

基于遗传算法的柔性作业车间柔性分批调度问题研究

周超¹, 胡燕海^{2,3,4*}, 李建辉²

(1. 宁波大学 机械工程与力学学院, 浙江 宁波 315211; 2. 宁波大学 海运学院, 浙江 宁波 315211; 3. 国家道路交通安全管理工程技术研究中心宁波大学分中心, 浙江 宁波 315211; 4. 现代城市交通技术江苏高校协同创新中心, 浙江 宁波 315211)

摘要: 针对柔性作业车间柔性分批调度问题的复杂性, 采用遗传算法进行求解, 在引入柔性分批方法后, 并提出了染色体的两级编码方法, 设计了2种交叉和变异操作方法, 以防止操作中非法解的产生, 提高求解效率. 该算法不仅解决批量分批, 而且也解决子批排序和选择加工路线的问题. 对比整批调度和柔性分批调度的 Makespan 后, 仿真结果表明, 柔性分批调度可缩短生产周期, 验证了采用新算法的可行性和有效性.

关键词: 遗传算法; 柔性作业车间; 调度; 柔性分批

中图分类号: TP301

文献标志码: A

文章编号: 1001-5132 (2015) 02-0128-05

柔性作业车间柔性分批调度问题是传统柔性作业车间调度问题的进一步延伸. 该调度问题在传统柔性作业车间调度问题基础上考虑了批量, 即同种工件加工时, 可以分为若干个调度子批进行加工, 对应的子批批量是可变的, 将分批后的所有子批进行排序, 并给各个子批选取合理的加工路线. 由于传统柔性作业车间调度的研究忽略工件批量, 使调度结果不能很好地应用在生产实践中^[1-4]. 而柔性作业车间分批调度增加批量柔性分批的子问题, 增加该调度问题的复杂性, 使算法设计变得更加复杂.

由于柔性作业车间柔性分批调度问题具有重要的研究价值. 近年来, 学者们对柔性作业车间分批调度进行了研究. Low 等^[5]通过仿真验证了工件分批调度可缩短机床等待时间, 提高机床利用率及缩短生产周期, 但未提出如何分批. 孙志峻等^[6]针对单一路径的柔性作业车间调度问题, 提出一种动态分批调度方法, 但工件加工路线缺乏柔性, 算法缺乏说服力. 曾强等^[7-9]针对作业车间提出了等量分批方法, 缩短生产周期, 但工件批量缺乏柔性. 针对以上算法不足, 笔者提出柔性分批调度优化算法, 确定子批最优子批批量与最佳加工路线,

该算法能够减少机床闲置时间, 缩短制造周期.

1 柔性作业车间分批调度

1.1 问题描述

柔性作业车间柔性分批调度问题描述: 1个加工车间有 M 台机床, 现有待加工工件 N 种, 每种工件 Q 个, 每种工件被分为固定子批和对应的随机批量的多个调度子批^[10]. 每类工件有确定的工艺次序, 任一调度子批中的任意1道工序可在多台机床上进行加工. 调度目标是将所有调度子批进行排序, 并给任一调度子批选取合适的加工路线, 使得生产周期最小.

在加工过程中须满足以下约束: (1)在零时刻, 所有工件均可以在被选择机床上进行加工; (2)若2个子批选择同台机床进行加工, 子批加工须有先后顺序, 不能同时进行加工; (3)同一子批的工件进行加工时, 前道工序加工结束才能进行后道工序; (4)同台机床不能同时加工2个工序.

1.2 数学模型

变量定义: N 为待加工工件种类; Q_i 为第 i 种工件的数量; M 为该作业车间所具有机床总数; K 为从 M 台机床中选择 K 台机床; n_i 为第 i 种工件子批

收稿日期: 2014-03-20.

宁波大学学报(理工版)网址: <http://journal.lg.nbu.edu.cn/>

基金项目: 浙江省科技厅重点软科学项目(2010C35022); 宁波市自然科学基金(2013A610285).

