

文章编号: 1001-0920(2004)10-1197-04

遗传算法求解柔性 job shop 调度问题

杨晓梅, 曾建潮

(太原重型机械学院 系统仿真与计算机应用研究所, 山西 太原 030024)

摘 要: 在分析柔性 job shop 调度问题特点的基础上, 提出一种新的求解该问题的遗传算法, 即利用编码方法表示各工序的优先调度顺序及工序的加工机器, 由此产生可行的调度方案, 使得问题的约束条件在染色体中得以体现。所设计的遗传算子不仅能避免非法调度解的出现, 保证后代的多样性, 而且可使算法具有记忆功能。仿真结果证明了该算法的有效性。

关键词: 遗传算法; 柔性 job shop 调度; 编码

中图分类号: TP18 **文献标识码:** A

Solving flexible job shop scheduling problem using genetic algorithm

YANG Xiaomei, ZENG Jianchao

(Division of System Simulation and Computer Application, Taiyuan Heavy Machinery Institute, Taiyuan 030024, China Correspondent: YANG Xiaomei, E-mail: yxm001-73@sohu.com)

Abstract Based on the analysis of the characteristic of flexible job shop scheduling problem, a new genetic algorithm is presented. The encoding method represents the priority dispatching sequence and machines where each operation can be processed. Then, the feasible schedules are found and the constraints are explained in the chromosome. The genetic operators not only avoid the illegal schemes and ensure the diversification of evolving descendants, but also make the algorithm have memory function. The simulation results validate the effectiveness of the proposed algorithm.

Key words: genetic algorithm; flexible job shop scheduling problem; encode

1 引 言

利用遗传算法研究 job shop 调度问题, 大都集中在以传统的 job shop 调度问题为背景, 解决每道工序的加工机器被唯一确定的调度问题, 由此与现有算法进行对比, 以验证算法的性能^[1,2]。而在实际生产中, 由于机器的数目有限, 各机器的功能不同, 各道工序的加工机器可以动态改变。柔性 job shop 调度问题就是每道工序的加工机器可予以选择的调度问题, 其自身的性质决定了柔性 job shop 调度问

题更接近于实际生产环境

本文首先对柔性 job shop 调度问题进行阐述; 然后在分析该问题特点的基础上, 提出一种基于遗传算法的新的求解方法。该算法具有如下特点: 1) 问题的约束条件体现在染色体中; 2) 通过分部编码相继确定各工序的加工机器及所有工序的优先调度顺序, 染色体编码精简, 能唯一确定一个有效的调度解; 3) 利用分部遗传算子的操作产生可行的染色体后代, 有效地保留了父代的优良性质, 并将记忆功能

收稿日期: 2003-10-29; 修回日期: 2004-02-11

基金项目: 山西省自然科学基金资助项目(20031041)

作者简介: 杨晓梅(1973—), 女, 陕西临潼人, 硕士生, 从事调度问题、智能算法的研究; 曾建潮(1963—), 男, 陕西大荔人, 教授, 博士生导师, 从事智能算法、计算机仿真等研究

引入算子的设计,减少了算法对相似个体的重复操作次数

2 柔性 job shop 调度问题

2.1 柔性 job shop 调度问题描述

所谓柔性 job shop 调度问题(FJSP),就是研究 n 个工件在 m 台机器上加工.已知各工件的加工程序数,每道工序可在一台或多台可供选择的机器上加工,工序的加工时间随机器的性能不同而不同,在满足一定的约束条件下,如何对 n 个工件的加工顺序进行排序,以使某个性能指标最小(或最大).

该问题所需满足的约束条件是:1)所有工件在初始时刻都可加工;2)工序在可供选择的若干机器上加工的时间已确定;3)每台机器在固定的时间段内只能加工一个工件;4)每个工件在固定时刻只能在一台机器上加工

2.2 柔性 job shop 调度问题的特点

研究表明,通过对问题的了解可以节省计算时间^[3,4].因此首先对 FJSP 调度问题加以分析,以利于在求解过程中充分利用问题的特点,提高 FJSP 调度问题的求解效率

