

带准备时间的任务单机学习效应排序问题

王吉波^{*1,2}, 刘 璐¹

(1. 沈阳航空航天大学 经济与管理学院, 辽宁 沈阳 110136;

2. 西安交通大学 机械制造系统工程国家重点实验室, 陕西 西安 710054)

摘要: 具有学习效应的任务的加工时间和带有准备时间的任务问题是排序论中的重要研究内容, 它们对任务的完工时间有重要影响. 研究了具有学习效应且带有准备时间的任务单机排序问题, 其中学习效应指的是任务的实际加工时间是该已经排好的任务对数加工时间的递减函数, 目标函数为最小化总完工时间. 这个问题是 NP-难问题, 用分支定界法给出了此问题的最优解, 为了提高分支定界法的运行效率, 同时给出了一个启发式算法、几个优势性质和两个下界. 计算结果表明分支定界法和启发式算法求解此问题非常有效.

关键词: 排序; 单机; 学习效应; 准备时间

中图分类号: O223; C934 **文献标志码:** A

0 引言

带有学习效应的排序问题目前越来越受到人们的关注. 在经典的排序问题中, 任务的加工时间通常被认为是确定的常数. 然而在实际生活中, 任务的加工时间却会随着机器磨合度的增加、技术工人熟练度的提升等因素而缩短, 这个现象被称为学习效应. 从 Biskup^[1] 首次提出具有学习效应的排序问题后, 就不断有大量学者在这方面投入研究. 王吉波等^[2] 研究了单机排序中的一般学习效应——Dejong 学习效应, 并给出了在这种情况下下的单机最大完工时间问题、总完工时间问题及两类多目标问题的多项式时间算法. 文献[3]研究了具有学习效应的流水作业排序问题, 对最大完工时间问题和总完工时间问题给出了一些结果. 文献[4]研究了在学习效应下的两个单机排序问题, 把加权加工时间优先 (WSPT) 规则及最小工期优先 (EDD) 规则作为加权完工时间及最大延迟排序问题的启发式算法, 同时分析了这两个算

法的最坏情况界. 此外, Wu 等^[5] 利用启发式算法研究在学习效应下的最大延误问题. Wang 等^[6] 还研究了在学习效应下的单机排序问题, 这里的学习效应指的是任务的加工时间与已经排好的任务的加工时间有关. 他们对加权总完工时间、最大延迟及总误工数问题分别给出了启发式算法, 并分析了这些算法的最坏情况界. Cheng 等^[7] 提出了具有对数加工时间学习效应的排序模型, 即任务的加工时间与已经排好的任务的对数加工时间有关. 他们对几个正则目标函数分别给出了一些结论. Wu 等^[8] 研究了具有学习效应的单机排序问题, 其中每个任务都有准备时间. 对最大完工时间问题, 他们给出了分支定界算法. 文献[9] 研究了任务加工时间具有一般学习效应的流水作业排序问题, 对最大完工时间、总完工时间、加权总完工时间、加权总折扣完工时间及完工时间平方和问题分别给出了启发式算法, 并分析了这些算法的最坏情况界. Wu 等^[10] 对加权总完工时间问题, 给出了遗传算法. Wang 等^[11] 研究了具有学习效

收稿日期: 2012-12-25; 修回日期: 2013-09-15.

基金项目: 国家自然科学基金资助项目(11001181); 辽宁省高等学校优秀人才支持计划资助项目(LJQ2011014); 西安交通大学机械制造系统工程国家重点实验室开放课题资助项目(skllms201306).

作者简介: 王吉波^{*} (1975-), 男, 大连理工大学 2005 届博士, 教授, E-mail: wangjibo75@163.com.

应和恶化效应的任务加工时间流水作业排序问题,机器限定为两台,目标函数为极小化最大完工时间.对此问题,他们给出了分支定界算法.其他最新的关于学习效应的文章可参考文献[12-16].关于具有学习效应的排序问题方面的综述,可参考文献[17].