第一作者: 周超(1989-), 男, 浙江江山人, 在读硕士研究生, 主要研究方向: 企业信息化及智慧物流. E-mail: zhouchao5591@163.com

*通信作者: 胡燕海(1966-), 男, 浙江舟山人, 博士/教授, 主要研究方向: 企业信息化及智慧物流. E-mail: huyanhai@nbu.edu.cn

总数; J_i 为第 i 中工件的子批; L_i 为 J_i 子批对应的批量; o_i 为第 i 中工件的工序数; F_i 为第 i 种工件的完工时间; S_{ijok} 为第 i 种工件的 j 子批的第 o 道工序在选定的设备 k 上开始加工的时间; F_{ijok} 为第 i 种工件的 j 子批的第 o 道工序在选定的设备 k 上结束加工的时间; t_{ijok} 为第 i 种工件的 j 子批的第 o 道工序在选定的设备 k 上单件加工时间; C 为所有工件的最后结束加工时间;

根据上述柔性作业车间分批调度问题, 建立符合实际的数学模型, 模型如下:

$$\min(C) = \max_{i=1,2,\dots,N} F_i. \quad (1)$$

s.t.,

$$Q_i = \sum_{i=1}^{n_i} L_i \quad (2)$$

$$X_{ijk} = \begin{cases} 1, & \text{第 } i \text{ 种工件的 } j \text{ 子批的第 } o \text{ 道工序} \\ & \text{在选定机床 } k \text{ 上进行加工,} \\ 0, & \text{否,} \end{cases} \quad (3)$$

$$Y_{iefgk} = \begin{cases} 1, & \text{第 } j \text{ 子批的第 } e \text{ 道工序和第 } f \text{ 子批} \\ & \text{的第 } g \text{ 道工序都在选定的机床 } k \\ & \text{上进行加工, 并且工序 } e \text{ 先于工} \\ & \text{序 } g \text{ 加工,} \\ 0, & \text{否,} \end{cases} \quad (4)$$

$$S_{ij(o+1)k} \leq S_{ijok} + t_{ijok} \times L_i \quad \forall X_{ijok} = X_{ij(o+1)k} = 1, \quad (5)$$

$$S_{ij(o+1)k} \leq S_{ijom} + t_{ijom} \times L_i \quad \forall X_{ij(o+1)k} = X_{ijom} = 1, k \neq m, \quad (6)$$

$$S_{ifgk} \leq S_{hjek} + t_{hjek} \times L_h \quad \forall Y_{jefgk} = 1, i \neq h, \quad (7)$$

其中, (1)式表示通过不同的调度方案, 比较所有工件最后结束加工时间, 取出最小结束加工时间对应的调度方案. (2)式表示将每种工件分为固定子批和对应的随机批量; (3)式当 $X_{ijok}=1$ 时, 表示第 i 种工件的 j 子批的第 o 道工序在选定的机床 k 上进行加工; (4)式当 $Y_{jefgk}=1$, 表示第 j 子批的第 e 道工序和第 f 子批的第 g 道工序都在选定的机床 k 上进行加工, 并且工序 e 先于工序 g 加工. (5)式表示同一子批中的工件选择在同台机床上进行加工, 它的后道工序需要前道工序加工完成后才能进行加工; (6)式表示同一子批中的工件选择在不同机床上进行加工, 它的后道工序需要前道工序加工完成后才能进行加工; (7)式表示同台机床 k 不能同时加工 2 个工序.

2 遗传算法

2.1 柔性分批方法

在柔性作业车间分批调度问题中, 生产批量与生产周期呈 U 型关系, 过大与过小的生产批量都会导致较长生产周期^[11]. 适当批量分割可提高生产效率, 缩短周期. 柔性分批策略如下: 将每种工件均分为 n 个子批, 同种工件的 n 个子批对应的批量随机产生. 每种工件的子批总数 n 由决策者确定, 而子批批量由算法随机产生. 1 种工件批量为 m , 在 1 至 $m-1$ 内产生 $n-1$ 个不同随机数, 这样便可将这类工件分为 n 个随机批量的子批. 若 $m=8$, $n=2$, 产生的随机数为 3, 则得到(3,5) 2 个子批.

