文章编号: 1001-0920(2006)03-0253-05

一种基于分解交货期的 Job Shop 启发式调度算法

刘 琳, 谷寒雨, 席裕庚
(上海交通大学 自动化研究所, 上海 200030)

摘 要: 针对以拖期加权和为目标的 Job shop 调度问题, 提出一种基于分解交货期的启发式调度方法. 首先根据工件的允许流比率确定每道工序的初始交货期; 然后在活动调度框架下应用改进的 MOD 规则确定工件在机器上的加工顺序. 在迭代优化过程中不断调整关键工序的交货期以改善调度的质量, 并考虑了工件之间的相互影响. 算例仿真研究表明, 该算法可以在较短计算时间内得到较好解, 可以满足实际 Job shop 系统对调度质量和计算效率的要求.

关键词: 车间调度; 拖期; MOD 规则
中图分类号: TP301 **文献标识码**: A

A Heuristic Method for Job Shop Scheduling Based on Decomposed Due Date

LIU Lin, GU Han-yu, XI Yu-geng
(Institute of Automation, Shanghai Jiaotong University, Shanghai 200030, China. Correspondent: LIU Lin, E-mail: liulin2003@sjtu.edu.cn)

Abstract: A heuristic method for job shop scheduling based on decomposed due date to minimize total weighted tardiness is proposed. The initial due date of each operation is determined according to the flow allowance rate of each job, then the job sequences on all machines are obtained by the improved modified operation due date rule based on Giffler-Thompson scheme. And the due date of the critical operation is adjusted to improve the solution quality with considering interactions among jobs at each iteration. The simulation results show that the proposed method can obtain good solutions with acceptable computational efficiency, and can be used to the real job shop scheduling system.

Key words: Job shop scheduling; Tardiness; Modified operation due date rule

1 引 言

现代制造系统中, 特别是面向客户订单的生产系统, 评价生产的一个重要指标是满足客户交货期要求的程度. 因此, 性能指标以交货期为基础的调度问题得到了较为广泛的研究[1~11]. 已经证明, 单机 $1\|\sum \omega T_i$ 问题是强 NP-hard 的[1], 其扩展形式 $Jm\|\sum \omega T_i$ 的Job shop 问题也是强NP-hard的. 因此, 以往的研究大都采用启发式方法, 而分派规则[2~5] 是其中的一种主要方法. 分派规则的主要优势在于计算效率高, 可以实时利用局部信息不断地适应调度环境的变化, 具有较强的鲁棒性, 更适于求解动态大规模调度问题. 但是, 一般分派规则最初都是针对单机调度问题提出的, 对于更加复杂的 Job shop 环境如何采用合适的规则或规则与局部搜索方法相结合是研究的热点问题[12~14].

1960 年, Giffler 等提出了生成活动调度的策略[15], 在解决调度问题时得到了成功的应用[16,17]. 针对性能指标以交货期为基础的调度问题, 文献 [2~5] 的研究结果表明,MOD 规则 (Modified operation due date rule) 首先将工件的交货期分解成每道工序的交货期; 然后在机器的决策时刻采用该规则计算工序的优先权, 从而确定要加工的工序,

收稿日期: 2005-01-24; **修回日期**: 2005-04-25.
基金项目: 国家自然科学基金项目(60274013, 60474002); 上海市科委重大项目(04dz11008).
作者简介: 刘琳(1978—), 女, 河北保定人, 博士生, 从事制造系统车间生产规划与调度算法等研究; 席裕庚(1946—), 男, 上海人, 教授, 博士生导师, 从事预测控制、智能机器人系统及生产调度等研究.

可以明显改善调度的质量. 文献[6]针对目标为 $\sum T_i$ 的 Job shop 问题提出了一种分解方法, 借鉴移动瓶颈机的思想[18], 以单机为基础, 通过不断调整工序的交货期来计算原问题的次优解. 文献[7]针对目标为 $\sum T_i$ 的 Flow shop 问题设计了一种分解策略, 改进了 MOD 规则, 算例仿真取得了较好效果. 以上研究结果都表明了基于交货期分解的MOD 动态分派规则适于解决性能指标为拖期的 Job shop 调度问题.

本文针对目标为 $\sum \omega T_i$ 的Job shop 调度问题, 提出了关键工序的概念, 根据改进的MOD 规则, 在 Giffler-Thompson 活动调度的框架下设计了以分解交货期为基础的一种迭代优化启发式方法. 算例仿真分析表明, 该方法可以在较短时间内得到满意解, 明显优于改进MOD 规则的计算结果, 可以成功地应用于工件到达时间确定的动态 Job shop 调度问题.

