq \mu < m, \mu \in R$), m 表示机器数。当 $\mu \leq \frac{m+3}{4}$ ($k=3$) 时, $\frac{4m-\mu}{3m}$ 是紧的, 其它区间不是紧界。

本文考虑两台机情形, 即 $m=2$, 证明了 LPT 算法的最坏性能比为

$$R(2, LPT, r) = \begin{cases} \frac{7}{6}, & r \geq \frac{3}{2}, \\ \frac{kr+k+1}{2k+1}, & 1 + \frac{1}{2k} - \frac{2}{2k(2k+3)} \leq r < 1 + \frac{1}{2k}, \\ \frac{4k+7}{4k+6}, & 1 + \frac{1}{2k+2} \leq r < 1 + \frac{1}{2k} - \frac{2}{2k(2k+3)}, \\ 1 & r = 1. \end{cases}$$

其中 $k \geq 1$, k 为整数。

当 $\frac{11}{8} \leq r \leq \frac{3}{2}$ 时(相当于[10]里的 $\mu \in [1, \frac{5}{4}]$), 我们得到的性能比和[10]一样。当 $r < \frac{11}{8}$ 时, 我们得到的最坏性能比比[10]的更小。

2. 定义符号

在本文中我们引入下面记号:

- 1) $L = \{J_1, J_2, \dots, J_n\}$ 为一个工件序列, 其中每个工件 J_j 的加工长度为 $p_j, j=1, 2, \dots, n$;
- 2) $p_{i(j)}$ 表示第 i 台机器上的第 j 个工件的加工时长;
- 3) $L_i (i=1, 2, \dots, m)$ 表示在 LPT 算法下最后一个工件 J_n 被安排之前机器 M_i 上所安排的所有工件加工时长的总和。并且我们将机器重新编号, 使之满足 $L_1 \leq L_2 \leq \dots \leq L_m$;
- 4) 设 A 是任一算法, $C_{\max}^A(L)$ 表示工件集 L 在算法 A 下所有机器的最大完工时间;
- 5) OPT 表示最优算法。

定义 1: 算法 A 是一个近似算法, $C_{\max}^A(L)$ 和 $C_{\max}^{OPT}(L)$ 分别表示在 A 算法和最优算法下该工件序列的最大完工时间。定义

$$R(m, A) = \sup_L \frac{C_{\max}^A(L)}{C_{\max}^{OPT}(L)}$$

并称其为算法 A 的最坏性能比, 其中 $L = \{J_1, J_2, \dots, J_n\}$ 为任一符合条件的工件序列。

下面我们分析 LPT 算法在两台机上的最坏性能比。

3. 定理及其证明

对于任意工件序列 $L = \{J_1, J_2, \dots, J_n\}$, 易知

$$C_{\max}^{OPT}(L) \geq \frac{\sum_{j=1}^n p_j}{m} \quad (1)$$

其中 m 为机器台数。

在没有特别说明的情形下, 下面我们总是假设工件序列 $L = \{J_1, J_2, \dots, J_n\}$ 满足, $p_i \in [1, r], p_1 \geq p_2 \geq \dots \geq p_n$, 并且总是假设 $m=2$ 。

定理 1. 当 $m=2$ 时, LPT 算法的最坏性能比为:

$$R(2, LPT, r) = \begin{cases} \frac{7}{6}, & r \geq \frac{3}{2}, \\ \frac{kr+k+1}{2k+1}, & 1 + \frac{1}{2k} - \frac{2}{2k(2k+3)} \leq r < 1 + \frac{1}{2k}, \\ \frac{4k+7}{4k+6}, & 1 + \frac{1}{2k+2} \leq r \leq 1 + \frac{1}{2k} - \frac{2}{2k(2k+3)}, \\ 1 & r = 1. \end{cases}$$