2.2 编码方法

柔性作业车间柔性分批调度问题需要将每种工件随机分为若干个调度子批, 并将所有调度子批排序, 给任一调度子批选择合理的加工路线. 因此笔者将柔性作业车间柔性分批调度的编码分为两部分: 一部分基于工序的编码, 确定所有调度子批中工件加工的先后顺序; 另一部分是基于机床的编码, 即为所有调度子批中工件的任一道工序确定加工机床. 通过这两部分的编码, 得出柔性作业车间调度柔性分批调度问题的可行解. 表 1 是个传统的单件 FJSP 调度问题, 不涉及批量^[12-13]. 笔者在表 1 的基础上增加批量, 假设每种工件有 8 个. 以表 1 为例, 具体描述染色体的编码方法.

表 1 4×6 型 FJSP 可选机床和加工时间表

工件	工序	可选加工机床					
		M1	M2	M3	M4	M5	M6
N1	O ₁₁	2	3	4	*	*	*
	O ₁₂	*	3	*	2	4	*
	O ₁₃	1	4	5	*	*	*
N2	O ₂₁	3	*	5	*	2	*
	O ₂₂	4	3	*	*	6	*
	O ₂₃	*	*	4	*	7	11
N3	O ₃₁	5	6	*	*	*	*
	O ₃₂	*	4	*	3	5	*
N4	O ₃₃	*	*	13	*	9	12
	O ₄₁	9	*	7	9	*	*
	O ₄₂	*	6	*	4	*	5
	O ₄₃	1	*	3	*	*	3

注: *表示对应的工序不能在该机床进行加工.

若将每种工件随机分为2个调度子批,即(3,5)2个调度子批(每种工件在分成2个子批情况下,不一定是(3,5)2个子批,为阐述方便,假设4种工件都被分成(3,5)2个调度子批).这样得到8个调度子批.假定1条染色体基于工序的编码为1-2-3-4-5-6-7-8-7-8-5-6-1-3-2-4-4-3-2-7-5-1-8-6.对应的工序 $O_{11}-O_{21}-O_{31}-O_{41}-O_{51}-O_{61}-O_{71}-O_{81}-O_{72}-O_{82}-O_{52}-O_{62}-O_{12}-O_{32}-O_{22}-O_{42}-O_{43}-O_{33}-O_{23}-O_{73}-O_{53}-O_{13}-O_{83}-O_{63}$.在基于工序编码中1至8的具体意义在表2中给予具体说明.

表2 基于工序编码中数字1~8对应的调度子批

工序编码	子批
1	工件1的第1个子批
2	工件2的第1个子批
3	工件3的第1个子批
4	工件4的第1个子批
5	工件1的第2个子批
6	工件2的第2个子批
7	工件3的第2个子批
8	工件4的第2个子批

另一条染色体基于机床的编码为[1 2 1; 5 2 3; 1 2 5; 3 4 1; 3 5 3; 1 1 3; 2 5 5; 1 4 6](基于机床的编码,以Matlab中的矩阵方式表达),机床编码矩阵对应的第1行代表工件1的第1个子批选定的加工机床为M1-M2-M1,对应的时间为6-9-3,第5行代表工件1的第2个子批选定加工机床为M3-M5-M3,对应的时间为20-20-25.后面的以此类推.

2.3 交叉操作

针对柔性作业车间分批调度问题不仅要确定所有子批中工件的先后加工顺序,还须确定所有子批中工件的工序所选择机床.工序编码采用单亲双点交叉,机床编码采用双亲行交叉.

2.3.1 基于工序编码染色体的交叉操作

基于工序编码染色体的交叉操作如图1所示,具体操作:从初始种群中随机取出1个染色体,选取染色体中的任意2个基因进行互换(箭头表示选中的2个基因),便可得到1个新的染色体.

