Q6-51 9

# 并行加工设备组生产调度的一般模型及算法*

# A General Model and Its Algorithms on the Scheduling Problems of the Parallel Processors

黄德才 徐宗俊 杨 丹 胡立德 王时龙
Huang Decai Xu Zongjun Yang Dan Hu Lide Wang Shilong
(重庆大学系统工程及应用数学系；机械工程一系，重庆，630044)

A **摘 要** 给出了一个描述并行加工设备组生产调度问题的一般模型及两个启发式算法(极大消去法和ELPT方法)，对ELPT方法，另提供了一个误差分析结果，对极大消去法给出了一个数值计算实例。

**关键词** 并行加工设备组；调度问题；极小化 makespan 生产管理
**中国图书资料分类法分类号** N36

**ABSTRACT** In this paper, a general model on the scheduling problems of the parallel processors was presented. It made models in [1~5] a spacial case. Meanwhile, Two heuristie algorithms, maximum deleting algorithm and ELPT algorithm, were suggested. A worst-case analysis theorem on ELPT algorithm was given too. Finally, An illustrative example was also presented.

**KEYWORDS** parallel processors; scheduling problem; minimzing makespan

## 0 引 言

并行加工设备组的生产调度问题是生产调度理论中的基本问题之一。对于由 $m$ 台相同的加工设备构成的并行加工设备组的生产调度问题，已有许多研究成果[1~5]，该问题可叙述为：

设有 $m$ 台相同的加工设备 $M_1, M_2, \cdots, M_m$ 构成一个并行加工设备组。给定 $n$ 项独立的加工任务，它们的加工时间分别是 $p_1, p_2, \cdots, p_n$(正整数)。其目标是安排每项任务到某台设备上加工，使设备组的 makespan 达极小。即使设备组中加工时间最长的设备，其加工时间达极小。

以上生产调度问题，可用以下的整数规划模型来描述[1]：

**模型1**：

$$\text{minimize } V$$

$$s.t. \quad \sum_{i=1}^{n} X_{ij} p_i \leqslant V \qquad j = 1, 2, \cdots, m$$

* 收文日期 1993-02-15
本文得到重庆大学青年基金部分资助

$$\sum_{j=1}^{m} X_{ij} = 1 \qquad i = 1,2,\cdots,n$$

$$X_{ij} = 0,1$$

$$V \geqslant 0 \text{ 且为整数}$$

其中 $X_{ij} = 1$ 表示任务 $i$ 被安排给第 $j$ 台设备加工，否则 $X_{ij} = 0$.

由于假定设备组中的加工设备相同，因此模型 1 中假设每项加工任务在各台设备上的加工时间相同，是基本合理的。然而，在实际的生产加工设备中，虽然其中的加工设备均能完成各项加工任务，但同一任务，在不同的设备上其加工时间一般是不同的。产生这种情况的原因是多种多样的。比如，设备组中的加工设备型号不一样，或者组中各台设备已服役时间的长短不同，甚至设备的生产厂家不一样等等。因此，模型 1 则无法描述这种情况下的生产调度问题。本文将给出一个描述并行加工设备组生产调度的一般模型，它使[1 ～ 5]中的模型成为其特例。由于[1 ～ 5]中的算法，不能求解这个一般模型，我们还将给出两个启发式算法：极大消去法和ELPT方法。其特点是算法复杂性低，且极大消去法还能在一个矩阵上运算求解的优点。对ELPT方法，我们还给出了一个误差分析结果。最后，还给出了一个数值例子。

## 1　数学模型及启发式算法

设有 $m$ 台（不一定相同的）加工设备 $M_1, M_2, \cdots, M_m$ 构成一个并行加工设备组。有 $n$ 项独立的加工任务 $T_1, T_2, \cdots, T_n$，它们均可由组中任一台设备加工完成。它们在各台设备上的加工时间由以下矩阵给出：

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}$$

其中 $a_{ij}$ 表示 $T_i$ 由 $M_j$ 加工时的加工时间。我们称矩阵 $A$ 为加工任务集 $T = \{T_1, T_2, \cdots, T_n\}$ 关于设备组的负荷矩阵。这里，我们假定调整时间已包含在 $a_{ij}$ 之中。问如何安排 $T$ 中的每项任务到组中的设备上加工，使设备组的 makespan 达极小。

