

文章编号:1000-582x(2001)01-0020-04

一种新型的单件小批量生产车间调度方法

郑华林, 刘飞, 熊峰, 尹超

(重庆大学 制造工程研究所, 重庆 400044)

摘要:提出了解决具有开完工限制的约束 Job-shop 生产调度问题的一种新方法,该方法将粗排算法有机结合起来,在粗排算法中,系统优先安排机床虚拟工序队列集合中剩余加工时间最大的工序,得到一个初始可行方案,该方案使工件在系统中的流通时间最短。细排算法以粗排算法所得方案为基础,以工件的提前/拖期时间最大的工艺块为调整目标,通过交换工艺块中相邻工序位置的调整策略,得到一个优化的生产作业计划。经分析实例验证,该算法易于实现,具有一定的优越性,并且在工程中是可行的。

关键词:单件车间;调度算法;交货期

中图分类号: **文献标识码:**A

单件车间调度(Job-Shop)是生产计划中的一个重要问题,比较符合工厂车间的实际生产情况,同时也是一个比较复杂且极具代表性的生产调度问题:给定 m 台机器加工 n 个工件,每个工件有其特定的加工顺序或路线,调度目标就是确定每台机器上各工件工序的加工顺序及开工时间,使某个性能指标最优(如制造周期最小),生产调度问题是一种比较典型的组合优化问题,而且大部分是 NP-hard 问题,对于单件车间调度问题,目前的研究成果较多,例如分支定界法、启发式方法^[1]、神经网络方法^[2]以及遗传算法^[3,4]等,这些方法在理论上的探索较多,对于小规模调度应用效果较好。由于单件车间调度问题的复杂性,对于机器数量和工件数量均较多的调度问题,上述算法在计算机上的运行时间较长,工程中应用不太理想。

本文提出了一种新的调度方法,该方法将粗细排算法有机结合起来,在粗排算法中,系统优先安排机床虚拟工序队列集合中剩余加工时间最大的工序,得到一个使工件在系统中的流通时间最短的初始可行方案。细排算法以粗排方案为基础,以工件的提前/拖期时间最大的工艺块为调整对象,得到一个优化的生产作业计划。经分析及实例验证、比较,利用这种算法求

解单件车间调度问题,计算时间短,调度结果令人满意。

1 调度问题的形成

通常,调度意味着在确定时间区段上,为完成某些活动所必需的有限资源的分配方案。本文研究开完工时间受限制的约束 Job-shop 调度问题。

1.1 描述调度问题的变量

n —工件数量; m —机器数量; q_i —工件 i 所包含的工序数; O_{ij}^k —工件 i 的第 j 道在第 k 台机器加工的工序; S_{ij}^k —工序的开始加工时间; S_{i1}^k —工件 i 的首道工序(假定在机器 k 上)的开工时间; S_{in}^k —工件 i 的末道工序(假定在机器 k 上)的开工时间; E_{ij}^k —工序 O_{ij}^k 的完工时间; E_{i1}^k —工序 O_{i1}^k 的完工时间; T_{ij}^k —工序 O_{ij}^k 的工序加工时间; ES_i —工件 i 的最早可能开始时间; D_i —工件 i 的交货期。

注意: k 与 i 和 j 相关。

1.2 约束 Job-shop 调度的数学描述

● 顺序约束——工艺要求的同一工件相邻工序间的加工顺序

收稿日期:2000-09-17

基金项目:国家 863/CIMS 项目资助(课题编号:863-511-910-271)

作者简介:郑华林(1965-),男,四川营山县人,重庆大学制造工程研究所博士研究生。主要从事 CAD/CAPP/CAM/PDM 集成技术的研究。

$$S_{ij}^k - S_{i(j-1)}^k - T_{i(j-1)}^k \geq 0 \quad (1)$$

式(1)表示工件*i*的第*j*道工序必须在第*j*-1道工序完成后才能开始。

● 资源约束——同一台机器*k*上一个加工任务完成之后才能开始另一个任务的加工,设在同一台机器*k*上工序*O_v^k*先于工序*O_p^k*加工,则:

$$S_{pv}^k - S_{v}^k - T_{v}^k \geq 0 \quad (2)$$

式(2)表示任一确定时刻,机器*k*不能同时加工任意两个工件*p*和*v*。

● 开工时间限制——工件的开工时间受到物料的投放时间限制:

$$S_{i1}^k - ES_i \geq 0 \quad (3)$$

● 调度目标——调度的目标是使系统具有较好的加工性能,一方面能使系统具有较高的生产率,即较高的设备利用率,另一方面能及时完成给定的加工任务。在本文中,调度主要考虑2方面的性能,即尽可能减少系统生产时间和尽量满足加工任务中工件的交货期。其调度目标为:

