

# 柔性作业车间调度问题的集成启发式算法

陈 亮, 王世进, 周炳海

(上海交通大学机械与动力工程学院, 上海 200240)

**摘 要:** 柔性作业车间调度问题, 包括路径分配和加工排序 2 大子问题, 是组合优化理论和实际生产管理的重要研究方向。作为传统作业车间调度的扩展, 柔性作业车间调度问题的内在复杂性(强 NP-Hard)使得传统的最优化方法难以有效求解。文章针对以多目标权重和最优为目标的柔性作业车间调度问题, 提出基于过滤定向搜索的集成启发式算法, 设计改进了节点分枝策略和局部/全局评价函数, 能同时解决 2 大子问题。通过实例仿真, 对算法性能进行比较分析和评价, 结果表明了算法的可行性和有效性。

**关键词:** 柔性作业车间调度; 过滤定向搜索; 启发式方法

## Integrated Heuristic Algorithm for Flexible Job-shop Scheduling Problems

CHEN Liang, WANG Shi-jin, ZHOU Bing-hai

(School of Mechanical Engineering, Shanghai Jiaotong University, Shanghai 200240)

**【Abstract】** Flexible job-shop scheduling problem, containing routing sub-problem and sequencing sub-problem, is an important issue in both fields of combinatorial optimization and production operations. Being an extension of the classical job-shop scheduling problem, it presents additional complexity(NP-hard in strong sense), which makes traditional optimization approaches difficult to achieve an optimal solution in reasonable computational time. In this paper, a filtered-beam-search based heuristic algorithm is proposed to solve the flexible job-shop scheduling problem with the objective of optimal weighted sum of multiple measure performances. The proposed algorithm makes improvement in the generation procedures of branches and the local/global evaluation functions, which makes it possible to solve two sub-problems. the performance of the proposed algorithm is evaluated and compared with those of other representative approaches through simulations, and the results demonstrate the feasibility and efficiency of the proposed algorithm.

**【Key words】** flexible job-shop scheduling; filtered beam search; heuristic algorithm

### 1 概述

传统作业车间调度问题仅考虑作业具有唯一确定的加工工艺路线, 使得加工计划和生产调度脱节, 缺乏实用性<sup>[1-2]</sup>。针对这种情况, 柔性作业车间调度问题对传统作业车间调度问题进行了扩展, 结合实际制造车间中由于多功能加工设备引入而产生的具有柔性加工路径的生产调度需求, 比较符合生产实际情况。具有柔性加工路径生产的显著效益性(如缩短加工时间, 减少在制品库存等)和对应的柔性作业车间调度的高度复杂性(强NP-Hard问题)使得柔性作业车间调度问题的研究具有重要的理论和现实意义。

柔性作业车间调度问题包括 2 个子问题: (1) 路径分配子问题, 即把所有的工序分配给可选机床以确定加工路径; (2) 加工排序子问题, 即对已分配工序进行排序使得预先确定的目标最优。目前有关柔性作业车间调度问题的文献还较少。现有的研究方法主要集中于基于人工智能的启发式算法<sup>[1,3]</sup>, 代表性的有基于模拟退火和微粒群优化的混合方法(PSO+SA)<sup>[1]</sup>, 以局部最小化为分配模型的进化算法(AL+CGA)<sup>[3]</sup>等。这些方法可归为 2 类<sup>[1]</sup>: (1) 分解方法, 即分别考虑 2 个子问题; (2) 集成方法, 同时考虑 2 个子问题。为了有利于计划和调度的集成, 得到更优良的调度解, 近年来多数研究者强调对 2 个子问题同时集成进行求解。本文提出了一种基于过滤定向搜索的集成启发式算法, 用于研究以多