该问题的规划模型如下：

**模型 2：**

$$\text{minimize } V$$

$$\text{s.t.} \quad \sum_{i=1}^{n} a_{ij} X_{ij} \leqslant V \qquad j = 1,2,\cdots,m$$

$$\sum_{j=1}^{m} X_{ij} = 1 \qquad i = 1,2,\cdots,n$$

$$X_{ij} = 0,1$$

$$V \geqslant 0 \text{ 且为整数}$$

其中 $X_{ij} = 1$ 表示任务 $T_i$ 安排给 $M_j$ 加工完成，否则，$X_{ij} = 0$. 假设 $a_{ij}$ 是整数。

在模型 2 中，若令 $a_{ij} = p_i$，$j = 1,2,\cdots,m$ 则得到模型 1. 因此，模型 1 是模型 2 的一个特

例。

文献[1 ～ 5] 中所介绍的方法，都是求解模型 1 的算法。这些算法都无法用来求解模型 2. 当然，我们可用通常的规划方法求解模型 2. 然而正如[1][4] 等所指出的那样，模型 1(显然也包含模型 2) 是一个 NP 一完全问题，除特别情形外，一般没有算法复杂性为多项式的方法求其最优解。对于稍大一点的 $n$ 和 $m$，即使用分枝定界法求解，也是困难的。下面，我们给出一个求解模型 2 的启发式算法：极大消去法。值得注意的是，该方法同样可用于求解模型 1.

令 $M(j)=\sum_{i=1}^{n}a_{ij}, j=1,2,\cdots m$. 我们称 $M(j)$ 为设备 $M_j$ 的虚负荷。极大消去法是利用最速下降法的思想，它先在 $m$ 个虚负荷中，找出一个极大者 $M(j_0)$，然后在 $a_{ij_0}(i=1,2,\cdots,n)$ 中，选择一个极大元 $a_{i_0j_0}$，则确定 $T_{i_0}$ 不安排给 $M_{j_0}$ 加工，这样使 $M(j_0)$ 下降最快。进行 $(m-1)*n$ 次消去，使 $T_1,T_2,\cdots,T_n$ 都只有唯一的可供选择的加工设备。这样便得到一个近优的生产调度方案。

极大消去法的算法步骤：

Step 1. $M(j)\Leftarrow\sum_{i=1}^{n}a_{ij}$, $\quad j=1,2,\cdots,m$

$N(j)\Leftarrow m \quad\quad i=1,2,\cdots,n$

$M\Leftarrow\{1,2,\cdots,m\}$

Step 2. 按以下方式选取极大元 $a_{i_0j_0}$.

a) $I\Leftarrow\{i=1,2,\cdots,n$ 且 $N(i)>1\}$

b) $j_0\Leftarrow\min\{j\,|\,M(j)=\max\limits_{k\in I}M(k)$ 且存在 $i\in I$ 使 $a_{ij}>0\}$

c) $i_0\Leftarrow\min\{i\,|\,a_{ij_0}=\max\limits_{k\in I}(a_{kj_0})\}$

Step 3. $M(j_0)\Leftarrow M(j_0)-a_{i_0j_0}; a_{i_0j_0}\Leftarrow 0; N(i_0)\Leftarrow N(i_0)-1$

Step 4. 若 $\sum_{i=1}^{n}N(i)>n$ 则转 Step 2，否则结束。

算法结束后，矩阵 $A$ 的第 $j$ 列上所有非零元 $a_{ij}>0(i=1,2,\cdots,n)$ 对应的 $T_i$ 即是安排给 $M_j$ 的全部加工任务。而 $M(j)$ 就是 $M_j$ 设备完成这些任务所需加工时间之和($j=1,2,\cdots,m$). 由于寻找极大元 $a_{i_0j_0}$ 至多需要 $m*n$ 次计算，而算法完成要循环 $(m-1)*n$ 次，因此该算法的计算复杂性不超过 $O(m^2*n^2)$.