1) 使工件平均流通时间最短:

$$\min y = \max_{i=1, \dots, n} E_{iq}^k \quad (4)$$

工件平均流通时间反映工件在系统中的逗留时间,能较全面地反映系统生产时间。

2) 对工件提前/拖期完工进行的惩罚最小:

$$\min y = \sum_{i=1}^n [r_i \times \max(0, D_i - E_{iq}^k) + w_i \times \max(0, E_{iq}^k - D_i)] \quad (5)$$

其中*r_i*和*w_i*分别是对工件提前和拖期完工的惩罚系数。

上述(1)~(5)式中,

$$i, p \in (1, 2, \dots, n), h, k \in (1, 2, \dots, m), j \in (1, 2, \dots, q_i) \quad (6)$$

2 算法的基本思想与算法描述

文中的算法分初排算法和细排算法两个层次,分别实现目标(4)和目标(5)。

2.1 初排算法

初排算法的目的是实现目标(4),使工件平均流通时间最短,为了描述该算法,先介绍几个基本概念。

定义1 机床虚拟工序队列^[5],制造系统内每一台机床前都存在一队列(集合),该集合中存放的是当前该设备能够加工的工序(机床的可加工工序集合)。

假设在调度开始时,机床虚拟工序队列集合为空(也可按给定系统状态设定),当有新工件进入系统或

设备加工完一工件时,首先判明该工件当前能加工的工序及其对应的机床,然后使这些工序进入对应的机床虚拟工序队列集合中。设备的下一个加工工件是在该机床虚拟工序队列集合中按一定的规则进行选择的。当机床选择好1个工序后,从对应的机床虚拟工序队列集合中删去该工序对应工件的所有工序。

定义2 剩余加工时间,设在某一时刻,工件*i*的前*e*(*e* ≤ *q_i*)道工序已经加工,其工时之和为*TS_e* =

$$\sum_{l=1}^e T_{il}, \text{工件 } i \text{ 的所有工序工时之和为 } T_i = \sum_{l=1}^{q_i} T_{il}, \text{则}$$

工件*i*第*e*道工序的剩余加工时间为:*R_i* = *T_i* - *TS_e*。

在初排算法中,机床虚拟工序队列中各工序的调度规则是优先安排虚拟工序队列集合中剩余加工时间最大的工序。这样,可以保证各工件相对均衡地加工,从而使工件在系统中的平均流通时间最短。

定义3 时间决策点*t_d*,通过计算各机床上所有工件的加工完成时刻,确定具有最小加工完成时间的机床,把该机床当前加工工件的加工完成时刻作为时间决策点*t_d*。该方法一方面全面考虑了各机床的加工负荷情况,另一方面也找出了系统中最迫切需要调度决策的机床进行决策,从而提高整个加工系统的利用率,使总加工时间最短。

具体算法如下:

step 1 初始化,将有关变量赋初值0;

step 2 依次访问各机床*k*,求机床*k*在当前时刻决策点*t_d*的机床虚拟工序队列集合*P_k*(*k* = 1, 2, ..., *m*),设工序*O_v^k*为集合*P_k*中的一个元素,*MT*(*k*)为机床*k*的加工完成时间,则*O_v^k*必须满足下列条件:①工件的前一工序*O_{v(j-1)}^k*已经加工完成;②工序*O_v^k*的开工时间不能大于*t_d*,即:

$$S_{v}^k = \max\{MT(k), E_{v(j-1)}^k\} \leq t_d$$

step 3 若机床虚拟工序队列集合*P_k*中只有一个元素*O_v^k*,则直接选择该元素为机床*k*的当前加工工序,转 step 6,否则,转 step 4;

step 4 分别计算集合*P_k*中各元素对应工件*i*的剩余加工时间*R_i*,并按*R_i*从大到小排序,如果其剩余加工时间相等,则按加工时间大小由小到大排序,也就是当剩余加工时间相等时,优先安排加工时间最短的工序;

step 5 对于经 step 4 排序后的集合*P_k*中相邻的二元素*O_v^k*和*O_p^k*,且工序*O_v^k*排在工序*O_p^k*之前,如果满足下列条件:

$$E_{i(j-1)}^k \geq \max\{MT(k), E_{p(j-1)}^k\} + T_{pv}^k$$

则交换 O_{ij}^k 和 O_{pq}^k 的顺序,也就是说,应该优先安排工序 O_{pq}^k ,以避免机器等待 O_{ij}^k 上一工序而使机器空闲,从而充分利用机床;

step 6 经过上述各步后,设机床 k 选定的加工工序为 O_{ij}^k ,则令:

$$S_{ij}^k = \max\{MT(k), E_{i(j-1)}^k\}$$

$$E_{ij}^k = S_{ij}^k + T_{ij}^k \quad MT(k) = E_{ij}^k$$

step 7 机床是否已经访问完毕,若是,则重新计算时间决策点 t_d, t_d 为该循环中机床的最小加工完成时刻。转 step 8;否则,转 step 2;