2 问题的数学描述

本文所要研究的经典 Job shop 调度问题具体描述如下:

1) m 台不同的机器, 每台机器执行的操作不同;

2) n 个不同的工件, 每个工件包含 m 道工序, 需要分别在 m 台机器上加工, 工件的工艺顺序事先给定;

3) 一道工序只能执行一次;

4) 加工过程不允许抢占;

5) 不同工件的工序之间没有先后顺序约束;

6) 每台机器一次至多加工一道工序.

本文采用整数规划的形式描述目标为 $\sum \omega T_i$ 的调度问题. 对于 m 台机器 n 个工件的Job shop 问题的数学描述如下:

目标函数

$$\min_{X_{i,j,t}} \sum_i \omega\Big[\sum_{t>d_i-p_{i,m}} (1-X_{i,m,t})\Big]. \tag{1}$$

约束条件

$$X_{i,j,t+1} \geq X_{i,j,t},\ i=1,2,\cdots,n,$$
$$j=1,2,\cdots,m,\ t=1,2,\cdots,H-1; \tag{2}$$
$$X_{i,j,t} \leq X_{i,j-1,t-p_{i,j-1}},\ i=1,2,\cdots,n,$$
$$j=2,3,\cdots,m,\ t=1,2,\cdots,H; \tag{3}$$
$$\sum_{i,j} \delta_{i,j,k}(X_{i,j,t}-X_{i,j,t-p_{i,j}}) \leq 1,$$
$$k=1,2,\cdots,m,\ t=1,2,\cdots,H; \tag{4}$$
$$X_{i,j,t} \in \{0,1\},\ i=1,2,\cdots,n,$$
$$j=1,2,\cdots,m,\ t=1,2,\cdots,H. \tag{5}$$

$$X_{i,j,t}=\begin{cases}1,\ \text{工件 } i \text{ 的第 } j \text{ 道工序} \\ \quad \text{在 } t \text{ 时段已经开始加工;} \\ 0,\ \text{否则}\end{cases}$$
$$\delta_{i,j,k}=\begin{cases}1,\ \text{工件 } i \text{ 的第 } j \text{ 道工} \\ \quad \text{序在机器 } k \text{ 上加工;} \\ 0,\ \text{否则}\end{cases}$$

其中: $p_{i,j}$ 为工件 i 第 j 道工序的加工时间, d_i 为工件 i 的交货期, ω 为工件 i 的权重, H 为计划时域的长度.

式(1) 表明目标函数为最小化拖期加权和; 式(2) 保证了工件加工过程不允许抢占; 式(3) 保证了同一工件工序之间的顺序, 表明一道工序必须在其前面的所有工序都完成后才可以开始加工; 式(4) 说明在给定的时段一台机器至多可以加工一道工序, 从而保证机器能力约束得到满足; 式(5) 表明决策变量是 0-1 整数变量.

3 基于分解交货期的启发式方法

3.1 背景知识

定义 1[5] 工序的修改交货期为

$$d_{ij}^{\mathrm{MOD}} = \max\{t_m + p_{ij}, d_{ij}\}. \tag{6}$$

其中: d_{ij}^{MOD} 为工件 i 第 j 道工序的修改交货期, t_m 为机器 m 的当前可用时刻, d_{ij} 为工件 i 第 j 道工序的分解交货期.

定义 2[7] 工序的改进修改交货期为

$$d_{ij}^{\mathrm{MOD}} = \max\{\max\{r_{ij}, t_m\} + p_{ij}, d_{ij}\} + \max\{r_{ij}, t_m\}. \tag{7}$$

其中: d_{ij}^{MOD} 为工件 i 的第 j 道工序的改进修改交货期, r_{ij} 为工件 i 的第 j 道工序的最早可以开始时间.

以往研究结果表明, 将工件的交货期分解成每道工序的交货期, 可改善调度的质量. MOD 规则是一种动态分派规则, 它选择修改交货期最小的工序加工, 调度决策与机器的当前工作时刻相关. 对于 $\sum T_i$ 问题, 文献[5]中MOD 规则明显优于其他分派规则, 它实质上是在交货期紧和交货期松两种情况下完成由 SPT 规则和 EODD 规则的相互切换. 但是,MOD 规则仅考虑到机器当前时刻等待加工的工件, 没有考虑即将到达需要在该机器上加工的工件. 文献[7]改进了原来的MOD 规则, 在所考虑的工件集合中加入了未来即将到达的工件, 在局部利用了未来的一部分信息, 从而得到了比MOD 规则质量更好的调度. 本文在文献[7]的基础上, 利用该规则在机器的决策时刻确定要加工的工序.

