DOI:10.16731/j.cnki.1671-3133.2008.03.007

Flow shop排序问题 Fm|prmu|C_{max}的改进分枝定界法*

谢金华,叶春明,马良,傅家旗
(上海理工大学管理学院,上海 200093)

摘要:针对 Flow shop排序问题,提出一种改进的分枝定界法,该算法融入了 Gupta启发式算法和分枝定界算法,在保证求得最优解的前提下减少了计算量,提高了效率。实例结果证明算法的有效性。

关键词:排序;分枝定界算法;Flow shop排序问题;启发式算法
中图分类号:TP301.6　**文献标识码**:A　**文章编号**:1671-3133(2008)03-0025-03

A G-B&B algorithm for the Flow shop scheduling problem Fm|prmu|C_{max}

Xie Jin-hua, Ye Chun-ming, Ma Liang, Fu Jia-qi
(School of Management, University of Shanghai for Science & Technology, Shanghai 200093, CHN)

Abstract: Presented an improved Branch and Bound(G-B&B) algorithm for flow shop scheduling problem. It has merged with the Gupta heuristic algorithm and the Branch and Bound(B&B) algorithm. This method has improved efficiency to a great extent, the example result showed the effectiveness of the algorithm.

Key words: Scheduling; Branch and Bound(B&B) algorithm; Flow shop scheduling problem; Heuristic algorithm

0 引言

Flow shop排序问题具有广泛的实用价值,对于目标函数是最大完工时间,且机床数 $m=2$时,Jhonson给出了一个最优算法[1];对于 $m\geqslant 3$时,已被证明,该问题是非多项式(Non Polynomial NP)难题。处理这类问题的常用方法是启发式算法和分枝定界法。

文献[2]~文献[4]给出了应用启发式算法求解这类问题,文献[5]、文献[6]提出了应用分枝定界法求解这类问题。对于启发式算法,计算量相对较小,通常只能求得近优解;而分枝定界法可以求得最优解,但其计算量较大,特别是对于较大规模的问题,更是如此。

本文提出一种改进的分枝定界算法,该算法综合了文献[2]和文献[5]的优点,首先通过 Gupta[4]启发式算法求得问题 Fm|prmu|C_{max}的一个较优的排序,然后把该排序的目标函数值即最大完工时间作为分枝定界法的初始上界,使上界初始化更接近最优的目标函数值,这样能在很大程度上减少分枝和计算量,从而提高求解的效率,最终求得这类问题的最优解。

通过实例证明,该算法对于求解最优排序的问题,具有很好的求解性能,有一定的应用价值。

1 问题描述

问题一般表述为[7]:设有 n个工件 $J_1, J_2, J_3, \cdots, J_n$,m台机器 $M_1, M_2, M_3, \cdots, M_m$。n个工件均依同一顺序在 m台机器 $M_1, M_2, M_3, \cdots, M_m$ 上加工。工件 J_j 在机器 M_i上的加工时间为 p_{ij}($i=1, 2, \cdots, m$; $j=1, 2, \cdots, n$)。目标函数为最大完工时间。由三参数表示法,以上求解问题可表示为:

$$Fm|prmu|C_{max} \quad (1)$$

式中:Fm为 m个处理机,流水作业;prmu为所有工件的每一工序的加工顺序都相同;C_{max}为目标函数是最大完工时间。

* 上海市重点学科建设资助项目(T0502)

2 改进的分枝定界算法

本文对分枝定界(Branch and Bound, B&B)算法所作的改进主要集中在其上界的初始化方面,这里引入 Gupta启发式算法确定一个较优排序,并将其目标函数值即最大完工时间作为分枝定界的初始上界,从而避免了以往标准分枝定界法对于上界初始化的随意性和夸大性。使得分枝定界能更快速地得到最优排序。

2.1 Gupta启发式算法

Gupta启发式算法是由 Gupta于 1972年提出的求解问题 $Fm|prmu|C_{max}$的启发式算法,其算法的复杂度为 $O(n\log n+mn)$[4]。该算法先定义工件 J_j的优先因子 π_j如下:

$$\pi_j=\frac{e_j}{\min\limits_{1\leqslant k\leqslant m-1}(p_{k,j}+p_{k+1,j})}\quad(j=1,2,\cdots,n)\qquad(2)$$

式中:$e_j=\begin{cases}1 & p_{1,j}<p_{m,j}\\ -1 & p_{1,j}\geqslant p_{m,j}\end{cases}$