与此同时,研究带准备时间的学习效应模型的学者还不是太多,但是在实际生活中,任务的准备时间在很多情况下都是必不可少的,文献[8]和[10]研究了如下学习效应模型,即任务 J_j 被放在第 β 个位置进行加工,则它的实际加工时间 $p_{j\beta} = p_j \left(1 - \sum_{l=1}^{\beta-1} p_{[l]} / \sum_{l=1}^n p_l\right)^a$,其中 $a \geq 1$ 为学习因子.然而此模型有不足之处,即任务的加工时间下降得太快,所以 Cheng 等^[7]提出了具有对数加工时间学习效应的排序模型.本文在其基础上引入准备时间因素,对总完工时间问题进行讨论.没有学习效应的此问题是 NP-难的,有学习效应的问题更是 NP-难的,本文给出几条优势性质、两个下界及一个启发式算法,并提出分支定界法.

1 问题描述

设有 n 个任务 $\{J_1, J_2, \dots, J_n\}$ 要在同一台机器上加工.机器同一时间只能加工一个任务,并且任务在加工过程中不允许中断.每个任务 J_j 都有一个基本的加工时间 p_j 和一个准备时间 r_j .同文献[7]中一样,如果任务被放在第 β 个位置进行加工,则任务 J_j 的实际加工时间为 $p_{j\beta} = p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a$,其中 $\ln p_j \geq 1, a < 0$ 为学习因子, $p_{[j]}$ 表示排在第 j 个位置任务的正常加工时间.令 $C_j(S)$ 表示在排序 S 中任务 J_j 的完工时间,目标是寻找一个最优排序使总完工时间 $\sum C_j$ 最小.

2 优势性质

首先,本文将提出一些基于相邻任务两两交换方法的优势性质.设 $S = (\pi, J_i, J_j, \pi')$, $S' =$

(π, J_j, J_i, π') ,其中 π 和 π' 是部分排序.其次,假设 π 中有 $\beta-1$ 个任务,那么在 S 中 J_i 和 J_j 分别是第 β 个位置和第 $\beta+1$ 个位置加工的任务,在 S' 中 J_j 和 J_i 分别是第 β 个位置和第 $\beta+1$ 个位置加工的任务.为了表明排序 S 优于 S' ,需要证明 $C_j(S) < C_i(S')$ 以及 $C_i(S) + C_j(S) < C_i(S') + C_j(S')$.此外,本文假设 A 为部分排序 π 中最后一个任务的完工时间.

引理 1^[7] 如果 $\lambda > 1, 0 < \partial \leq 1, k \geq 1$,则 $(\lambda-1) + (1 + \partial \ln \lambda + \partial k)^a - \lambda(1 + \partial k)^a > 0$.

性质 1 如果 $r_j > A > r_i, A + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a > r_j$,且 $p_i < p_j$,则 S 优于 S' .

证明 因为 $r_j > A > r_i, A + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a > r_j$,任务 J_i 和 J_j 在排序 S 和 S' 中的完工时间分别为

$$C_i(S) = A + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a$$

$$C_j(S) = A + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a +$$

$$p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_i\right)^a$$

$$C_j(S') = r_j + p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a$$

$$C_i(S') = r_j + p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a +$$

$$p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_j\right)^a$$

又因为 $p_i < p_j$,所以

$$C_i(S') - C_j(S) = (r_j - A) + p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a +$$

$$p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_j\right)^a -$$

$$p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a -$$

$$p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_i\right)^a =$$

$$(r_j - A) + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a \times$$

$$\left[\frac{p_j}{p_i} + \left(1 + \ln p_j\right) / \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)\right]^a - 1 - \frac{p_j}{p_i} \left(1 + \ln p_i\right) / \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a \Big] \tag{1}$$

令 $\lambda = \frac{p_j}{p_i} > 1, 0 < \partial = \frac{1}{1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}} \leq 1, k =$