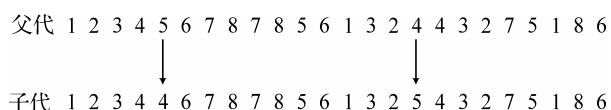


图1 基于工序编码染色体的交叉操作

2.3.2 基于机床编码染色体的交叉操作

取出初始种群(种群个数为偶数)中连续的2个染色体——父代1、父代2,得到2个机床编码的矩阵.将任何一类工件中对应的所有子批进行行互换,如果选中工件1的所有子批互换,即父代1的第1行和第5行与父代2的第1行和第5行.交叉操作如下:

$$\begin{array}{l}
 \text{父代 1:} \begin{bmatrix} 1 & 2 & 1 \\ 5 & 2 & 3 \\ 1 & 2 & 5 \\ 3 & 4 & 1 \\ 3 & 5 & 3 \\ 1 & 1 & 3 \\ 2 & 5 & 5 \\ 1 & 4 & 6 \end{bmatrix}, \quad \text{父代 2:} \begin{bmatrix} 1 & 4 & 1 \\ 5 & 1 & 6 \\ 2 & 4 & 6 \\ 3 & 6 & 6 \\ 3 & 4 & 3 \\ 1 & 1 & 6 \\ 2 & 4 & 6 \\ 1 & 4 & 6 \end{bmatrix}, \\
 \text{子代 1:} \begin{bmatrix} 1 & 4 & 4 \\ 5 & 2 & 3 \\ 1 & 2 & 5 \\ 3 & 4 & 1 \\ 3 & 4 & 3 \\ 1 & 1 & 3 \\ 2 & 5 & 5 \\ 1 & 4 & 6 \end{bmatrix}, \quad \text{子代 2:} \begin{bmatrix} 1 & 2 & 1 \\ 5 & 1 & 6 \\ 2 & 4 & 6 \\ 3 & 6 & 6 \\ 3 & 5 & 3 \\ 1 & 1 & 6 \\ 2 & 4 & 6 \\ 1 & 4 & 6 \end{bmatrix}.
 \end{array}$$

2.4 变异操作

2.4.1 基于工序编码染色体的变异操作

基于工序编码染色体的变异操作如图2所示,变异操作具体步骤如下:在种群中任取1个染色体作为父代1,将染色体中所有基因进行倒序处理得到子代1~,然后再随机选择染色体中任意1个基因插入到染色体的最前面,其余基因顺序保持不变,得到子代1.假设随机产生的基因位是第10位,则选择子代1~的第10位基因插入到最前面,得到子代1.

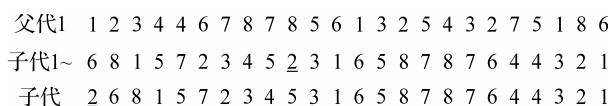


图2 基于工序编码染色体的变异操作

2.4.2 基于机床编码染色体的变异操作

在初始种群中任意取出父代1,将任何类工件中所有子批进行与行的互换.若选中工件2的所有子批互换,交换第2行与第6行.变异操作如下:

$$\text{父代 1:} \begin{bmatrix} 1 & 4 & 1 \\ 5 & 2 & 3 \\ 1 & 2 & 5 \\ 3 & 4 & 1 \\ 3 & 4 & 3 \\ 1 & 1 & 3 \\ 2 & 5 & 5 \\ 1 & 4 & 6 \end{bmatrix}, \quad \text{子代 1:} \begin{bmatrix} 1 & 4 & 1 \\ 1 & 1 & 3 \\ 1 & 2 & 5 \\ 3 & 4 & 1 \\ 3 & 4 & 3 \\ 5 & 2 & 3 \\ 2 & 5 & 5 \\ 1 & 4 & 6 \end{bmatrix}.$$

2.5 选择操作

笔者采用锦标赛选择方法. 锦标赛选择就是从种群中随机选择2个个体, 通过比较2个的适应度值, 保留优胜个体, 淘汰劣质个体, 直至种群个数与父代种群个数相同, 停止选择. 在种群初期, 采用锦标赛选择法, 选择最优个体, 而当连续几代的最优个体适应度仍未更新, 产生一定比例的新个体, 替换旧个群, 避免过早收敛.

2.6 算法流程

步骤1 初始化参数, 包括种群规模、迭代次数、交叉概率和变异概率.