其中 $k \geq 1$, k 为整数。

证明: $r \geq \frac{3}{2}$ 时的结论即是 Graham [2] 给出的 $R(2, LPT) = \frac{4}{3} - \frac{1}{3m}$ 当 $m=2$ 时的情形。当 $r=1$ 时, 所有工件的加工时长都一样, 显然 LPT 算法也是最优算法, 所以结论显然成立。下面讨论 $\frac{2k+3}{2k+2} \leq r < \frac{2k+1}{2k}$ 的情形。我们首先证明对任意满足条件的工件序列 L , 下面不等式成立:

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq \begin{cases} \frac{kr+k+1}{2k+1} & 1 + \frac{2k+1}{2k(2k+3)} \leq r < \frac{2k+1}{2k}, \\ \frac{4k+7}{4k+6} & \frac{2k+3}{2k+2} \leq r < 1 + \frac{2k+1}{2k(2k+3)}. \end{cases} \quad (2)$$

如果(2)不成立, 设 $L = \{J_1, J_2, \dots, J_n\}$ 是使(2)不成立且工件个数 n 最小的工件序列, 这样的工件序列我们称之为最小反例。对于最小反例, 易知最后一个工件 J_n 是最后完工的工件, 即 $C_{\max}^{LPT}(L) = L_1 + p_n$ 。所以当 $1 + \frac{2k+1}{2k(2k+3)} \leq r < \frac{2k+1}{2k}$ 时, 我们有

$$\begin{aligned} \frac{kr+k+1}{2k+1} &< \frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} = \frac{2(L_1 + p_n)}{2C_{\max}^{OPT}(L)} \\ &\leq \frac{L_1 + L_2 + 2p_n}{2C_{\max}^{OPT}(L)} = \frac{\sum_{j=1}^n p_j + p_n}{2C_{\max}^{OPT}(L)} \\ &\leq 1 + \frac{p_n}{2C_{\max}^{OPT}(L)} \end{aligned}$$

最后不等式由(1)得。即

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

当 $\frac{2k+3}{2k+2} \leq r < 1 + \frac{2k+1}{2k(2k+3)}$ 时, 我们同样可得(3)。因此, 对于最小反例 L , 其总工件个数 $n \leq 4k+4$ 。

下面我们令

$$S_i = \{j | J_j \text{ 在 } LPT \text{ 算法下安排在 } M_i \text{ 上并且 } j \neq n\}, i=1, 2; \quad S_i^* = \{j | J_j \text{ 在 } OPT \text{ 算法下安排在 } M_i \text{ 上}\}, i=1, 2。$$

$$\text{易知 } |S_1 \cup S_2| = n-1, |S_1^* \cup S_2^*| = n。$$

为了完成定理 1 的证明, 我们首先给出下面的几个引理。

引理 2. 对最小反例 L , 在 LPT 算法下, J_{2i-1} 和 J_{2i} 安排在不同机器上 ($i=1, 2, \dots, 2k, 2k+1$)。

证明: 假设不然, 设 i_0 是使 J_{2i-1} 和 J_{2i} 为第一对被 LPT 算法安排在同一台机器上的两个工件, 不失一般性, 假设都安排在 M_2 上, 那么在 J_{2i_0-1} 和 J_{2i_0} 安排之前, M_1, M_2 上的工件集 S'_1, S'_2 满足 $|S'_1| = |S'_2| = i_0 - 1$, 则由 LPT 算法规则下面不等式必然成立:

$$(i_0 - 1)r \geq \sum_{j \in S'_1} p_j \geq \sum_{j \in S'_2} p_j + p_{2i_0-1} \geq i_0$$

所以有 $r \geq \frac{i_0}{i_0 - 1} \geq 1 + \frac{1}{2k}$ 。这显然与 $r < 1 + \frac{1}{2k}$ 矛盾, 即引理得证。

由引理 2 可以得出, 在 LPT 算法下未安排工件 J_{4k+4} 时, 两台机器上的工件个数相差不会超过 1。

引理 3. 对最小反例 L , 若 $|S_1^*| \geq |S_2^*|$, 则 $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$ 和 $|S_2 \cap S_1^*| \geq |S_2 \cap S_2^*|$ 至少有一个成立。