$\ln p_i \geq 1$, 由式(1) 和引理 1, 可得

$$C_i(S') - C_j(S) = (r_j - A) + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a \times \left[(\lambda - 1) + (1 + \partial \ln \lambda + \partial k)^a - \lambda (1 + \partial k)^a\right] > 0$$

$$\begin{aligned} & [C_j(S') + C_i(S')] - [C_i(S) + C_j(S)] = \\ & 2(r_j - A) + 2p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_j\right)^a - 2p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a - \\ & p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_i\right)^a = \end{aligned}$$

$$\begin{aligned} & 2(r_j - A) + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a \times \\ & \left[2 \frac{p_j}{p_i} + \left(1 + \ln p_j\right) / \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)\right]^a - \\ & 2 - \frac{p_j}{p_i} \left(1 + \ln p_i\right) / \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a \Big] = \\ & 2(r_j - A) + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a [2(\lambda - 1) + \\ & (1 + \partial \ln \lambda + \partial k)^a - \lambda (1 + \partial k)^a] > 0 \tag{2} \end{aligned}$$

从式(1) 和(2) 可得排序 S 优于 S' . 性质 1 证毕.

性质 2~6 的证明与性质 1 的证明类似, 在此不一一证明, 直接给出.

性质 2 如果 $r_j > r_i > A, r_i + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a > r_j$, 且 $p_i < p_j$, 则 S 优于 S' .

性质 3 如 果 $\max\{A, r_i\} + p_i \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a < r_j$, 则 S 优于 S' .

性质 4 如果 $r_i > A > r_j, A + p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a > r_i, p_i < p_j$ 且

$$p_j \left[\left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a - \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_i\right)^a\right] > (r_i - A) + p_i \left[\left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a - \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_j\right)^a\right]$$

则 S 优于 S' .

性质 5 如果 $A > \max\{r_i, r_j\}$, 且 $p_i < p_j$, 则排序 S 优于 S' .

性质 6 如果 $r_i > r_j > A, r_j + p_j \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a > r_i, p_i < p_j$ 且

$$p_j \left[\left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a - \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_i\right)^a\right] > (r_i - r_j) + p_i \left[\left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]}\right)^a - \left(1 + \sum_{l=1}^{\beta-1} \ln p_{[l]} + \ln p_j\right)^a\right]$$

则 S 优于 S' .

3 下 界

分支定界法在很大程度上依赖于下界的确定来减少分支, 以提高搜索效率. 在这个部分, 本文将提出两个下界以达到快速剪枝的目的. 假设 S_p 为部分排序, 其中前 k 项任务的排序是确定的. 此外, 令 S 为包含 S_p 的完整排序. 由定义可知, 第 $k + 1$ 个位置任务的完工时间为

$$C_{[k+1]}(S) = \max\{C_{[k]}(S), r_{[k+1]}\} + p_{[k+1]} \times \left(1 + \sum_{l=1}^k \ln p_{[l]}\right)^a \geq C_{[k]}(S) + p_{[k+1]} \left(1 + \sum_{l=1}^k \ln p_{[l]}\right)^a$$

类似地, 第 $k + \delta$ 个位置的任务的完工时间为

$$C_{[k+\delta]}(S) \geq C_{[k]}(S) + \sum_{j=1}^{\delta} p_{[k+j]} \times \left(1 + \sum_{l=1}^{k+j-1} \ln p_{[l]}\right)^a; 1 \leq \delta \leq n - k$$

因此, 排序 S 的总完工时间为

$$C_1(S) = \sum_{j=1}^k C_{[j]}(S) + \sum_{j=1}^{n-k} C_{[k+j]}(S) \geq \sum_{j=1}^k C_{[j]}(S) + \sum_{j=1}^{n-k} \left[(n-k-j+1) \times p_{[k+j]} \left(1 + \sum_{l=1}^{k+j-1} \ln p_{[l]} \right)^a \right] + (n-k)C_{[k]}(S) \quad (3)$$