step 8 判断各机床虚拟工序队列集合 $P_k (k = 1, 2, \dots, m)$ 是否都不为空,若是,转 step 2,否则,算法结束。

2.2 调节算法

调节算法是在初排算法的基础上,调整机床上相关加工操作的顺序,从而满足目标函数(5),使工件提前/拖期完工进行的惩罚最小。为了实现该算法,设计了时间参数计算器^[6],时间参数计算器是在已知各机床 $k (k = 1, 2, \dots, m)$ 上加工的各工件加工顺序的前提下,推算各工序的开工时间 S_{ij}^k 和完工时间 E_{ij}^k 。其算法步骤如下:

step 1 对于机床 k 的第一个加工操作,如果该加工操作是任一工件 i 的第一道工序,则令 $S_{i1}^k = 0, E_{i1}^k = T_{i1}^k, MT(k) = E_{i1}^k$;否则,寻找工件 i 该道工序 j 在工艺路线中的前道工序 $O_{i(j-1)}^k$,令:

$$S_{ij}^k = E_{i(j-1)}^k, E_{ij}^k = S_{ij}^k + T_{ij}^k, MT(k) = E_{ij}^k$$

step 2 对于机床 k 的第一个加工操作后的各个操作,如果该加工操作是任一工件 i 的第一道工序,则令 $S_{i1}^k = 0, E_{i1}^k = T_{i1}^k, MT(k) = E_{i1}^k$;否则,先找到机床 k 与该操作 O_{ij}^k 的前一操作和工件 i 该道工序 j 在工艺路线中的前道工序 $O_{i(j-1)}^k$,令:

$$S_{ij}^k = \max\{MT(k), E_{i(j-1)}^k\}$$

$$E_{ij}^k = S_{ij}^k + T_{ij}^k \quad MT(k) = E_{ij}^k$$

step 3 如果工件完工时间 E_{iq}^k 小于其交货期 D_i ,且该工序 O_{iq}^k 是机床 k 的最后一个操作,则令: $E_{iq}^k = D_i$ 以减小工件提前/拖期完工进行的惩罚。

该时间参数计算器计算速度快,在后面的调整算法中要反复调用。

本系统调节算法是在初排算法的基础上,调整机床上相关工序的加工顺序,调整范围主要包括两个方面,一是最大拖期工件 i_a 所在加工路线上的工序块,二是最大提前期工件 i_b 所在加工路线上的工序块,调

整策略为工序块中相邻工序之间的交换。

调整算法分为两个层次,第一层次是根据给定的排序结果,交换工序块中相邻工序位置,其步骤为:

step 1 根据给定的排序结果,找出该方案中最大拖期工件 i_a 和最大提前期工件 i_b ;

step 2 找出该方案的搜索邻域,并开始移动操作,对于 i_a ,将 i_a 对应各工序与同一机床上的前一工序交换,而对于 i_b ,则将 i_b 对应各工序与同一机床上的后一工序交换;

step 3 调用时间参数计算器对 step 2 交换后的每个排序计算各工序的开工时间和完工时间,得到相应的生产作业计划方案,计算各方案的目标函数值,并找出该搜索邻域中目标值的最小方案 S 及其对应的目标值 y_i ;

step 4 如果 $y_i < y_{\min}$,则令 $y_{\min} = y_i$,并保存该生产作业计划方案;

step 5 如果循环次数小于给定值,则以 step 3 邻域搜索中目标值最小的方案 S 为待调整的排序结果,转 step 1;否则,找到该循环中的最小目标值 y_{\min} ,第一层次算法结束。

第二层次是对第一层次结果进行相应处理,如果第一层算法所得的最小目标值就是 y_{\min} 系统当前的最小目标值,则根据该排序结果调用第一层次算法继续调整,反之,则产生随机数 i_a 和 i_b ,并以系统当前的最小目标值对应的排序结果调用第一层次算法继续调整。其步骤为:

step 1 调用初排算法所产生的排序结果,并计算该方案的目标函数值 y_i ,令 $y_{\min} = y_i$,并保存该生产作业计划方案,系统开始循环;

step 2 如果循环次数达到给定值,则转 step 4;否则,转 step 3;

step 3 根据 y_{\min} 所对应的排序结果,调用第一层次算法继续调整,如果 $y_{\min} < y_{\min}$,则产生随机数 i_a 和 $i_b (i_a, i_b \in \{1, 2, \dots, n\}, i_a \neq i_b)$ 。

step 4 找出最小目标值 y_{\min} 的所对应的生产作业计划,即为优化了的生产作业计划方案。

3 调度实例

表 1 给出 6×6 Job-shop 调度问题的原始数据, (m, t) 表示工件对应工序,在第 m 台机器上加工,加工时间为 t ;设开工时间无限制,权重 $r_i = w_i = 0.5$ 。图 1 为其初排结果,流通时间是 55 min,为最小值,惩罚值是 11;图 2 为其优化结果,惩罚值是 2,流通时间是 60 min。