然后将工件按优先因子 π_j不增排列,即可得到一个较优的工件加工排序。该排序对应一个最大完工时间 C_{max}。

2.2 分枝定界算法

参考文献[5]和文献[8],求解问题 $Fm|prmu|C_{max}$,分枝定界法的要点是对于 n个工件的排列排序问题,一个完整的搜索树共有 n层节点。第 0层为根节点,从根节点分枝,产生第 1层 n个节点,每个节点对应一个第 1个位置已经排定好工件的部分排序。从第 1层的一个节点分枝可产生第 2层 $n-1$个节点,每个节点对应一个两个位置已排定工件的部分排序。一般地,从第 $r-1$层的一个节点分枝可产生第 r层 $n-r+1$个节点,第 r层的每个节点对应一个前 r个位置已排定工件的部分排序。确定各节点的下界。由第 $n-1$层的一个节点可以确定一个可行排序,其目标函数值作为最优排序目标函数值的一个上界。淘汰下界不小于当前最优目标函数值上界的节点。对下界小于当前最优目标函数值上界的节点进行分枝,不断改进最优排序目标函数值的上界可以得到最优排序。分枝定界搜索过程如图 1所示。

确定下界的方法如下。

设已按顺序 $J_1,J_2,\cdots,J_r$排好 r个工件,还有 $n-r$个工件 $J_{r+1},J_{r+2},\cdots,J_n$未排。工件 J_j在机器 M_i上的完工时间为 $C_i(J_j)$。记 $\xi_r=\{J_1,J_2,\cdots,J_r\}$,$\bar{\xi}_r=\{J_{r+1},J_{r+2},\cdots,J_n\}$。对于 m台机器的一般情况,其下界可表示为:

$$B(J_r)=\max_{1\leqslant k\leqslant m}\{C_k(J_r)+\sum_{J_j\in\bar{\xi}_r}p_{k,j}+\min_{J_j\in\bar{\xi}_r}\sum_{i=k+1}^{m}p_{i,j}\}\qquad(3)$$

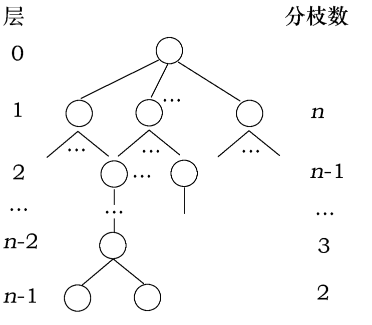

图 1 分枝定界法搜索树

2.3 改进的分枝定界(G-B&B)算法

对于启发式算法,计算量相对较小,但通常只能求得近优解,而分枝定界法虽然可以求得最优解,但其计算量较大,特别是对于较大规模的问题,更是如此。因此本文综合了这两种方法的优点,把 Gupta启发式算法的思想应用到分枝定界法,不但能够求得此类问题的最优解,而且很大程度上提高了求解的效率。G-B&B算法步骤如下。

1)首先利用 Gupta启发式算法,用式(2)可以确定一个较优的排序,然后把其排序的目标函数值(即最大完工时间 C_{max})作为分枝定界算法的初始上界 $\bar{f}$。然后从根节点分枝,产生第 1层 n个节点,用式(3)计算各节点下界 $\underline{z}$。如各节点下界均不小于 $\bar{f}$则转步骤 6),否则,选择具有最小下界的节点进行分枝(若有多个,选择下标最小者)。

2)从已选定的第 $r-1(r=2,3,\cdots,n-1)$层的节点分枝,产生第 r层 $n-r+1$个节点,如各节点下界均不小于 $\bar{f}$则转步骤 6),否则,选择具有最小下界的节点进行分枝(若有多个,选择下标最小者)。

3)若 $r=n-2$,转步骤 4),否则,转步骤 2)。

4)从选定的第 $n-2$层节点可产生第 $n-1$层两个节点,每个节点对应一个可行排序。其下界即为可行排序的目标函数值。如其目标函数值均不小于 $\bar{f}$则转步骤 5),否则比较两个可行排序的目标函数值,其中较小者为最优排序目标函数值新的上界 $\bar{f}$。

5)依次对当前搜索树中第 $r(r=1,2,\cdots,n-1)$层节点考察,淘汰下界不小于当前的目标函数值的上界的节点。如全部节点的下界均不小于 $\bar{f}$则转步骤 6)。否则,选择有最小下界的节点进行分枝,转步骤 2)。

6)给出 $\bar{f}$ 的节点确定一个最优排序,$\bar{f}$ 为最优排序目标函数值。算法终止。

3 计算实例

假设有五个工件和三台机器的 Flow shop排序问题 $Fm|prmu|C_{max}$,具体各个工件在每台机器上的加工时间如矩阵 P。矩阵中任意一个元素 P_{ij} 表示第 i个机器加工第 j个工件所用的时间($i=1,2,3$; $j=1,2,3,4,5$)。

$$P=\begin{pmatrix} 6 & 4 & 5 & 9 & 3 \\ 8 & 1 & 5 & 5 & 9 \\ 2 & 1 & 6 & 8 & 5 \end{pmatrix}$$