种目标加权和为最优的柔性作业车间调度问题。

### 2 柔性作业车间调度问题描述

柔性作业车间调度问题可描述为: 一加工系统中,  $n$  个作业要在  $m$  台机床上加工,  $m$  台机床的集合表示为  $M$ 。每个作业  $J_i$  ( $1 \leq i \leq n$ ) 有  $n_i$  道工序  $O_{ij}$  ( $1 \leq i \leq n, 1 \leq j \leq n_i$ ) 需要加工。每道工序  $O_{ij}$  由可加工的机床集  $M_{ij}$  中任一机床加工, 其中,  $M_{ij} \subseteq M$  (如果  $M_{ij} = M$ , 则是完全柔性作业车间调度问题; 如果  $M_{ij} \subset M$ , 则是局部柔性作业车间调度问题<sup>[3]</sup>)。调度的任务是在满足各种资源约束和工序前后关系约束的前提下, 把所有工序分配到可加工的机床上并确认工序的加工序列和开始加工时间使得预先确定的一个或多个目标最优。为了考虑常用的makespan、机床总负载和关键机床负载 3 种目标, 本文讨论的目标为  $F = w_1 F_1 + w_2 F_2 + w_3 F_3$ , 其中,  $F_1$  为 Makespan;  $F_2$  为机床总负载, 即各机床总的加工时间;  $F_3$  为关键机床负载, 即加工时间最长的机床负载;  $w_1 + w_2 + w_3 = 1$ 。

**基金项目:** 国家自然科学基金资助项目(60574054); 2006 新世纪优秀人才支持计划基金资助项目

**作者简介:** 陈 亮(1974 -), 男, 硕士研究生, 主研方向: 先进制造系统与控制技术; 王世进, 博士研究生; 周炳海, 副教授、博士

**收稿日期:** 2007-01-28 **E-mail:** wsjsy@sjtu.edu.cn

该调度问题的假设条件为：(1)工序加工一旦开始就不能中断；(2)所有作业相互独立，作业从下达时间  $r_i$  开始被加工；(3)所有机床相互独立，能在时间 0 开始加工；(4)每台机床一次只能加工一道工序，同一道工序也不能在多台机床上同时加工；(5)不考虑 Set-up 时间和作业在不同机床之间的转移时间，也不考虑机床故障情况。

### 3 集成启发式算法

#### 3.1 搜索策略定义

提出的集成启发式算法是基于过滤定向搜索的算法，过滤定向搜索方法是应用过滤机制的定向搜索方法，是分枝定界法的改进<sup>[4]</sup>。成功应用基于过滤定向搜索的启发式算法求解柔性作业车间调度问题需解决 4 个搜索策略定义问题：(1)解空间的搜索树表述；(2)分枝策略；(3)定向搜索宽度和定向过滤宽度确定；(4)评价函数的构造。

(1)解空间的搜索树表述采用图 1 所示的描述方法。在搜索树中，每一层中通过分枝策略产生的每个节点确定加工的机床和开始加工的时间，即确定  $O_{ijk}$  和  $s_{ijk}$ 。2 个节点之间的连线表示从上节点增加一道工序产生下一节点的决策，同时将所增加的工序添加到下一节点的局部调度中。