Giffler-Thompson 方法是所有以分派规则为核心的启发式方法的基础. 每次迭代选出一台机器与在该机器上加工的工序的冲突集合, 根据分派规则

计算冲突集合中所有工序的优先权, 选择优先权最高的工序加工, 逐步构建整个调度[15]. 相对于简单的分派规则, 该方法生成调度的质量较高.

3.2　算法设计思想

本文设计的基于分解交货期的启发式调度方法, 首先将工件的交货期分解为每道工序的交货期, 在 Giffler-Thompson 活动调度框架下采用改进的 MOD 规则选择机器, 同时确定该机器要加工的工序, 逐步构建一个完整调度, 并在迭代优化过程中不断调整关键工序的交货期, 逐步改善调度的质量. 以往的分派规则大多采用优先权的one-pass局部解过程, 而本文方法则在交货期调整过程考虑了被调整工件对其他工件的影响, 从全局的角度解决问题.

定义3　当前迭代中, 最大加权拖期工件 l 的具有最大工序分解拖期的工序, 称为工件 l 的关键工序 O_l^{cr}, 即

$$O_l^{cr} = \arg\max_{1\le j\le m}\{T_{lj}\},$$

$$l = \arg\max_{1\le i\le n}\{\omega T_i\}; \tag{8}$$

$$T_{lj} = \max\{0, C_{lj} - d_{lj}\}. \tag{9}$$

其中: T_{lj} 为工件 l 的第 j 道工序的拖期, C_{lj} 为工件 l 的第 j 道工序的完成时间, d_{lj} 为工件 l 的第 j 道工序的分解交货期.

本文提出的关键工序概念借鉴了瓶颈[18] 的思想, 解决一个问题, 并不是所有因素的重要性是相同的, 首先应找到最重要的因素, 解决了这个关键因素有助于整个问题的解决. 因此在迭代优化过程中, 每次迭代都要确定关键工序, 从关键工序开始逐步改善调度.

初始工序交货期如何确定对MOD 规则的求解效果有很大的影响, 一般有以下两种计算形式:

$$d_{ij} = d_i - P_i + P_{ij},$$

$$i = 1, 2, \cdots, n,\ j = 1, 2, \cdots, m. \tag{10}$$

其中: P_i 为工件 i 的总加工时间, P_{ij} 为工件 i 从第 1 道工序到第 j 道工序的加工时间和.

$$d_{ij} = \begin{cases} \alpha p_{ij}, j = 1; \\ d_{i,j-1} + \alpha p_{ij}, j = 2, 3, \cdots, m; \end{cases}$$
$$i = 1, 2, \cdots, n. \tag{11}$$

其中允许流比率 $\alpha = d_i / P_i$.

本文在算例仿真研究中, 对两种情况进行了比较, 发现采用第 2 种初始交货期的形式效果更好. 因此, 本文采用式(11) 计算初始交货期.

当 O_l^{cr} 的交货期确定后, 根据式(11) 中允许流比率的概念, 同时调整同一工件中位于关键工序前后的工序的交货期, 合理体现出关键工序交货期对其他工序的影响, 采用如下公式计算同一工件中其他工序的交货期:

$$\begin{cases} d_{ij}^s = (x - p_{ik}) p_{ij} / P_{ik-1}, j = 1; \\ d_{ij}^s = d_{ij-1}^s + (x - p_{ik}) p_{ij} / P_{ik-1}, \\ \qquad j = 2, \cdots, k-1; \\ d_{ij}^s = d_{ij-1}^s + (d_i - x) p_{ij} / (P_i - P_{ik}), \\ \qquad j = k+1, \cdots, n. \end{cases} \tag{12}$$

其中: d_{ij}^s 为工件 i 第 j 道工序调整后的交货期; O_l^{cr} 为工件 l 的第 k 道工序; x 为 O_l^{cr} 的交货期, 取值范围为 $[\sum_{j=1}^{k} p_{ij}\quad d_{ik}]$; P_{ik} 为工件 i 从第 1 道工序到第 k 道工序的总加工时间.