表 1 6 × 6 的调度问题原始数据

工件	工序 1	工序 2	工序 3	工序 4	工序 5	工序 6	交货期
工件 1	(3,1)	(1,3)	(2,6)	(4,7)	(6,3)	(5,6)	58
工件 2	(2,8)	(3,5)	(5,10)	(6,10)	(1,10)	(4,4)	56
工件 3	(3,5)	(4,4)	(6,5)	(1,9)	(2,1)	(5,7)	50
工件 4	(2,5)	(1,5)	(3,5)	(4,3)	(5,8)	(6,9)	50
工件 5	(3,9)	(2,3)	(5,5)	(6,4)	(1,3)	(4,1)	52
工件 6	(2,3)	(4,3)	(6,9)	(1,10)	(5,4)	(3,1)	45

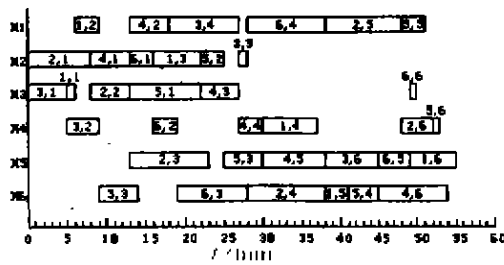


图 1 初排结果(流通时间最小为 55,惩罚值为 11)

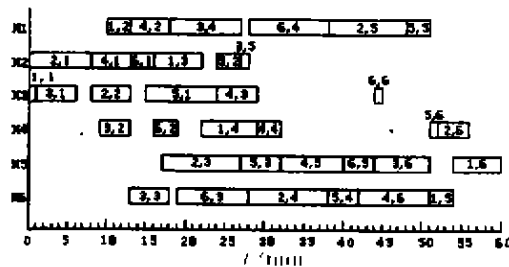


图 2 优化结果(流通时间为 60,惩罚值为 2)

4 结论

根据本文内容,可得出如下结论:

1) 在初排算法中,机床虚拟工序队列中各工序的调度规则是优先安排虚拟工序队列集合中剩余加工时间最大的工序。这样,既保证各工件相对均衡地加工,又保证每个工件加工工序的偏序关系,从而使工件在系统中的平均流通时间最短。

2) 开发了一种时间参数计算器,它可以在已知排序和各工序加工时间的条件下,计算出各工序的开工时间和完工时间。

3) 在调整算法中,它以工件的提前/拖期时间最小为目标,可以对初排算法产生的初始生产作业计划方案进行优化。理论分析和实践表明它可以得到最优或高质量的近似最优调度解,具有实用意义。

参考文献:

- [1] 毛宁,刘霞 关于 M 类机器的非流水复杂作业排序问题的一种启发式算法[J]. 工程数学学报,1996,(2):65-73
- [2] 张长水,阎平凡. 解 Job-Shop 调度问题的神经网络方法[J] 自动化学报,1995,(11):706-712
- [3] 张长水,沈刚,阎平凡. 解 Job-Shop 调度问题的遗传算法[J] 电子学报,1995,(7):1-5
- [4] LEE C Y, CHOI J Y. A genetic algorithm for job sequencing problems with distinct due dates and general early-tardy penalty weights[J]. Computer Ops. Res, 1995, 22:857-869
- [5] 赵天奇,陈禹六,李培根. 基于零件虚拟工序队列的 FMS 动态调度研究[J]. 中国机械工程,1999,(12):1 367-1 3669.
- [6] 胡斌,黎志成. 单件车间生产作业计划再生方法研究[J]. 华中理工大学学报,1999,(2):16-18

A Study on the Job-shop Scheduling Problem

ZHENG Hua-lin, LIU Fei, XIONG Feng, YIN Chao

(Insitute of manufacturing engineering of Chongqing University, Chongqing 400044, China)

Abstract: This paper discusses a new approach of job-shop production scheduling with the time limit of starting and completing work. In this approach, all operations in the queue are sorted according to their values of remaining operating time, and a preliminary feasible solution of production scheduling is achieved, which makes a shortest circulation time of parts in the system. Then adjustment is made aiming at maximum ahead/behind time span of operation block. The process is iterated until a satisfactory scheduling solution is obtained. Through analysis and application proof, the scheduling algorithm is suitable and feasible in engineering projects.

Key words: job-shop; scheduling algorithm; due date

(责任编辑 成孝义)