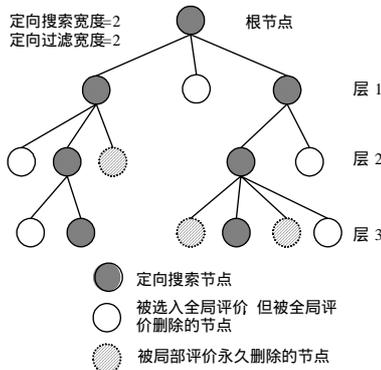


图 1 过滤定向搜索树的表述

(2)分枝策略采用最早可调度的工序优先策略<sup>[4]</sup>的改进形式，称为 M\_NDELAY 策略。假设  $PS_l$  是一个含有  $l$  道已调度工序的局部调度；对应  $PS_l$ ， $S_l$  是在层  $l$  时可以调度的工序集合； $s_{ij}$  是工序  $O_{ij} \in S_l$  最早可以开始加工的时间， $s_{i(j+1)} = s_{ij} + \gamma_{ij}$ ，其中， $\gamma_{ij} = \min_k(p_{ijk})$ ， $\forall 1 \leq i \leq n, 1 \leq j \leq n_i, M_k \in M_{ij}$ 。 $s_{i1} = r_i$  ( $r_i$  为前面所说的作业下达时间)。 $T_{ijk}$  为一可加工工序  $O_{ij}$  机床的最早可用时间。M\_NDELAY 策略描述如下：1) 确定  $T^* = \min_{O_{ij} \in S_l} \{s_{ij}\}$ ，并令工序具有  $s_{ij} = T^*$  且  $O_{ij} \in S_l$  的工序为一集合  $\bar{S}_l$ 。2) 对应局部调度，选择一道工序  $O_{ij} \in \bar{S}_l$ ，对每台可以加工该工序的机床  $M_k \in M_{ij}$  生成下一层节点并把该节点加入到局部调度  $PS_l$  中，每个节点开始时间为  $s_{ijk} = \max(T^*, T_{ijk})$ 。这样，在层  $l$ ，每道工序同时确定了分配的加工机床及在该机床上开始加工的时间。

(3)一般而言，定向搜索宽度和定向过滤宽度越大，优化结果越好，但搜索时间越长；而定向搜索宽度和定向过滤宽度过小，虽然搜索速度快，但解的性能差。因此，为了平衡搜索时间和解的性能之间的矛盾，在计算过程中，定向搜索宽度将在一定范围内由实验设定，一般不大于 10。

(4)对于评价函数，算法采用 4 个改进的调度规则来构造

局部和全局评价函数。下面描述中， $(O_{ij}, m_k), m_k \in M_{O_{ij}}$  表示工序  $O_{ij}$  分配到机床  $m_k$  上的决定。1)改进的最短加工时间规则(M\_SPT)：选择具有  $\min(p_{ijk})$  的决定  $(O_{ij}, m_k)$ ；若有多个决定可选，则选择当前使用最少的机床；若仍有多个选择，则从中随机选择一个。2)改进的结束时间最早规则(M\_EET)：选择具有  $\min(s_{ijk} + p_{ijk})$  的决定  $(O_{ij}, m_k)$ ；若有多个决定可选，则选择当前使用最少的机床；若仍有多个选择，则从中随机选择一个。3)改进的剩余加工时间最少规则(M\_LWR)：选择具有  $\min(\sum_{q=j}^n \min(p_{iqk}))$  的决定  $(O_{ij}, m_k)$ ；若有多个决定可选，则选择当前使用最少的机床；若仍有多个选择，则从中随机选择一个。4)改进的工序加工时间与作业总加工时间比率最小规则(M\_PT/TOT)：选择具有  $\min(p_{ijk} / \sum_{q=1}^{n_i} \min(p_{qjk}))$  的决定  $(O_{ij}, m_k)$ ；若有多个决定可选，则选择当前使用最少的机床；若仍有多个选择，则从中随机选择一个。

#### 3.2 算法描述

提出的基于过滤定向搜索的集成启发式算法完整步骤描述如下：

(1)初始化：令循环标识  $bn = 0$ ，搜索树层次标识符  $l = 0$ ；输入定向搜索宽度  $b$ ，定向过滤宽度  $f$ ；输入总的工序数量  $T_o$ ；输入  $n$  个作业， $m$  台机床的详细信息(或者通过导入订单信息)，令局部调度  $PS_l = \emptyset$ 。

(2)确定定向搜索节点：

1)通过使用 M\_NDELAY 分枝策略子程序，从根节点处生成分枝节点，并检查生成节点的总数  $N$ 。令  $l = l + 1$ ，用生成的节点更新  $PS_l$ 。

2)如果  $N < b$ ，则继续进入下一层(即  $l = l + 1$ )，采用 M\_NDELAY 子程序生成新的节点(更新节点的总数  $N$ )，更新  $PS_l$ 。如果  $N < b$ ，则转到 2)；否则，转到 3)。

3)计算所有节点的全局评价函数值，并选择最好的  $b$  个节点作为初始定向搜索节点，同时确定局部调度集  $PS_l(1), \dots, PS_l(b)$ 。