从式(3)中可以发现,不等式右边第一项和最后一项是已知的(因为前 k 项任务的排序是确定的),所以排序 S 的总完工时间的下界可以通过极小化不等式右边第二项来获得. $\sum_{j=1}^{n-k} (n-k-j+1) p_{[k+j]} \left(1 + \sum_{l=1}^{k+j-1} \ln p_{[l]} \right)^a$ 是 S 中除去 S_p 的部分任务的总完工时间,此项可根据最小加工时间优先(SPT)规则将未排序的任务进行排序^[7],以此来达到极小化的目的.因此,第一个下界为

$$B_{11} = \sum_{j=1}^k C_{[j]}(S) + \sum_{j=1}^{n-k} \left[(n-k-j+1) \times p_{(k+j)} \left(1 + \sum_{l=1}^k \ln p_{[l]} + \sum_{l=k+1}^{k+j-1} \ln p_{(l)} \right)^a \right] + (n-k)C_{[k]}(S) \quad (4)$$

其中那些仍未排序的任务是按照任务的正常加工时间不减的顺序进行排列的,即 $p_{(k+1)} \leq p_{(k+2)} \leq \dots \leq p_{(n)}$.

另一方面,如果任务的准备时间太长则会导致式(4)中所示下界太小.为了解决这个问题,得出了一个考虑准备时间的第二个下界.在这种情况下,第 $k+1$ 个位置任务的完工时间为

$$C_{[k+1]}(S) = \max \{ C_{[k]}(S), r_{[k+1]} \} + p_{[k+1]} \times \left(1 + \sum_{l=1}^k \ln p_{[l]} \right)^a \geq r_{[k+1]} + p_{[k+1]} \left(1 + \sum_{l=1}^k \ln p_{[l]} \right)^a$$

类似地,第 $k+\alpha$ 个位置任务的完工时间为

$$C_{[k+\alpha]}(S) \geq r_{[k+\alpha]} + p_{[k+\alpha]} \left(1 + \sum_{l=1}^{k+\alpha-1} \ln p_{[l]} \right)^a; \quad 1 \leq \alpha \leq n-k$$

因此,排序 S 的总完工时间为

$$C_1(S) = \sum_{j=1}^k C_{[j]}(S) + \sum_{j=1}^{n-k} C_{[k+j]}(S) \geq$$

$$\sum_{j=1}^k C_{[j]}(S) + \sum_{j=1}^{n-k} r_{[k+j]} + \sum_{j=1}^{n-k} p_{[k+j]} \left(1 + \sum_{l=1}^{k+j-1} \ln p_{[l]} \right)^a \quad (5)$$

从式(5)中可以看到,不等式右边的前两项都是已知的,所以排序 S 的总完工时间的下界可以通过极小化不等式右边第三项来获得. $\sum_{j=1}^{n-k} p_{[k+j]} \left(1 + \sum_{l=1}^{k+j-1} \ln p_{[l]} \right)^a$ 是 S 中除去 S_p 的部分任务的最大完工时间,此项可以根据 SPT 规则将未排序的任务进行排序^[7],以此来达到极小化的目的.因此,第二个下界为

$$B_{12} = \sum_{j=1}^k C_{[j]}(S) + \sum_{j=1}^{n-k} \left[r_{[k+j]} + p_{(k+j)} \left(1 + \sum_{l=1}^k \ln p_{[l]} + \sum_{l=k+1}^{k+j-1} \ln p_{(l)} \right)^a \right] \quad (6)$$

为了使目标函数的下界更严谨,取式(4)和(6)中的较大者作为部分排序 S_p 的下界,即

$$B_1 = \max \{ B_{11}, B_{12} \}$$

下面举例来说明如何用分支定界法来求解此问题.