证明: 若 $|S_i \cap S_1^*| < |S_i \cap S_2^*| (i=1,2)$ 成立, 则 $|(S_1 \cup S_2) \cap S_1^*| \leq |(S_1 \cup S_2) \cap S_2^*| + 2$ 成立。由定义我们有

$$|(S_1 \cup S_2) \cap S_i^*| = \begin{cases} |S_i^*| & n \notin S_i^* \\ |S_i^*| - 1 & n \in S_i^* \end{cases}$$

这样我们有 $|S_1^*| - 1 \leq |S_2^*| + 2 (n \in S_1^*)$ 或者 $|S_1^*| \leq |S_2^*| + 1 (n \in S_2^*)$, 此时都与假设 $|S_1^*| \geq |S_2^*|$ 矛盾。

引理 4. 设 $p, q \in \{1, 2\}$, 令 $\bar{q} = q \bmod 2 + 1$, 即 $q = 1$ 时 $\bar{q} = 2$, 而 $q = 2$ 时 $\bar{q} = 1$ 。若 $|S_p| \leq k_0$, $|S_q^*| \geq k_0 + 1$, $|S_p \cap S_q^*| \geq |S_p \cap S_{\bar{q}}^*|$ 且 $k_0 \leq 2k$, 则有

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq \frac{kr + k + 1}{2k + 1} \tag{4}$$

证明: 由 $C_{\max}^{LPT}(L) \leq \sum_{j \in S_p} p_j + p_n$ 和 $C_{\max}^{OPT}(L) \geq \sum_{j \in S_q^*} p_j$ 得

$$\begin{aligned} \frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} &\leq \frac{\sum_{j \in S_p} p_j + p_n}{\sum_{j \in S_q^*} p_j} = \frac{\sum_{j \in S_p \cap S_q^*} p_j + \sum_{j \in S_p \cap S_{\bar{q}}^*} p_j + p_n}{\sum_{j \in S_q^* \setminus S_p} p_j + \sum_{j \in S_p \cap S_q^*} p_j} \\ &\leq \frac{\sum_{j \in S_p \cap S_{\bar{q}}^*} p_j + |S_p \cap S_q^*| + 1}{\sum_{j \in S_q^* \setminus S_p} p_j + |S_p \cap S_q^*|} \leq \frac{|S_p \cap S_{\bar{q}}^*| r + |S_p \cap S_q^*| + 1}{|S_p \cap S_q^*| + |S_q^* \setminus S_p|} \\ &= \frac{|S_p \cap S_{\bar{q}}^*| r + |S_p \cap S_q^*| + |S_p \cap S_{\bar{q}}^*| r + |S_p \cap S_q^*| + 2}{2|S_q^*|} \\ &\leq \frac{|S_p \cap S_q^*| r + |S_p \cap S_{\bar{q}}^*| + |S_p \cap S_{\bar{q}}^*| r + |S_p \cap S_q^*| + 2}{2|S_q^*|} \\ &\leq \frac{|S_p| r + |S_{\bar{q}}| + 2}{2|S_q^*|} = \frac{k_0 r + k_0 + 2}{2k_0 + 2} \leq \frac{kr + k + 1}{2k + 1} \end{aligned}$$

上面第二个不等式是因为 $p_n \geq 1$ 和函数 $f(x) = \frac{a+x}{b+x} (a \geq b > 0)$ 在 $x \geq 0$ 里是 x 的减函数, 第四个不等

式由引理 3 得, 最后一个不等式是因为 $k_0 \leq 2k$ 和函数 $g(x, y) = \frac{(r+1)x+2}{2x+2}$ 是 $x \geq 0$ 的增函数。

引理 5. 设 $p, q \in \{1, 2\}$, 令 $\bar{q} = q \bmod 2 + 1$ 。若 $|S_p| \leq 2k_0 + 1$, $|S_q^*| \geq 2k_0 + 2$ 且 $k_0 \leq k$, 则有(4)。