(3)  $bn = bn + 1$ ；如果  $bn > b$ ，则转到(4)；否则，继续。

1)从每个初始定向搜索节点开始，生成第  $l$  层的节点： $l = l + 1$ ，如果  $l > T_o$ ，那么转到 2)；否则，继续。

以  $PS_l(bn)$  为局部调度，根据 M\_NONDELAY 分枝策略子程序从定向搜索节点处生成下一层的节点(节点总数为  $N_{bn,l}$ )。计算所有生成节点的局部评价函数值，并选择其中有最好局部评价函数值的  $\min(N_{bn,l}, f)$  个节点，作为下一步全局评价的候选节点。

计算  $\min(N_{bn,l}, f)$  个节点的全局函数值，选择具有最好值的节点加入到局部调度  $PS_l(bn)$  中，更新  $PS_l(bn)$ 。转到 。

2)形成第  $bn$  条完整调度  $PS(bn)$ 。

(4)在最终的  $b$  个调度结果中，选择具有最好目标函数值的调度或者调度集合。停止。

### 4 实验仿真

为了进一步分析该算法对柔性作业车间调度问题的有效性，在配置为 Intel Pentium IV, 512 MB 内存电脑、Windows 2000 平台上，利用 Visual C++ 6.0 实现了该算法，并通过不同实例对算法进行了分析评价。

#### 4.1 算法评价

由于算法的性能与局部和全局评价函数有关,因此首先在相同的定向搜索宽度和定向过滤宽度下评价不同评价函数的组合对算法性能的影响。考虑由上述4种评价函数规则组成的4种不同组合,分别为(L:M\_SPT,G:M\_SPT), (L:M\_EET, G:M\_EET), (L:M\_LWR,G:M\_LWR), (L:M\_PT/TOT,G:M\_PT/TOT),其中,“L”和“G”分别表示局部评价函数和全局评价函数。

测试数据共包括10个问题(问题 Mk01,Mk02,...,Mk10),这些问题的原始数据可从文献[5]上下载。问题都是通过随机数据生成,每个问题的作业数服从10~20之间的离散均匀分布,即  $n \sim U[10,20]$ ,机床数量  $m \sim U[4,15]$ ,每个作业的工序数量  $n_i \sim U[5,15]$ 。由于都是局部柔性作业车间调度问题,因此通过文献[3]的方法(令工序  $O_{ij}$  在不能加工其的机床上的加工时间  $p_{ijk} = +\infty$ )转换为完全柔性作业车间调度问题。测试时,目标函数中权重系数  $w_1, w_2, w_3$  进行了不同组合,定向搜索宽度  $b=2,3,\dots,8$ ,定向过滤宽度  $f=2,3,\dots,8$ 。

测试结果表明,对于相同的  $b$  和  $f$ ,局部评价函数和全局评价函数组合(L: M\_EET, G: M\_EET)的结果最好,而组合(L: M\_LWR, G: M\_LWR)最差。为了进一步说明问题,图2描述了当  $b=5, f=2\sim 8$  时( $w_1=0.4, w_2=w_3=0.3$ ),在不同评价函数组合下10个问题平均目标函数值的变化趋势。

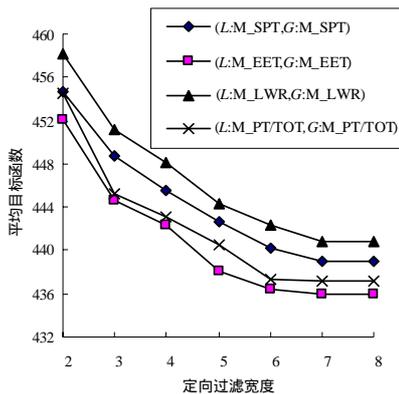


图2 评价目标函数值的比较

#### 4.2 实例验证

利用评价结果,在评价函数组合(L: M\_EET, G: M\_EET)下对文献[3]中2个较大规模的实例进行了验证,并与代表性方法进行了比较。为了平衡搜索速度和解性能,取  $b=5, f=7$ 。