例 1 $n = 4$, 它们的加工时间向量 $p = (3 \ 18 \ 20 \ 10)$, 准备时间向量 $r = (35 \ 36 \ 57 \ 68)$, 学习因子 $a = -0.322$, 则根据上文提到的计算下界的方法可以得到分支过程,如图 1 所示.其中工件 J_1 固定的下界计算过程如下:

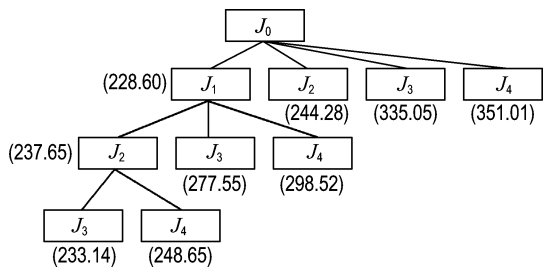


图 1 分支定界法搜索树

Fig. 1 Search tree of branch-and-bound algorithm

由于工件 J_1 固定,根据式(4)及式(6)给出

的计算下界方法,将数值代入得

$$B_{11} = C_1(S) + \sum_{j=1}^3 (4-j)p_{(1+j)} \left(1 + \ln p_1 + \sum_{l=2}^j \ln p_{(l)} \right)^a + 3C_1(S) = (r_1 + p_1) + 3p_4(1 + \ln p_1)^a + 2p_2(1 + \ln p_1 + \ln p_4)^a + p_3(1 + \ln p_1 + \ln p_4 + \ln p_2)^a + 3(r_1 + p_1) = 38 + 3 \times 10 \times (1 + \ln 3)^{-0.322} + 2 \times 18 \times (1 + \ln 3 + \ln 10)^{-0.322} + 20 \times (1 + \ln 3 + \ln 10 + \ln 18)^{-0.322} + 3 \times 38 = 208.52$$

$$B_{12} = C_1(S) + \sum_{j=1}^3 r_{[1+j]} + \sum_{j=1}^3 p_{(1+j)} \left(1 + \ln p_1 + \sum_{l=2}^j \ln p_{(l)} \right)^a = (r_1 + p_1) + (r_4 + r_2 + r_3) + p_4(1 + \ln p_1)^a + p_2(1 + \ln p_1 + \ln p_4)^a + p_3(1 + \ln p_1 + \ln p_4 + \ln p_2)^a = 38 + (68 + 36 + 57) + 10(1 + \ln 3)^{-0.322} + 18(1 + \ln 3 + \ln 10)^{-0.322} + 20 \times (1 + \ln 3 + \ln 10 + \ln 18)^{-0.322} = 228.60$$

因为 $B_1 = \max\{B_{11}, B_{12}\}$, 所以 $B_1 = 228.60$, 即工件 J_1 固定的下界为 228.60. 其余下界的计算过程类似, 在此不再一一列出. 从图 1 可以看出, 最优排序为 $[T_1, T_2, T_3, T_4]$, 最优值为 233.14.

4 启发式算法及数值仿真

为了提高分支定界法的搜索效率, 下面给出一个启发式算法来确定上界, 如此, 在分支过程中, 如有下界大于这个上界的分支可以直接进行剪枝.

启发式算法 1:

步骤 1 按工件的准备时间非减顺序排列, 即 $r_1 \leq r_2 \leq \dots \leq r_n$;

步骤 2 按工件的正常加工时间非减顺序排列, 即 $p_1 \leq p_2 \leq \dots \leq p_n$;

步骤 3 计算工件准备时间与其自身的正常加工时间的和, 然后对这个和进行非减顺序排列,

即 $r_1 + p_1 \leq r_2 + p_2 \leq \dots \leq p_n + r_n$;

步骤 4 从步骤 1 到步骤 3 中选取总完工时间最小的解, 得到一个初始排序 S_0 ;

步骤 5 置 $k = 1$;

步骤 6 置 $i = k + 1$;

步骤 7 通过交换第 k 和第 i 个工件得到新的排序 S_1 , 如果排序 S_1 的总完工时间小于 S_0 , 则用 S_1 代替 S_0 ;

步骤 8 如果 $i < n$, 那么置 $i = i + 1$, 转步骤 7.

步骤 9 如果 $k < n - 1$, 那么置 $k = k + 1$, 转步骤 6; 否则, 停止.