3.3　算法具体过程

本文设计的迭代优化启发式调度算法过程如下:

Step1: 初始化. 初始最优目标值 $z(q) = \infty$, $q = 0$, 根据工件的交货期, 采用式(11) 计算每道工序的初始交货期.

Step2: 在 Giffler-Thompson 方法框架下, 根据改进的MOD 规则(7), 逐步确定工件在每台机器上的加工顺序:

(1) 令 $h = 1$, 部分调度集合 $PS_h = \Phi$, S_h 为没有前道工序的所有工序集合;

(2) 对 S_h 中的所有工序, 计算其当前时刻的修改交货期, 并按非减顺序排列;

(3) 选择排在首位的工序 O_{ij}^s 进行加工, $PS_{h+1} = PS_h \cup \{O_{ij}^s\}$, $S_{h+1} = S_h \setminus \{O_{ij}^s\} \cup \{O_{i,j+1}^s\}$, $h = h + 1$;

(4) 如果 PS_h 为完整调度, 则结束; 否则返回第(1) 步.

Step3:

(1) 令初始未经交货期调整的工件集合为 Ω, 已经调整的工件集合为 Ψ, 其中 $|\Omega| = n$, $|\Psi| = 0$.

(2) 计算所有工件的独立目标值, 按单个工件加权拖期的非增顺序排列所有未调整的工件, 对排在首位的工件 l^s 计算每道工序的拖期, 选择该工件具有最大拖期的工序作为关键工序 $O_{l^s}^{cr}$.

(3) 在关键工序 $O_{l^s}^{cr}$ 交货期可选择的范围内, 通过迭代过程确定其最优交货期:

1) 在 $[\sum_{j=1}^{k} p_{ij}\quad d_{ik}]$ 范围内从小到大依次选择整数值 x 作为 $O_{l^s}^{cr}$ 的交货期, 按式(12) 调整同一工件其他工序的交货期;

2) 应用 Step2 重新排列工件在机器上的加工顺序, 记录每次的目标值;

3) 令得到最小目标值的交货期作为 $O_{l^s}^{cr}$ 的最优交货期 x^*, 并相应确定同一工件其他工序的交货

期

(4) 在 Ω 集合中删掉已调整的工件 l^s, $\Omega = \Omega \setminus \{l^s\}$. 若 $\Omega = \Phi$, 则结束; 否则返回步骤(2).

Step 4: 记录 Step 3 结束后得到的目标值, 与最优目标值进行比较, 若连续两次迭代中目标值没有得到改善, 则迭代优化过程结束, $z(q) = z(q-1)$, 输出最优解, 否则返回 Step 3, $q = q+1$.

本文方法中值得注意的一点是, 应用改进的 MOD 规则计算工序的修改交货期时, 如果机器等待优先权最高工序到达的空闲时间足够加工一个其他准备在该机器上加工的工序, 则先加工这道工序, 这并不影响优先权最高工序在该机器上的完成时间. 工序交货期的范围大小显著影响着算法的运算效率, 同时影响解的性能, 范围取的越大, 对解的性能改善的程度越大, 但并不是所有关键工序的交货期范围取的越大越好. 从实际应用的角度出发, 这里工序交货期范围取的较小, 降低运算时间的同时兼顾了解的质量.

4 算例仿真

本文在 Celeron-2.0G 的个人计算机上, 采用 C 语言编程, 对规模为 10 × 10 的算例进行了仿真研究. 其中 ABZ5 和 ABZ6 参见文献[18], LA 16～LA 24 参见文献[19], 它们是 15 × 10 问题, 将最后 5 个工件剔除便构成 10 × 10 问题. 工件的权重以及交货期紧度控制参数的选取参照文献[20], $\omega_1 = \omega_2 = 4$, $\omega_i = 2$, $i = 3, 4, \cdots, 8$, $\omega_9 = \omega_{10} = 1$, 交货期按下式计算:

$$d_i = r_i + \left[f \sum_{j=1}^{10} p_{ij}\right], \tag{13}$$

其中 f 为交货期的紧度控制参数.

文献[20]列出了算例在 $r_i = 0$, $i = 1, 2, \cdots, 10$, $f = 1.5$ 和 $f = 1.6$ 两种情况下的最优解. 本文对上述 11 个算例在同样的情况下进行了仿真研究, 仿真结果如表 1 所示, 表中所列为问题的目标值. 可以看出, 本文的迭代优化方法明显优于改进的 MOD 规则, 对于 22 个问题求到了 6 个问题的最优解, 在较松交货期情况下的效果更好.