证明: 与引理 4 的证明类似可得

$$\begin{aligned} \frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} &\leq \frac{\sum_{j \in S_p \cap S_q^*} P_j + \sum_{j \in S_p \cap S_q} P_j + P_n}{\sum_{j \in S_q^* \setminus S_p} P_j + \sum_{j \in S_p \cap S_q^*} P_j} \\ &\leq \frac{|S_p \cap S_q^*| r + |S_p \cap S_q^*| + 1}{|S_p \cap S_q^*| + |S_q^* \setminus S_p|} \\ &= \frac{|S_p \cap S_q^*| (r-1) + |S_p| + 1}{|S_q^*|} \\ &\leq \frac{k_0 r + k_0 + 2}{2k_0 + 2} \leq \frac{kr + k + 1}{2k + 1} \end{aligned}$$

上面最后一个不等式是因为函数 $g(x, y) = \frac{(r+1)x + y}{2x + y}$ 是 $x \geq 0$ 的增函数, 是 $y \geq 0$ 的减函数。

由(3), 我们下面分 $n = 2k' + 1, 2k' + 2 (k' \leq 2k)$ 和 $n = 4k + 3, 4k + 4$ 这 4 种情况证明(4)成立。

情况 1. $n = 2k' + 1, k' \leq 2k$ 。

由 $|S_1^*| + |S_2^*| = 2k' + 1$ 不妨设 $|S_1^*| \geq k' + 1$ 。由引理 2 得 $|S_1| = |S_2| = k'$, 由引理 3 知, $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$ 和 $|S_2 \cap S_1^*| \geq |S_2 \cap S_2^*|$ 至少有一个成立。若 $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$ 成立。则令 $p = 1, q = 1, k_0 = k'$, 由引理 4 得(4)成立。若 $|S_2 \cap S_1^*| \geq |S_2 \cap S_2^*|$ 成立。则令 $p = 2, q = 1, k_0 = k'$, 由引理 4 也得(4)成立。

情况 2. $n = 2k' + 2, k' \leq 2k$ 。

由引理 2 不妨设有 $|S_1| = k', |S_2| = k' + 1, |S_1^*| \geq k' + 1$ 。若 $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$, 则令

$p = 1, q = 1, k_0 = k'$, 由引理 4 得(4)成立。

若 $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$ 不成立, 此时若有 $|S_2^*| \geq k' + 1$, 令 $p = 1, q = 2, k_0 = k'$, 由引理 4 得(4)成立。否则有 $|S_2^*| \leq k'$, 因而 $|S_1^*| \geq k' + 2$ 。则由引理 3 知 $|S_2 \cap S_1^*| \geq |S_2 \cap S_2^*|$ 成立, 这时令 $p = 2, q = 1, k_0 = k' + 1$, 由引理 4 知当 $k' < 2k$ 时(4)成立。

若 $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$ 不成立, 且 $|S_2^*| \leq k'$ 和 $k' = 2k$ 成立, 令 $p = 1, q = 2, k_0 = k'$, 由引理 5 知(4)成立, 因为由 $|S_1 \cap S_1^*| \leq |S_1 \cap S_2^*|$ 可得 $|S_1 \cap S_1^*| \leq k$ 。

情况 3. $n = 4k + 3$ 。

不妨设 $|S_1^*| > |S_2^*|$ 。由引理 2 及(3)知 $|S_1^*| = 2k + 2$ 且 $|S_1| = |S_2| = 2k + 1$ 。由引理 3 知 $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$ 和 $|S_2 \cap S_1^*| \geq |S_2 \cap S_2^*|$ 中至少有一个成立。若 $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$ 成立, 则有 $|S_1 \cap S_1^*| \geq k + 1, |S_1 \cap S_2^*| \leq k$ 。令 $p = 1, q = 1, k_0 = k$, 则由引理 5 即得(4)成立。

若 $|S_1 \cap S_1^*| \geq |S_1 \cap S_2^*|$ 不成立, 则必有 $|S_2 \cap S_1^*| \geq |S_2 \cap S_2^*|$ 成立, 所以有 $|S_2 \cap S_1^*| \geq k + 1, |S_2 \cap S_2^*| \leq k$ 。令 $p = 2, q = 1, k_0 = k$ 则由引理 5 即得(4)成立。