下面通过对前面提到的分支定界法和启发式算法进行数值仿真, 来测试算法的有效性. 通过 C# 语言编程, 在个人计算机 (硬件环境: AMD AthlonII X3 445 处理器, 2 GB 内存; 软件环境: SharpDevelop 2.0) 上运行, 选取 4 个不同工件数 $n=10, 11, 12, 13$, 对于不同的工件数 n , 再随机选取 10 组数, 其中工件的加工时间 p_j 在 $[3, 100]$ 随机产生, 工件的准备时间 r_j 在 $[1, 50]$ 随机产生, 学习效应因子 a 分别取 $-0.1, -0.5, -0.9$. 对于分支定界法, 分别给出平均和最大计算时间 (单位 ms)、平均和最大搜索节点数 2 个指标. 对于启发式算法 1, 给出启发式算法得到的解与最优解的平均相对误差 $\left(\left(\sum C_j(H) - \sum C_j(S^*) \right) / \sum C_j(S^*) \right)$, 其中 H 代表启发式算法 1 得到的排序, S^* 代表分支定界法得到的最优排序) 和最大相对误差, 由于启发式算法的计算时间非常短, 这里省略, 具体结果见表 1. 从表 1 能够看到分支定界法比较有效, 对 13 个工件的最大计算时间为 4 103 312 ms. 再有, 启发式算法 1 的结果非常有效, 大部分实例产生最优排序, 最大相对误差为 2.2%.

表 1 计算结果

Tab. 1 Computational results

<i>n</i>	<i>a</i>	分支定界法计算时间/ms		分支定界法搜索节点数		启发式算法 1 的相对误差	
		平均值	最大值	平均值	最大值	平均值	最大值
10	−0.1	1 678.100	3 235	82 199.500	170 880	0	0
	−0.5	1 084.400	1 813	50 652.500	86 346	0	0
	−0.9	1 146.900	2 531	56 075.700	134 012	0.002	0.022
11	−0.1	10 209.400	19 140	401 618.700	758 630	0	0
	−0.5	14 726.500	44 187	610 773.300	1 920 984	0.003	0.016
	−0.9	10 773.500	21 500	423 804.900	913 644	0.003	0.017
12	−0.1	56 751.500	111 297	1 799 771.200	3 628 691	0	0
	−0.5	85 731.300	221 000	2 757 305.500	7 373 708	0	0.001
	−0.9	126 010.900	428 625	4 261 579.400	15 831 196	0.006	0.020
13	−0.1	1 501 273.250	4 103 312	36 561 806.600	109 764 508	0	0
	−0.5	876 928.000	2 382 031	21 083 068.200	60 733 369	0.003	0.016
	−0.9	565 781.200	1 098 093	13 114 934.800	25 875 798	0.003	0.011

5 结 语

现实生活中,有很多任务具有准备时间且有些任务的准备时间足以影响整个加工过程的进行,所以合理地考虑准备时间的影响是很有必要的.本文研究了一个有关对数学习效应且带有准备时间的单机排序问题,目标函数为总完工时间.此问题是 NP-难的,为了找出此问题的最优排序,文中给出了几个优势性质、一个启发式算法以及两个下界来减少搜索范围.对于其他情况下带有准备时间的学习效应排序问题,比如多机问题、其他目标函数问题等,是以后要研究的方向.

参考文献：

[1] Biskup D. Single-machine scheduling with learning considerations [J]. **European Journal of Operational Research**, 1999, **115**(1):173-178.

[2] 王吉波,王明征,夏尊铨. 具有一般学习效应的单机排序问题[J]. 数学研究与评论, 2005, **25**(4):642-646.

WANG Ji-bo, WANG Ming-zheng, XIA Zun-quan. Single machine scheduling problems with general learning effect [J]. **Journal of Mathematical Research and Exposition**, 2005, **25**(4):642-646. (in Chinese)

[3] WANG Ji-bo, XIA Zun-quan. Flow shop

scheduling with a learning effect [J]. **Journal of the Operational Research Society**, 2005, **56**(11):1325-1330.

[4] 王吉波,马 力,王利岩,等. 具有学习效应的两个单机排序问题[J]. 大连理工大学学报, 2008, **48**(6):932-936.