一般而言,job shop 调度问题是 NP 难题,随着调度问题规模的增大,计算量会出现“指数爆炸”现象.FJSP 问题不仅增加了机器约束条件,而且扩大了最优调度的搜索空间,并在求解中需要满足更多的约束条件,导致问题更加复杂.由于工序加工路径不确定,工序的加工机器可随机变化,体现出 FJSP 问题动态随机性的特点.总之,FJSP 问题的复杂性、约束性、动态性决定了对该问题的研究能够反映实际生产过程,具有一定的实际意义

3 算法描述

FJSP 问题求解的关键在于确定一个任务的起始加工时刻和各工序的加工机器,并保证所有工件在规定的时间内完成,而遗传算法求解实际问题的关键是编码方法的表示和遗传算子的设计.本文针对问题的特点,提出一种新的编码方法,在此基础上设计相应的遗传算子

3.1 编码方法和解码算法

3.1.1 编码方法

编码过程由两部分组成:第 1 部分是对机器顺序进行编码,选择各道工序可行的加工机器;第 2 部分是根据一定的编码顺序选择下道应加工工序.设 l 为所有工序的总数,下面以生产周期为评价指标,对编码方法的意义进行阐述

(1) 确定工序加工的机器

这部分染色体由 l 个基因构成,各向量表示位于这道工序加工时的机器号.编码方法如下:

1) 初始化参数:对于 l 道工序,形成 l 个可选择机器的子集分别为 S_1, S_2, \dots, S_l ,确定各工序可选择机器总数的集合 $M_o = \{m_{o1}, \dots, m_{oi}, \dots, m_{ol}\}$.如第 1 道工序有 3 台机器作为可选择机器,则 $m_{o1} = 3$

2) 对于集合 S_1 ,设 $S_1 = \{a, c, f\}$ (即第 1 道工序有 a, c, f 3 台机器作为可选机器),根据随机整数 $r_1(1 \leq r_1 \leq m_{o1})$,从集合 S_1 中确定第 1 道工序加工时所用机器 m_1 .如设 $r_1 = 1$,则机器 a 为第 1 道工序加工所用的机器

3) 对于集合 S_2, S_3, \dots, S_l ,分别重复步骤 2),依次确定第 2, 3, ..., l 道工序加工所用的机器 m_2, m_3, \dots, m_l

4) 机器染色体由 l 个基因组成,设 M_c 为所需确定的染色体编码,则 $M_c = (m_1, m_2, \dots, m_l)$.

通过这部分编码可得到基于机器编码的染色体.由于各机器的确定均从各工序对应的可供选择的机器中选取,所得到的染色体满足问题中限定的机器约束条件

(2) 确定未安排工序加工的优先顺序

这部分染色体的全部基因是 l 个不重复的自然数表示的工序号,所有基因的先后顺序表示各工序加工时的优先调度顺序.即当未安排工序中有多道工序均符合进行调度的要求时,为避免可安排工序的冲突,按本行向量中各工序的顺序从前到后作为排序的依据

通过这部分编码可得到基于工序加工顺序的染色体,该染色体能满足问题中所限定的工序约束条件

分部编码方法是针对 FJSP 问题特点而提出的,通过二次编码的方法确定加工的机器及工序优先调度顺序.由于在编码中体现了问题的约束条件,保证了经编码后染色体的有效性

3.1.2 解码算法

对于分部编码方法,解码算法在确定工序应加工机器的基础上设计如下:

1) 根据工序优先调度顺序的染色体编码确定第 1 道应加工工序.定义 $s_1 = 0, c_1 = s_1 + p_1$

2) 设机器当前可加工工件时间为 t ,各工序加工时间为 p_j .对于尚未排序的工序,定义工序的开始加工时间 $s_j = \max\{r_j, t\}$,完工时间 $c_j = s_j + p_j$.利用第 2 部分染色体依次从可安排工序集中提取出当前应加工工序,根据上面两个公式确定各工序的

加工时间和完工时间

3) 计算目标函数值(本文为最大生产周期) $C_{max} = C_l$, 其中 C_l 为最后一道工序的完工时间

3.2 遗传算子

鉴于遗传编码方法中第 1 部分采用基于机器选择顺序的编码, 第 2 部分采用基于工序优先顺序的编码, 对于不同的编码链, 遗传算子的设计方法应有所区别 这里设计了相应的遗传算子, 具体设计方法如下:

3.2.1 交叉算子

第 1 部分: 采用一种新的两点交叉的方法, 该方法不仅操作简单, 而且保证产生的后代是可行的 交叉算法的过程是: 对于选定的两个父体, 随机选择两个交叉点; 然后将两个父体位于两个交叉点之外的两部分基因互换, 得到两个交叉后代