下面，我们对模型 2 的一个特殊情形，给出一个启发式算法：ELPT 方法，这是[2] 中 LPT 方法的一个推广。

设负荷矩阵 $A$ 中 $a_{ij}=\alpha_jp_i$，这时 $A$ 变成：

$$A=\begin{bmatrix}\alpha_1p_1 & \alpha_2p_1 & \cdots & \alpha_mp_1\\ \alpha_1p_2 & \alpha_2p_2 & \cdots & \alpha_mp_2\\ \cdots & \cdots & \cdots & \cdots\\ \alpha_1p_n & \alpha_2p_n & \cdots & \alpha_mp_n\end{bmatrix} \quad (*)$$

这样假定的物理意义是：以某台加工设备(不一定在该设备组内)为标准，任务 $T_i$ 在它上面加工完成需时间 $p_i(i=1,2,\cdots,n)$. 而把它安排给 $M_j$ 加工时，所需时间是 $\alpha_jp_i(j=1,2,\cdots,m)$. 并且

1) $\alpha_i<1$ 时，说明 $M_j$ 的加工能力比标准设备的加工能力强。

2) $\alpha_i < \alpha_j$ 时，说明 $M_i$ 的加工能力比 $M_j$ 的加工能力强。

值得注意的是，当负荷矩阵为(*)时，本文的模型 2 已包含了[4]中的模型。在[4]中假定组内 $m$ 台设备相同按不同的速度运行。此外，前面介绍的极大消去法，这里仍然适用。

为了讨论方便，我们不妨设：

(1) $\alpha_1 \leqslant \alpha_2 \leqslant \cdots \leqslant \alpha_m$；(2) $p_1 \geqslant p_2 \geqslant \cdots \geqslant p_n$

ELPT 方法的算法步骤：

Step 1. $M_j \Leftarrow \Phi, M(j) \Leftarrow 0, j = 1,2,\cdots,m; l \Leftarrow 1$

Step 2. $j_0 \Leftarrow \min\{j \mid M(j) = \min\limits_{1 \leqslant k \leqslant m} M(k)\}$

Step 3. $M(j_0) \Leftarrow M(j_0) + \alpha_{j_0} p_l$

$M_{j_0} \Leftarrow M_{j_0} \cup \{l\}; l \Leftarrow l + 1$

Step 4. 若 $l \leqslant n$ 则转 Step 2，否则结束。

算法结束后，集 $M_j$ 中元素即是安排给设备 $M_j$ 的加工任务的下标全体。$M(i)$ 就是 $M_j$ 完成所安排任务需要的时间。包括 $\alpha_j(j = 1,2,\cdots,m), p_i(i = 1,2,\cdots,n)$ 为排序时间在内，该算法的计算复杂性显然不会高于 $O(m * n^2)$.

## 2　最坏情形分析

我们知道，由启发式算法求得的解，通常是近优解(有时也会是最优解)。因此，分析一个启发式算法求得的解在最坏情形时，同理论最优解比较的误差究竟有多大，有着十分重要的理论和实际意义。下面，我们给出 ELPT 算法的一个比较分析结果。

对于 $k \geqslant 0$，我们在 $P = \{p_1, p_2, \cdots, p_n\}$ 中，选出 $k$ 项加工时间最长的任务，以最优方式安排给设备组加工，设其 makespan 是 $\omega_k$. 对余下的 $n - k$ 项任务按任意顺序排列成 $\bar{P} = \{p_{k+1}, \cdots, p_n\}$，再对 $\bar{P}$ 调用 ELPT 算法，设这时设备组关于 $P$ 的 makespan 是 $\omega(k)$，而设备组关于 $P$ 的理论最优 makespan 为 $\omega_0$，则我们可得如下定理。

定理：
$$\frac{\omega(k)}{\omega_0} \leqslant \frac{\alpha_m}{\alpha_1}\left[1 + \frac{1 + \frac{1}{m}}{1 + \left[\frac{k}{m}\right]}\right]$$

其中 $\left[\frac{k}{m}\right]$ 是不超过 $\frac{k}{m}$ 的最大整数。

证明：$\because \omega_k \leqslant \omega_0 \leqslant \omega(k)$

$\therefore$ 若 $\omega_k = \omega(k)$，则有 $\omega(k) = \omega_0$，则定理成立。

以下我们设 $\omega(k) > \omega_k$，这时有 $n > k$.