情况 4. $n = 4k + 4$ 。

由(3)得 $|S_1^*| = |S_2^*| = 2k + 2$ 。由引理 2 可不妨设 $|S_1| = 2k + 1, |S_2| = 2k + 2$ 。

若 $|S_1 \cap S_1^*| \geq k + 1$ 成立, 则有 $|S_1 \cap S_2^*| \leq k$ 这时令 $p = 1, q = 1, k_0 = k$, 则由引理 5 即得(4)成立。由于 $|S_1 \cap (S_1^* \cup S_2^*)| = 2k + 1$, 若 $|S_1 \cap S_1^*| \geq k + 1$ 不成立, 则 $|S_1 \cap S_2^*| \geq k + 1$ 成立, 因而有 $|S_1 \cap S_1^*| \leq k$, 这时令 $p = 1, q = 2, k_0 = k$, 则由引理 5 也得(4)成立。

由上所述可知

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq \max \left\{ \frac{kr+k+1}{2k+1}, \frac{4k+7}{4k+6} \right\}$$

$$= \begin{cases} \frac{kr+k+1}{2k+1}, & 1 + \frac{2k+1}{2k(2k+3)} \leq r < \frac{2k+1}{2k} \\ \frac{4k+7}{4k+6}, & \frac{2k+3}{2k+2} \leq r < 1 + \frac{2k+1}{2k(2k+3)} \end{cases}$$

下面我们分别给出上面不等式等号成立的实例:

当 $1 + \frac{2k+1}{2k(2k+3)} \leq r \leq \frac{2k+1}{2k}$ 时, 我们令 $L = \{J_1, J_2, \dots, J_{4k+1}\}$, 其相应的加工时长为 $\{r, r, \dots, r, 1, 1, \dots, 1\}$, 其中时长为 r 和 1 的工件分别有 $2k$ 和 $2k+1$ 个。在 LPT 算法下, M_1 上安排时长为 r 和 1 的工件各 $k, k+1$ 个, M_2 上安排时长为 r 和 1 的工件各 k 个, 故 $C_{\max}^{LPT}(L) = kr+k+1$ 。

在 OPT 算法下, M_1 上安排时长为 r 的工件 $2k$ 个, M_2 上安排时长为 1 的工件 $2k+1$ 个, 故 $C_{\max}^{OPT}(L) = 2k+1$, 即

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} \leq \frac{kr+k+1}{2k+1}$$

当 $\frac{2k+3}{2k+2} \leq r < 1 + \frac{2k+1}{2k(2k+3)}$ 时, 我们分别给出 k 为奇数和偶数时紧界的实例:

当 k 为奇数时, 令 $L = \{J_1, J_2, \dots, J_{4k+5}\}$, 其相应加工时长为 $\{r, r, \dots, r, r', r', \dots, r', 1, 1, \dots, 1\}$ 时长为 r 和 r' 的各 $k+1$ 个, 时长为 1 的有 $2k+3$ 个, 并且满足

$$1 < r' < r, r+r' = \frac{2k+3}{k+1}$$

在 LPT 算法下, M_1 上安排时长为 r 和 r' 的工件各有 $\frac{k+1}{2}$ 个, 安排 $k+2$ 个时长为 1 的工件。 M_2 上安排时长分别为 r 和 r' 的工件各 $\frac{k+1}{2}$ 个, 时长为 1 的工件 $k+1$ 个。则

$$C_{\max}^{LPT}(L) = \frac{(k+1)(r+r')}{2} + k+2$$

在 OPT 算法下, M_1 上安排时长为 r 和 r' 的工件各 $k+1$ 个, M_2 上安排 $2k+3$ 个时长为 1 的工件, 则 $C_{\max}^{OPT}(L) = 2k+3$, 故