WANG Ji-bo, MA Li, WANG Li-yan, *et al.* Two single machine scheduling problems with a learning effect [J]. **Journal of Dalian University of Technology**, 2008, **48**(6):932-936. (in Chinese)

[5] Wu C C, Lee W C, Chen T. Heuristic algorithms for solving the maximum lateness scheduling problem with learning considerations [J]. **Computers and Industrial Engineering**, 2007, **52**(1):124-132.

[6] WANG Ji-bo, Ng C T, Cheng T C E, *et al.* Single machine scheduling with a time-dependent learning effect [J]. **International Journal of Production Economics**, 2008, **111**(2):802-811.

[7] Cheng T C E, Lai P J, Wu C C, *et al.* Single-machine scheduling with sum-of-logarithm-processing-times-based learning considerations [J]. **Information Sciences**, 2009, **179**(18):3127-3135.

[8] Wu C C, Liu C L. Minimizing the makespan on a single machine with learning and unequal release times [J]. **Computers and Industrial Engineering**,

- 2010, **59**(3):419-424.
- [9] WANG Ji-bo, WANG Ming-zheng. Worst-case behavior of simple sequencing rules in flow shop scheduling with general position-dependent learning effects [J]. **Annals of Operations Research**, 2011, **191**(1):155-169.
- [10] Wu C C, Hsu P H, Chen J C, *et al.* Genetic algorithm for minimizing the total weighted completion time scheduling problem with learning and release times [J]. **Computers and Operations Research**, 2011, **38**(7):1025-1034.
- [11] WANG Ji-bo, Ji P, Cheng T C E, *et al.* Minimizing makespan in a two-machine flow shop with effects of deterioration and learning [J]. **Optimization Letters**, 2012, **6**(7):1393-1409.
- [12] LU Yuan-yuan, WEI Cai-min, WANG Ji-bo. Several single-machine scheduling problems with general learning effects [J]. **Applied Mathematical Modelling**, 2012, **36**(11):5650-5656.
- [13] Ahmadizar F, Hosseini L. Bi-criteria single machine scheduling with a time-dependent learning effect and release times [J]. **Applied Mathematical Modelling**, 2012, **36**(12):6203-6214.
- [14] ZHANG Xin-gong, YAN Guang-le, HUANG Wan-zhen, *et al.* A note on machine scheduling with sum-of-logarithm-processing-time-based and position-based learning effects [J]. **Information Sciences**, 2012, **187**:298-304.
- [15] WANG Ji-bo, WANG Jian-jun. Single-machine scheduling with precedence constraints and position-dependent processing times [J]. **Applied Mathematical Modelling**, 2013, **37**(3):649-658.
- [16] WANG Ji-bo, Hsu C J, YANG D L. Single machine scheduling problems with effects of exponential learning and general deterioration [J]. **Applied Mathematical Modelling**, 2013, **37**(2):2293-2299.
- [17] Biskup D. A state-of the-art review on scheduling effects [J]. **European Journal of Operational Research**, 2008, **188**(2):315-329.

Single-machine learning effect scheduling jobs with release time

WANG Ji-bo^{*1,2}, LIU Lu¹

(1. School of Economics and Management, Shenyang Aerospace University, Shenyang 110136, China;

2. State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Xi'an 710054, China)

Abstract: The processing time of jobs with a learning effect and the jobs with release time are the important research content in scheduling, which have important effects on the completion time of jobs. A single-machine learning effect scheduling problem with job release time is considered, where the learning effect means that the actual processing time of a job is a decreasing function of total logarithm normal processing time of jobs in front of it in the sequence, the objective function is to minimize the total completion time. This problem is well-known NP-hard, and a branch-and-bound algorithm is proposed to solve the problem. A heuristic algorithm, several dominance properties and two lower bounds are derived to speed up the elimination process of the branch-and-bound algorithm. Computational results show that the proposed heuristic algorithm and the branch-and-bound algorithm can perform effectively and efficiently.

Key words: scheduling; single-machine; learning effect; release time