设 $\bar{p} = \max\limits_{k+1 \leqslant j \leqslant n} p_j$，又设 $p^*$ 是设备组最后加工完成的任务。

$\therefore$ 第 $m$ 台设备在时间 $\omega(k) - \alpha_m p^*$ 前没有空闲。

故有：

$$\alpha_m\left(\frac{1}{m}\sum_{\substack{p_i \in P \\ p_i \neq p^*}} p_i\right) \geqslant \omega(k) - \alpha_m p^*$$

由此可得：

$$\alpha_m\left(\frac{1}{m}\sum_{i=1}^{n}p_i\right)\geqslant \omega(k)-\alpha_m p^* + \frac{\alpha_m}{m}p^* = \omega(k)-\frac{m-1}{m}\alpha_m p^*$$

∴
$$\omega(k)\leqslant\frac{\alpha_m}{m}\sum_{i=1}^{n}p_i+\frac{m-1}{m}\alpha_m p^* \tag{1}$$

又∵
$$\omega_0\geqslant\alpha_1\left(\frac{1}{m}\sum_{i=1}^{n}p_i\right)$$

∴
$$\frac{1}{m}\sum_{i=1}^{n}p_i\leqslant\frac{\omega_0}{\alpha_1} \tag{2}$$

把(2)代入(1)得： $$\omega(k)\leqslant\frac{\alpha_m}{\alpha_1}\omega_0+\frac{m-1}{m}\alpha_m p^* \tag{3}$$

由假定，$P$ 中至少有 $k-1$ 项任务 $p_i\geqslant\bar{p}\geqslant p^*$

∴某台设备 $M_j$ 至少完成 $1+\left[\frac{k}{m}\right]$项加工时间比 $\bar{p}$ 长的加工任务。

∴
$$\omega_0\geqslant\alpha_j\left(1+\left[\frac{k}{m}\right]\right)\bar{p}\geqslant\alpha_1\left(1+\left[\frac{k}{m}\right]\right)p^*$$

故得
$$p^*\leqslant\frac{\omega_0}{\alpha_1\left(1+\left[\frac{k}{m}\right]\right)} \tag{4}$$

把(4)代入(3)，我们可得：

$$\omega(k)\leqslant\frac{\alpha_m}{\alpha_1}\omega_0+\frac{m-1}{m}\alpha_m\frac{\omega_0}{\alpha_1\left(1+\left[\frac{k}{m}\right]\right)}$$

∴
$$\frac{\omega(k)}{\omega_0}\leqslant\frac{\alpha_m}{\alpha_1}\left\{1+\frac{1-\frac{1}{m}}{1+\left[\frac{k}{m}\right]}\right\}$$

注意到当 $P=\{p_1,p_2,\cdots,p_n\}$ 中元素已按降排列好时，其前面的 $m$ 项任务即是 $P$ 中的 $m$ 项加工时间最长的任务，且按 ELPT 算法对这 $m$ 项任务进行调度，其结果是最优的。即有

推论：设 $P=\{p_1,p_2,\cdots,p_n\}$ 是按降序排列的，$\omega_E$ 是由 ELPT 算法求得的设备组的 makespan 值，则

$$\frac{\omega_E}{\omega_0}\leqslant\frac{\alpha_m}{\alpha_1}\left(\frac{3}{2}-\frac{1}{2m}\right)$$

证明：注意到 $\omega(m)=\omega_E$ 即得。

这个推论就是ELPT算法求得的设备组 makespan $\omega_E$ 在最坏情形同理论最优值 $\omega_0$ 的比较结果。

## 3 数值例子

本节，我们给出一个简单的数值例子，以说明极大消去法的求解过程。为计算方便，我们把每台设备的虚负荷写在负荷矩阵的下面。每一步选中的极大元和极大虚负荷，分别用圆圈圈上。