问题1的规模为10个作业,10台机床,共30道工序的完全柔性作业车间调度问题,所有作业的下达时间为  $r_i=0$ 。利用提出的基于过滤定向搜索的启发式算法解得的调度结果如图3所示,得到  $F_1=7, F_2=42, F_3=6$ ,CPU时间为4.2s。与其他方法得到结果的比较见表1,结果可见该算法的有效性。

表1 针对问题1的结果比较

	TD	经典GA	AL	AL+CGA	PSO+SA	本文算法
$F_1$	15	7	8	7	7	7
$F_2$	59	53	46	45	44	42
$F_3$	16	7	6	5	6	6

问题2的规模为15个作业,10台机床和共56道工序的完全柔性作业车间调度问题,具体数据参考文献[3],各个作业的下达时间  $r_i(i=1,2,\dots,15)$  各不相同。运用算法得到的调度结果如图4所示,  $F_1=23, F_2=93, F_3=10$ ,CPU时间为55s。

相比文献[3]利用AL+CGA算法得到的结果( $F_1=23, F_2=95, F_3=11$ )好,可见提出的启发式算法的优越性。

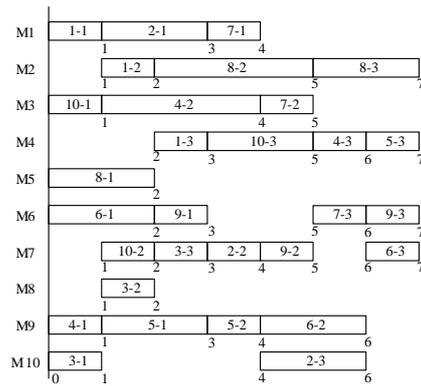


图3 问题1的调度结果

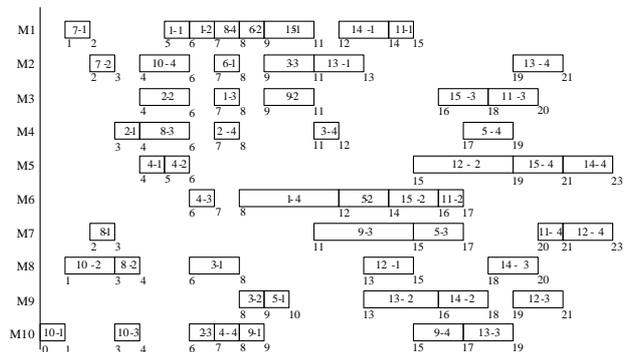


图4 问题2的调度结果

#### 5 结束语

针对makespan 机床总负载和关键机床负载多目标最优的柔性作业车间调度问题,提出了一种基于过滤定向搜索的启发式集成算法,通过设计的节点分枝策略和局部/全局评价函数,能同时解决路径分配和加工排序2大子问题,以此获得优化调度解。通过大量仿真实验,对算法性能进行了分析评价,并通过实例比较表明提出的启发式算法是一种具有潜力的优化算法,具有简单、计算速度快以及效率高等特点,适合解决实际制造环境中的柔性作业车间调度以及其他组合优化问题。

#### 参考文献

- [1] Xia Weijun, Wu Zhiming. An Effective Hybrid Optimization Approach for Multi-objective Flexible Job-shop Scheduling Problems[J]. Computers & Industrial Engineering, 2005, 48(2): 409-425.
- [2] 庞哈利. 柔性Job shop集成化计划调度模型及其求解算法[J]. 控制与决策, 2003, 18(1): 34-39.
- [3] Kacem, I, Hammadi S, Borne P. Pareto-optimality Approach for Flexible Job-shop Scheduling Problems: Hybridization of Evolutionary Algorithms and Fuzzy Logic[J]. Mathematics and Computers in Simulation, 2002, 60(3/5): 245-276.
- [4] Sabuncuoglu I, Bayiz M. Job Shop Scheduling with Beam Search[J]. European Journal of Operational Research, 1999, 118(2): 390-412.
- [5] Mastrolilli M, Gambardella L M. Effective Neighbourhood Functions for the Flexible Job Shop Problem[J]. Journal of Scheduling, 2002, 3(1): 3-20.