步骤2 产生初始种群, 随机生成种群规模为 N 的初始种群.

步骤3 交叉操作. 将种群进行两两交叉, 这样便得到 N 个子代, 合并父代和子代的种群, 得到1个种群规模为 $2N$ 的新种群.

步骤4 将新生成的种群进行变异操作.

步骤5 选择新种群的优胜个体. 将新种群采用锦标赛选择, 选择出 N 个优胜个体.

步骤6 判断算法是否到达终止条件, 若满足终止条件, 转步骤7, 若不满足, 将步骤5产生的 N 个个体作为初始种群, 并转步骤3.

步骤7 输出优化结果.

3 数值模拟

对某个 4×6 型(4种工件, 6台机床)的柔性作业车间问题进行计算机仿真, 参数使用文献[12]和文献[13]的实例数据, 遗传算法参数设置: 交叉概率为 0.85, 变异概率为 0.1. 现假设每种工件需要生产8个, 比较整批调度和柔性分批调度的生产周期. (1)整批调度的调度方案及生产周期, 文献[12]与文献[13]中的实例数据, 即文中的表1, 文献[12]与文献[13]中得出的最优解均为17, 但表1是单件

调度, 而现在每种工件需要生产8个. 整批调度是将每种工件当做是1个工件, 实质就是单件调度, 每道工序在可选择机床上的时间是原来单件调度的8倍, 故整批调度的最优的 Makespan 为 $17 \times 8 = 136$. (2)柔性分批调度的调度方案及生产周期($n=2$).

表3 4×6 型柔性作业车间分批调度方案

工件	工序	选定加工机床					
		M1	M2	M3	M4	M5	M6
N1	O ₁₁	2×3	-	-	-	-	-
	O ₁₂	-	-	-	2×3	-	-
	O ₁₃	-	4×3	-	-	-	-
N2	O ₂₁	-	-	-	-	2×7	-
	O ₂₂	-	3×7	-	-	-	-
	O ₂₃	-	-	4×7	-	-	-
N3	O ₃₁	5×5	-	-	-	-	-
	O ₃₂	-	-	-	3×5	-	-
	O ₃₃	-	-	-	-	9×5	-
N4	O ₄₁	9×1	-	-	-	-	-
	O ₄₂	-	-	-	4×1	-	-
	O ₄₃	-	-	-	-	-	3×1
N5	O ₅₁	2×5	-	-	-	-	-
	O ₅₂	-	3×5	-	-	-	-
	O ₅₃	1×5	-	-	-	-	-
N6	O ₆₁	3×1	-	-	-	-	-
	O ₆₂	4×1	-	-	-	-	-
	O ₆₃	-	-	-	-	-	11×1
N7	O ₇₁	-	6×3	-	-	-	-
	O ₇₂	-	4×3	-	-	-	-
	O ₇₃	-	-	-	-	-	12×3
N8	O ₈₁	-	-	7×7	-	-	-
	O ₈₂	-	-	-	4×7	-	-
	O ₈₃	1×7	-	-	-	-	-

注: 调度结果为 $O_{71}-O_{31}-O_{21}-O_{72}-O_{51}-O_{81}-O_{11}-O_{22}-O_{41}-O_{32}-O_{61}-O_{62}-O_{12}-O_{13}-O_{82}-O_{73}-O_{52}-O_{83}-O_{63}-O_{42}-O_{43}-O_{53}-O_{33}-O_{23}$, 工件1分为(3,5)2个子批, 工件2分为(7,1)2个子批, 工件3分为(5,3)2个子批, 工件4分为(1,7)2个子批.

对比整批调度和柔性分批调度, 得出柔性分批调度的 Makespan 为 89, 优于整批调度的 136.

4 结语

柔性作业车间柔性分批调度问题是在柔性作业车间调度问题上进行拓展与延伸, 符合目前作业车间生产的实际情况. 笔者采用遗传算法解决

柔性作业车间柔性分批调度问题, 引入柔性分批策略, 使调度结果偏向决策者偏好. 对比整批调度方案和柔性分批调度方案, 仿真结果表明, 柔性分批调度能缩短生产周期, 显示所提出方法的有效性, 具有一定的应用价值.