本文同时研究了工件在时域动态到达, 且到达时间确定已知的情况. 工件的到达时间 r_i 在[0, 100]服从离散均匀分布, 针对每一个算例分别在 $f = 1.5$ 和 $f = 1.6$ 两种情况下生成 10 组数据, 一共生成了 220 组数据, 仿真结果如表 2 所示. 表 2 中所列的改善率和 CPU 时间分别为针对一个算例的 10 组数据仿真结果的平均值, 其中目标值改善率为 $(J^{MOD} - J)/J^{MOD} 100\%$, J^{MOD} 为改进 MOD 规则的计算结果, J 为本文方法的计算结果. 对220组数据进行仿真, 本文方法共有 95 次达到最优解, 达优率为 43.18%, 改进MOD 规则的目标值平均改善率为 67.35%, 但改进MOD 规则仅有 32 次达到最优解, 达优率仅为 14.55%. 本文所设计算法的时间复杂度为 $O(mn^3K)$. 其中: m 为机器数; n 为工件数; $K = \max_i(d_i - P_i)$, d_i 为工件 i 的交货期, P_i 为工件 i 的加工时间. 对于 10 × 10 规模问题, 算法的运行时间一般在 1 s 左右, 可以满足计算效率的要求.

表 1　典型算例仿真结果

算例	$f = 1.5$ 本文方法	$f = 1.5$ 改进的 MOD 规则	$f = 1.6$ 本文方法	$f = 1.6$ 改进的 MOD 规则
ABZ5	275	363	9	118
ABZ6	0	68	0	106
LA 16	320	804	118	438
LA 17	451	1 049	229	710
LA 18	62	1 010	0	375
LA 19	225	864	0	300
LA 20	224	564	8	198
LA 21	41	450	0	339
LA 22	826	1 531	307	1 025
LA 23	195	353	0	138
LA 24	632	769	24	282

表 2　动态环境下的仿真统计结果

算例	$f = 1.5$ 情况 改善率 %	$f = 1.5$ 情况 CPU 时间/s	$f = 1.5$ 情况 达优数	$f = 1.6$ 情况 改善率 %	$f = 1.6$ 情况 CPU 时间/s	$f = 1.6$ 情况 达优数
ABZ5	61.21	1.44	1	34.56	1.45	7
ABZ6	50.00	0.89	10	20.00	0.84	10
LA 16	59.74	1.12	0	88.68	0.91	4
LA 17	69.20	1.18	0	73.73	1.08	1
LA 18	70.33	1.06	0	99.86	0.87	8
LA 19	82.67	1.26	2	49.34	0.74	7
LA 20	75.06	1.40	2	98.59	1.31	8
LA 21	79.89	0.95	1	77.05	0.67	9
LA 22	64.26	1.40	0	87.41	1.85	1
LA 23	29.23	0.76	5	40.00	0.74	10
LA 24	72.65	1.48	0	98.14	1.03	9

5 结　　语

本文针对以 $\sum \omega T_i$ 为目标的 Job shop 调度问题, 提出了一种基于分解交货期的迭代优化启发式调度方法. 首先将工件的交货期分解为每道工序的交货期; 然后在 Giffler-Thompson 方法框架下, 根据改进的 MOD 规则确定工件在机器上的加工顺

序, 在迭代优化过程中通过不断调整关键工序的交货期来改善解的质量, 在调整过程中从全局角度考虑了工件之间的相互影响. 本文算法的时间复杂度为 $O(mn^3K)$. 仿真研究表明, 本文方法可以在较短的计算时间内得到满意解, 满足了实际应用的需要. 另外, 本文方法可以为车间层调度提供一个较好的初始可行调度, 根据调度环境的变化快速调整初始可行调度, 同时可以结合滚动时域分解方法求解大规模的动态Job shop 调度问题.

参考文献(References)

[1] Lenstra J, Rinnooy Kan A, Brucker P. Complexity of Machine Scheduling Problems[J]. *Annals of Discrete Mathematics*, 1977, 1: 343-362.

[2] Conway R W. Priority Dispatching and Job Lateness in a Job Shop[J]. *J of Industrial Engineering*, 1965, 16(4): 228-237.

[3] Baker K R. Sequencing Rules and Due Date Assignments in a Job Shop[J]. *Management Science*, 1984, 30(9): 1093-1104.