例：设组内有设备台数 $m=4$，有待加工任务数 $n=8$，其负荷矩阵 0) 步给出。为了节省篇幅，我们给出了开始的 3 步和最后的 2 步。

$$
0)\begin{bmatrix} 3 & 2 & 6 & 8 \\ 8 & 6 & 3 & 5 \\ 4 & 3 & 6 & 7 \\ 3 & 9 & 4 & 6 \\ 9 & 3 & 5 & 3 \\ 3 & 4 & 8 & 5 \\ 5 & 4 & 4 & 6 \\ 7 & 5 & 4 & 3 \\ 42 & 36 & 40 & 43 \end{bmatrix},
\quad \rightarrow \quad
1)\begin{bmatrix} 3 & 2 & 6 & 0 \\ 8 & 6 & 3 & 5 \\ 4 & 3 & 6 & 7 \\ 3 & 9 & 4 & 6 \\ 9 & 3 & 5 & 3 \\ 3 & 4 & 8 & 5 \\ 5 & 4 & 4 & 6 \\ 7 & 5 & 4 & 3 \\ 42 & 36 & 40 & 35 \end{bmatrix}
\quad \rightarrow \quad
2)\begin{bmatrix} 3 & 2 & 6 & 0 \\ 8 & 6 & 3 & 5 \\ 4 & 3 & 6 & 7 \\ 3 & 9 & 4 & 6 \\ 0 & 3 & 5 & 3 \\ 3 & 4 & 8 & 5 \\ 5 & 4 & 4 & 6 \\ 7 & 5 & 4 & 3 \\ 33 & 36 & 40 & 35 \end{bmatrix}
$$

$$
\rightarrow \quad
3)\begin{bmatrix} 3 & 2 & 6 & 0 \\ 8 & 6 & 3 & 5 \\ 4 & 3 & 6 & 7 \\ 3 & 9 & 4 & 6 \\ 0 & 3 & 5 & 3 \\ 3 & 4 & 0 & 5 \\ 5 & 4 & 4 & 6 \\ 7 & 5 & 4 & 3 \\ 33 & 36 & 32 & 35 \end{bmatrix}
\quad \rightarrow \quad
23)\begin{bmatrix} 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 3 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 3 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 3 \\ 6 & 8 & 7 & 6 \end{bmatrix}
\quad \rightarrow \quad
24)\begin{bmatrix} 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 3 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 3 \\ 6 & 5 & 7 & 6 \end{bmatrix}
$$

从以上 24）步的矩阵，我们得一个近优的生产调度方案：分别把 $T_4$、$T_6$ 给 $M_1$；$T_1$、$T_3$ 给 $M_2$；$T_2$、$T_7$ 给 $M_3$；$T_5$、$T_8$ 给 $M_4$ 加工，这时 $M_1$，$M_2$，$M_3$，$M_4$ 完成所得的加上任务分别需要 6、5、7、6 个单位时间。因此，设备组中加工时间最长的设备其加工时间是 7. 即设备组的 makespan 是 7 个单位时间。此外，我们还容易（比如用穷举法）验证，以上的解还是最优的。

## 4　结束语

本文给出的模型 2，是描述并行加工设备组生产调度问题的一个较一般的模型，它在机械制造加工业中有着广泛的实际背景。这里给出的极大消去法也具有一般性，它不仅可用于求解模型 2，对模型 1 也同样适用。数值例子显示，该算法具有很好的优化效果。

### 参 考 文 献


1　Lee C Y, Massey J D. Multiprocessor Scheduling: An Extension of the MULTIFIT Algorithm, J. of Manufacturing System, 1988, 7(1): 25～32

2　Graham R L, Bounds on Multiproessing Timing Anomalies, SIAM J. Appl. Math., 1969, 17(2): 416～429

3　Friesen D K, Langston M A. Evaluation of a MULTIFIT Based Scheduling Algorithm, J. of Algorithm, 1986, 7(1): 35～59

4　H. Van De Vel, Sun Shijie. An Application of the Bin-Packing Technique to Job Scheduling on Uniform Processors, J. Opl. Res., 1991, 42(2): 169～172

5　Friesen D K. Tighter Bounds for the MULTIFIT Processor Scheduling Algorithm, SIAM J. on Computing, 1984, 13(1): 170～181