后续研究的主要方面有: (1)优化模型, 添加模型中未考虑的关键影响因素, 如批量启动时间, 批量切换时间. (2)研究多目标的柔性作业车间批量调度问题, 如添加模糊作业时间及模糊交货期等目标函数.

参考文献:

- [1] 王云, 冯毅雄, 谭建荣, 等. 柔性作业车间分批调度多目标优化方法[J]. 浙江大学学报: 工学版, 2011, 45(4): 719-726.
- [2] 黄刚, 庞志国, 胡振华, 等. 基于遗传与模拟退火算法的分批作业调度[J]. 机械与电子, 2007, 16(5):6-9.
- [3] 李风光, 初红艳, 李鹏, 等. 运用混合优化算法的作业车间分批调度[J]. 现代制造工程, 2010, 18(8):51-54.
- [4] 白俊杰, 龚毅光, 王宁生, 等. 批量生产柔性作业车间优化调度研究[J]. 机械科学与技术, 2010(3):293-298.
- [5] Low C Y, Hsu C M, Huang K I. Benefits of lot splitting in job-shop scheduling[J]. International Journal of Advanced Manufacturing Technology, 2004, 24(9/10):773-780.
- [6] 孙志峻, 安进, 黄卫清. 作业车间多工艺路线批量作业计划优化[J]. 中国机械工程, 2008, 23(2):183-187.
- [7] 曾强, 杨育, 程博, 等. 平顺移动下等量分批 FJSP 多目标优化研究[J]. 系统仿真学报, 2012(5):1046-1052.
- [8] 曾强, 杨育, 王小磊, 等. 并行机作业车间等量分批多目标优化调度[J]. 计算机集成制造系统, 2011, 17(4): 816-825.
- [9] 曾强, 杨育, 王勇智, 等. 复合工艺流程下批量生产车间调度多目标优化[J]. 中国机械工程, 2011(2):190-196.
- [10] 吴秀丽, 李苏剑, 杜彦华. 柔性作业车间多品种小批量调度算法研究[J]. 中国机械工程, 2010, 21(4):424-429.
- [11] 白俊杰, 龚毅光, 王宁生, 等. 多目标柔性作业分批优化调度[J]. 计算机集成制造系统, 2010, 16(2):396-403.
- [12] 席卫东, 乔兵, 朱剑英. 基于改进遗传算法柔性作业车间调度[J]. 哈尔滨工业大学学报, 2007, 39:1151-1153.
- [13] 杨晓梅, 曾建潮. 遗传算法求解 Job-Shop 调度问题[J]. 控制与决策, 2004, 19(10):1197-1200.

Genetic Algorithm Based Flexible Job-shop Scheduling and Flexible Size-lot Splitting

ZHOU Chao¹, HU Yan-hai^{2,3,4*}, LI Jian-hui²

(1.Faculty of Mechanical Engineering and Mechanics, Ningbo University, Ningbo 315211, China;

2.Faculty of Maritime and Transportation, Ningbo University, Ningbo 315211, China;

3.National Traffic Management Engineering & Technology Research Centre Ningbo University Sub-centre, Ningbo 315211, China;

4.Jiangsu Province Collaborative Innovation Center for Modern Urban Traffic Technologies, Ningbo 315211, China)

Abstract: In this paper, genetic algorithm is employed to solve the complex problems in flexible job-shop scheduling and size-lot splitting. A two-stage encoding method of chromosomes is presented and two kinds of crossover and mutation methods are designed to prevent the incorrect solutions in the manipulation. The algorithm not only solves the problem with lot splitting, but also the problem of sorting and selection of flexible size-lot splitting processing route. Based on the Makespan comparison of entire batch scheduling and flexible size-lot splitting, it is found that the production cycle with flexible size-lot splitting can be shortened using the proposed method, with which the feasibility and effectiveness of the proposed method are hence verified.

Key words: genetic algorithms; flexible job-shop problem; scheduling; flexible size-lot splitting

(责任编辑 章踐立)