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} = \frac{4k+7}{4k+6}$$

当 k 为偶数时, 令 $L = \{J_1, J_2, \dots, J_{4k+5}\}$, 其相应加工时长为 $\{r, r, \dots, r, r', r', \dots, r', r'', r'', 1, 1, \dots, 1\}$ 长分别为 r 和 r' 的各 k 个, 时长为 r'' 的工件有 2 个, 时长为 1 的有 $2k+3$ 个, 并且满足

$$1 < r'' < r' < r, k(r+r') + 2r'' = 2k+3$$

在 LPT 算法下, M_1 上安排时长为 r 和 r' 的工件各 $\frac{k}{2}$ 个, 安排 1 个时长为 r'' 的工件和 $k+2$ 个时长为 1 的工件。 M_2 上安排时长分别为 r 和 r' 的工件各 $\frac{k}{2}$ 个和 1 个时长为 r'' 的工件以及 $k+1$ 个时长为 1 的工件。则

$$C_{\max}^{LPT}(L) = \frac{k(r+r')}{2} + r'' + k + 2$$

在 OPT 算法下, M_1 上安排时长为 r 和 r' 的工件各 k 个, 以及 2 个 r'' , 在 M_2 上安排 $2k+3$ 个时长为 1 的工件, 则 $C_{\max}^{OPT}(L) = 2k+3$ 故:

$$\frac{C_{\max}^{LPT}(L)}{C_{\max}^{OPT}(L)} = \frac{4k+7}{4k+6}$$

综上所述, 定理 1 得证。

4. 小结

本文主要研究了工件具有相似加工时长时, LPT 算法在同型平行机上的最坏性能比。证明了当 $m=2$ 时对于任意的工件序列 L , LPT 算法的最坏性能比为

$$R(2, LPT, r) = \begin{cases} \frac{7}{6}, & r \geq \frac{3}{2}, \\ \frac{kr+k+1}{2k+1}, & 1 + \frac{1}{2k} - \frac{2}{2k(2k+3)} \leq r < 1 + \frac{1}{2k}, \\ \frac{4k+7}{4k+6}, & 1 + \frac{1}{2k+2} \leq r \leq 1 + \frac{1}{2k} - \frac{2}{2k(2k+3)}, \\ 1 & r = 1. \end{cases}$$

其中 $k \geq 1$, k 为整数。

本文成果改进了已有的最好结果, 并且这是不能改进的最好结果。本文的分析比较复杂, 更有趣的问题是能否改进一般性 m 台机时的最坏性能比。

基金项目

本文得到国家自然科学基金(编号 11471110)和湖南省教育厅重点课题(编号 16A126)资助。

参考文献

- [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] Graham, R.L. (1976) Bounds on the Performance of Scheduling Algorithms. In: Coffmann, E.G., Ed., *Computer and Job-Shop Scheduling Theory*, John Wiley Sons, New York, 165-227.
- [3] 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>
- [4] He, Y. and Dösa, 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>
- [5] He, Y. and Zhang, G. (1999) Semi On-Line Scheduling on Two Identical Machines. *Computing*, **62**, 179-187. <https://doi.org/10.1007/s006070050020>
- [6] 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](https://doi.org/10.1016/S0167-6377(98)00005-4)
- [7] 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>
- [8] 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>
- [9] He, Y. and Zhang, G. (1999) Semi On-Line Scheduling on Two Identical Machines. *Computing*, **62**, 179-187. <https://doi.org/10.1007/s006070050020>
- [10] 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>

知网检索的两种方式：

1. 打开知网首页：<http://cnki.net/>，点击页面中“外文资源总库 CNKI SCHOLAR”，跳转至：<http://scholar.cnki.net/new>，搜索框内直接输入文章标题，即可查询；
或点击“高级检索”，下拉列表框选择：[ISSN]，输入期刊 ISSN：2161-8801，即可查询。
2. 通过知网首页 <http://cnki.net/>顶部“旧版入口”进入知网旧版：<http://www.cnki.net/old/>，左侧选择“国际文献总库”进入，搜索框直接输入文章标题，即可查询。

投稿请点击：<http://www.hanspub.org/Submission.aspx>

期刊邮箱：csa@hanspub.org