[4] Vepsalainen A P J, Morton T E. Priority Rules for Job Shops with Weighted Tardiness Costs[J]. *Management Science*, 1987, 33(8): 1035-1047.

[5] Baker K R, Kanet J J. Job Shop Scheduling with Modified Due Dates[J]. *J of Operations Management*, 1983, 4(1): 11-22.

[6] Raman N, Talbot F B. The Job Shop Tardiness Problem: A Decomposition Approach[J]. *European J of Operational Research*, 1993, 69(2): 187-199.

[7] Raman N. Minimum Tardiness Scheduling in Flow Shops: Construction and Evaluation of Alternative Solution Approaches[J]. *J of Operations Management*, 1995, 12(2): 131-151.

[8] 吴悦, 汪定伟. 公共交货期窗口下提前/拖期惩罚不同的单机调度问题[J]. *控制与决策*, 1998, 13(6): 659-664.
(Wu Y, Wang D W. Optimal Single Machine Scheduling with a Common Due Window[J]. *Control and Decision*, 1998, 13(6): 659-664.)

[9] 黄德才, 张平. 公共交货期窗口下提前/拖期问题的多机调度算法[J]. *控制与决策*, 1999, 14(增): 569-572.
(Huang D C, Zhang P. Earliness/Tardiness Job Scheduling Problem with a Common Due Window on Parallel Machines[J]. *Control and Decision*, 1999, 14(S): 569-572.)

[10] 姚伟力, 杨德礼, 胡祥培. Job-shop 提前/拖期调度问题的研究[J]. *控制与决策*, 2000, 15(3): 322-324.
(Yao W L, Yang D L, Hu X P. Earliness/Tardiness Job-shop Scheduling Problem[J]. *Control and Decision*, 2000, 15(3): 322-324).

[11] 陈冬雪, 王宏欣. 基于剩余率求解非标准作业车间调度问题逆序算法[J]. *计算机集成制造系统* — CIMS, 2004, 10(10): 1238-1241.
(Cheng D X, Wang H X. Reverse Algorithm for Solving Nonstandard Job-shop Scheduling Problem Based on Redundancy[J]. *Computer Integrated Manufacturing Systems*, 2004, 10(10): 1238-1241.)

[12] 万国华. 用模糊调度系统求解动态Job Shop 问题[J]. *系统工程理论与实践*, 2001, 21(8): 97-101.
(Wan G H. Using Fuzzy Scheduling System for Solving a Dynamic Job Shop Problem[J]. *System Engineering Theory and Practice*, 2001, 21(8): 97-101.)

[13] 孙容磊, 熊有伦, 杜润生, 等. 规则调度的迭代优化[J]. *计算机集成制造系统*—CIMS, 2002, 8(7): 546-550.
(Sun R L, Xiong Y L, Du R S, et al. Iterative Optimization of Rule-based Scheduling[J]. *Computer Integrated Manufacturing Systems*, 2002, 8(7): 546-550.)

[14] 王书锋, 邹益仁. 车间作业调度(JSSP)技术问题简明综述[J]. *系统工程理论与实践*, 2003, 23(1): 49-55.
(Wang S F, Zou Y R. Techniques for the Job Shop Scheduling Problem: A Survey[J]. *System Engineering Theory and Practice*, 2003, 23(1): 49-55.)

[15] Giffler B, Thompson G L. Algorithms for Solving Production Scheduling Problems[J]. *Operations Research*, 1960, 8(4): 487-503.

[16] Hutchison J, Chang Y L. Optimal Nondelay Job Shop Schedules[J]. *Int J of Production Research*, 1990, 28(2): 245-257.

[17] Sun D, Batta R. Scheduling Larger Job Shops: A Decomposition Approach[J]. *Int J of Production Research*, 1996, 34(7): 2019-2033.

[18] Admas J, Balas E, Zawack D. The Shifting Bottleneck Procedure in Job Shop Scheduling[J]. *Management Science*, 1988, 34(3): 391-401.

[19] Lawrence S. *Resource Constrained Project Scheduling: An Experiment Investigation of Heuristics Scheduling Techniques*[R]. Pittsburgh: GSIA, Carnegie Mellon University, 1984.

[20] Singer M, Pinedo M. A Computational Study of Branch and Bound Techniques for Minimizing the Total Weighted Tardiness in Job Shops[J]. *IIE Transactions*, 1998, 30(2): 109-118.