根据改进的分枝定界算法,首先由 Gupta启发式算法可以求得一个较优的排序为:$[J_3, J_5, J_4, J_1, J_2]$,进而求得该排序的最大完工时间 $C_{max}=35$,即令分枝定界算法的初始上界 $\bar{f}=35$。然后根据 $\bar{f}$ 和每个节点的下界 $\underline{z}$ 进行分枝定界。其中 $\bar{f}<\underline{z}$ 时没有必要再分枝,称为剪枝。整个分枝定界过程如图 2所示,图 2中节点旁边括号内的数字为下界,x为剪枝。该实例的最优排序为 $[J_5, J_3, J_4, J_1, J_2]$,求得最优的最大完工时间 $C_{max}=34$。

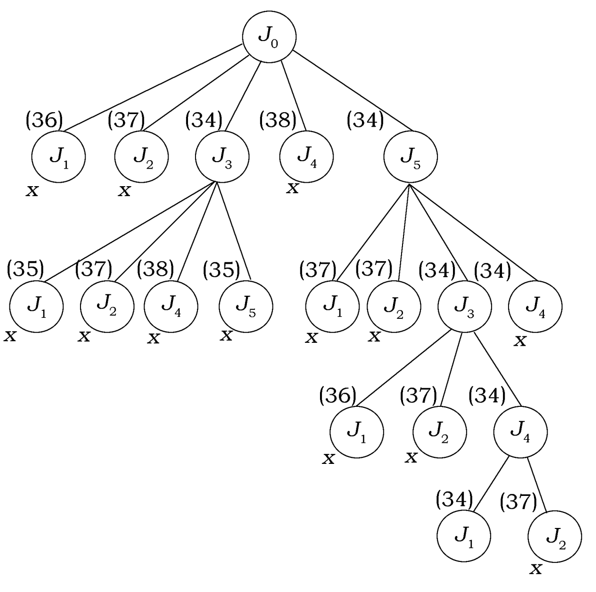

图 2 分枝定界法搜索实例

为了显示改进的分枝定界算法的性能,根据以上实例结果,将改进的分枝定界(G-B&B)算法同普通的分枝定界(B&B)算法进行比较。结果见表 1。比较结果表明,改进的分枝定界算法节点访问率和上界更新次数都明显少于普通的分枝定界法,结果显示本文算法的良好性能。

表 1 G-B&B算法与 B&B算法比较

算 法	节点访问率/%	上界($\bar{f}$)更新次数
G-B&B	8.8	1
B&B	13.7	3

4 结语

实例计算表明,由 Gupta启发式算法给出的节点初始下界比较接近最优目标函数值,能快速淘汰节点,很快地求出最优排序。针对 Flow shop排序问题,本文提出的改进分枝定界算法显示出了很高的效率,对求解最优排序的此类问题,表现出很好的求解性能,且便于计算机实现,具有一定的应用价值。下一步研究重点是对算法的应用,也可考虑更多的约束。

参考文献:

[1] Johnson S M. Optimal two-and-three-stage production schedules with set-up times included[J]. Naval Res Logist Quart, 1954(1):61-68.

[2] Palmer D S. Sequencing Jobs Through a Multi-Stage Process in the Minimum Total Time-A Quick Method of obtaining a Near Optimum[J]. Oper Res Quart, 1956(16):101-107.

[3] Campbell H G, Dudek R A, Smith M L. A Heuristic Algorithm for the n Job m Machine Sequencing Problem[J]. Management Science, 1970(16):630-637.

[4] Gupta J N D. Heuristic Algorithms for Multistage Flow Shop Problem[J]. AIIE Trans, 1972(4):11-18.

[5] Lomnicki Z. A Branch-and-Bound Algorithm for the Exact Solution of the Three-Machine Scheduling Problem[J]. Oper Res Quart, 1965(16):89-100.

[6] McMahon G B. Optimal Production Schedules for Flow-shops[J]. Canadian Operations Research Journal, 1969(7):141-151.

[7] 张传立,唐恒永. Flow Shop排序问题 $Fm|prmu|\sum w_j C_j$ 的分枝定界法[J].应用数学与计算数学学报,1999(13):30-36.

[8] 唐恒永,张传立.排序引论[M].北京:科学出版社,2002.

作者简介:谢金华,硕士研究生,研究方向:生产调度。

叶春明,教授,博士生导师,主要研究领域为工业工程,生产调度。

马良,博士生导师,主要研究领域为系统工程,运筹学,智能计算。

E-mail: jinhuaxie@126.com

收稿日期:2007-11-22