这种交叉方法得到的染色体仍然具有多样性, 而且符合实际问题的各种约束条件

第 2 部分: 基于工序优先顺序的染色体由 l 个不可重复的自然数组成, 可行的交叉后代中的基因也应为 l 个不可重复的自然数 本文采用基于位置的交叉方法产生交叉后代, 即根据交叉概率 P_c 选择需要交叉的个体, 对于每两个交叉父代, 随机选取一些位置, 交换两个父代位于被选中位置上的基因, 同时删除原一个父代从另一父代交换过来的基因, 并从第 1 个基因的位置开始依次填入未选中位置的基因

分部交叉过程不仅利用了柔性调度问题的特点, 而且交叉时引入了通用的交叉方法, 因而操作简便, 保证所产生的交叉后代既能继承父代个体的特点, 又体现出后代适应环境而产生新的性质

3.2.2 变异算子

第 1 部分: 目前的变异算子如逆序、插入等变异均是基于排序的操作, 并没有结合调度问题的特点, 在 FJSP 问题中采用这些变异算子有可能产生非法的个体 为此, 本文设计了一种新的变异算子: 根据变异概率 P_m 选取需要变异的个体, 并随机选择 i 个需要变异的基因 ($1 \leq i \leq l$); 对于每个被选择变异的基因, 如果该基因所在工序可选择的机器数为两个以上, 则对该道工序从可行的机器数集合中进行重新选择, 作为变异后该变异位的基因, 最终得到变异后代

第 2 部分: 为了体现变异后代的多样性, 采用单点变异的方法, 即对选取的应变异的父体, 随机选取一个变异点, 互换交叉点前后的基因, 构成变异后代

的染色体

3.2.3 复制算子

为将遗传算法更有效地应用于调度问题, 本文采用最优保留的选择策略 设群体规模为 n , 将父代与子代群体同时进行比较, 选择前 n 个个体作为下一代群体 这种选择策略有利于个体优良性质的遗传, 同时在遗传过程中始终保留目前最优的调度解, 将个体的优良特性加以记忆, 减少了重复操作的次数

4 实例仿真与分析

为了验证算法的有效性, 本文采用文献[5]的数据进行计算 该 FJSP 问题中有 4 个工件在 6 台机器上加工, 各工件可供选择的机器及在各机器上的加工时间如表 1 所示

表 1 各工件的可供选择机器及加工时间

工件	工序	可供选择的机器					
		1	2	3	4	5	6
1	O_{11}	2	3	4	—	—	—
	O_{12}	—	3	—	2	4	—
	O_{13}	1	4	5	—	—	—
2	O_{21}	3	—	5	—	2	—
	O_{22}	4	3	—	—	6	—
	O_{23}	—	—	4	—	7	11
3	O_{31}	5	6	—	—	—	—
	O_{32}	—	4	—	3	5	—
	O_{33}	—	—	13	—	9	12
4	O_{41}	9	—	7	9	—	—
	O_{42}	—	6	—	4	—	5
	O_{43}	1	—	3	—	—	3

此时, 总的加工工序为 12 道, 且 $M_o = \{3, 3, 3, 3, 3, 3, 2, 3, 3, 3, 3, 3\}$. 选取群体规模为 40, 以最大遗传代数 200 作为停止准则, 交叉概率 $P_c = 0.7$, 变异概率 $P_m = 0.15$ 经过 10 次仿真实验得到的结果如表 2 所示

表 2 遗传算法对 FJSP 调度问题的仿真结果

最优值	最差值	平均值	达最优值次数	平均收敛代数
17	18	17.5	5	67.3

仿真实验所得的 3 个最优调度方案的染色体如下:

染色体 1:

a 2 2 1 1 1 1 1 2 2 2 1 3
b 3 12 10 1 7 6 5 9 11 2 8 4

染色体 2:

a 2 2 2 3 3 1 1 2 2 2 1 1
b 12 5 2 4 3 10 6 7 1 11 8 9

染色体 3:

a 2 2 2 1 1 1 1 2 2 2 3 3
 b 7 6 11 3 10 12 1 9 2 5 4 8

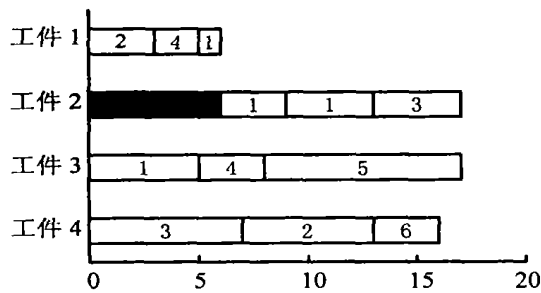


图 1 染色体 1 对应的甘特图

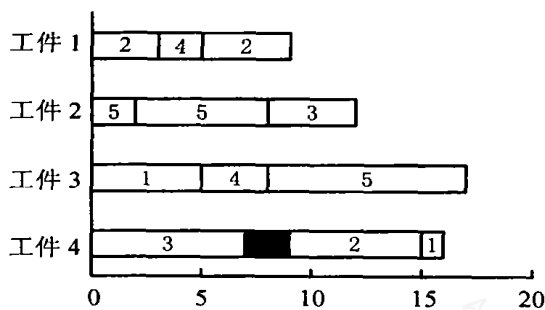


图 2 染色体 2 对应的甘特图

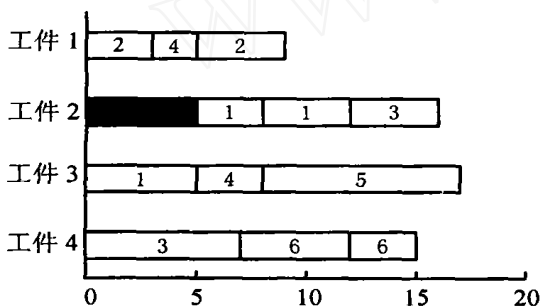


图 3 染色体 3 对应的甘特图

其中 a 和 b 分别表示第 1 部分和第 2 部分的染色体编码 染色体 1, 2, 3 所对应的调度甘特图分别如图 1 ~ 图 3 所示

文献[5]中最优生产周期为 18, 而遗传算法得到的更优调度解为 17, 体现了遗传算法强大的搜索能力. 从最优染色体的编码及甘特图可以看出, 若某一染色体中相邻位置的基因具有较优的性质, 则包含该顺序的其他染色体也有较好的适应值, 并反映出染色体编码与解码的相互对应关系. 这不仅为利用遗传算法解决 FJSP 调度问题提供了一定的理论依据, 而且有助于对分部遗传算子设计方法的深入理解

5 结 论

本文通过分析 FJSP 问题的特点, 提出一种新的遗传调度算法. 该算法利用分部编码方法及分部遗传算子, 有效地保证了遗传后代的可行性, 并能较快地得到最优调度方案, 有利于生产计划与实际生产调度相结合, 提高机器利用率, 缩短生产周期

参考文献 (References):

[1] Takeshi Yamada, Ryohei Nakana. Genetic algorithms for job-shop scheduling problems [A]. *Proc of Modern Heuristic for Decision Support* [C]. London, 1997. 67-81.
 [2] Jose Fernando Goncalves, Jorge Jose de Magalhaes Mendes, Maurício G C Resende. A hybrid genetic algorithm for the job shop scheduling problem [R]. *AT&T Labs Research*, 2002.
 [3] Davis L. *Handbook of Genetic Algorithms* [M]. New York: Van Nostrand Reinhold, 1991.
 [4] 邢文训, 谢金星. 现代优化计算方法 [M]. 北京: 清华大学出版社, 1999. 166-169.
 [5] Nasr N, Elsayed E A. Job shop scheduling with alternative machine [J]. *Int J of Production Research*, 1990, 28(9): 1595-1609.

(上接第 1196 页)

[6] 张连生, 高峰, 姚奕荣. 非线性整数规划的连续化 [J]. *运筹学学报*, 1998, 2(2): 59-66.
 (Zhang Liansheng, Gao Feng, Yao Yirong. Continuity method for nonlinear integer programming [J]. *Operation Research Trans*, 1998, 2(2): 59-66.)

[7] Luenberger D G. *Linear and Nonlinear Programming* [M]. Massachusetts: Addison-Wesley Publishing Company, 1984. 1-134.
 [8] 盛昭瀚. 主从递阶决策论——Stackelberg 问题 [M]. 北京: 科学出版社, 1998. 352-354.