一种基于遗传算法的车间调度算法求解*

安　晶,秦　珂
(盐城工学院 优集学院,江苏 盐城　224003)

摘　要:针对车间作业调度问题,讨论了应用于车间作业调度的遗传算法设计,给出了主要的编码、解码、以及死锁问题的算法模型。结合应用实例,说明了设计的可行性与有效性。

关键词:车间作业调度;遗传算法;编码;解码;死锁
中图分类号:TP391.9　　文献标识码:A　　文章编号:1671-5322(2007)01-0033-04

车间作业调度(Job-Shop Scheduling),简称JSS,是一个典型的NP难问题,也是CIMS领域中研究的重要课题,其研究不仅具有重大的现实意义,而且具有深远的理论意义。长期以来,JSS研究的方法始终以启发式算法为主导,绝大部分的JSS研究工作也都围绕着启发式算法进行,如基于启发式算法的JSS仿真系统,基于启发式算法的并行JSS系统,基于启发式算法的JSS专家系统,等等,尽管这些研究取得了一定的应用成果,但是却存在着难以克服的弱点,如计算规模不可能较大,寻优结果不具备全局特性等等。近年来,又有学者提出了基于神经网络的车间作业调度系统,但此种方法在JSS规模较大时,存在着计算速度慢与结构参数难以确定的弱点。由此可见,要想进一步研究JSS,选择一种有效的方法极为必要。遗传算法的出现给这类问题带来了新的希望,在此,我们给出了一个基于遗传算法的车间作业调度的求解方法。

1　Job-Shop 调度问题的问题描述

车间作业(Shop Job)就是利用车间资源对某一对象进行生产过程,作业调度则是对车间作业进行有效排序,使某个目标函数最小。作业的对象既可以是某一部件,也可以是某一零件。既可以是某一个零件的某一个工序,也可以是某一个工序的某一个工步。由于工序是部件和零件的基本元素,同一工序的工步通常是工序的绝对分割,即在同一台机器上连续完成,所以将车间作业的对象确立为零件的工序。于是,作业调度的对象也就确立为工序。

有 M 台机器及 N 个工件,由于工件的加工工艺的要求,每个工件使用 M 台机器的顺序以及每道工序所花费的时间已经给定。如何安排每台机器上工件的加工顺序,使得某种指标(如总的完成时间)最小,此指标就是作业调度问题的优化目标。

2　遗传算法

遗传算法(Genetic Algorithms,GA)研究的历史比较短,20世纪60年代末期到70年代初期,主要由美国Michigan大学的John Holland与其同事、学生们研究形成了一个较完整的理论和方法,从试图解释自然系统中生物的复杂适应过程入手,模拟生物进化的机制来构造人工系统的模型。

遗传算法采纳了自然进化模型,如选择、交叉、变异、迁移、局域与邻域等。计算开始时,一定数目 N 个个体(父个体1、父个体2、父个体3、父个体4…)即种群随机的初始化,并计算每个个体的适应度函数,第一代也即初始代就产生了。如果不满足优化准则,开始产生新一代的计算。为

* 收稿日期:2006-11-22
基金项目:黑龙江省自然科学基金资助项目(F0207)
作者简介:安　晶(1982-),女,江苏徐州市人,助教,硕士,主要研究方向为CAD、CAM。

了产生下一代,按照适应度选择个体,父代要求基因重组(交叉)而产生子代。所有的子代按一定概率变异。然后子代的适应度又被重新计算,子代被插入到种群中将父代取而代之,构成新的一代(子个体 1、子个体 2、子个体 3、子个体 4…)。这一过程循环执行,直到满足优化准则为止。

3　遗传算法在解 Job – shop 调度问题方面的研究现状

由于 Job – Shop 调度问题是一个 NP 难题,而遗传算法为求 NP 难度问题的近似解提供了一种有效手段,所以现在许多人都致力于用遗传算法解决 Job – shop 问题,各有特点。但就目前来看:

(1)由于 Job – Shop 调度问题的特殊性,编码机制显得尤为重要,因为编码机制选择不当,遗传算法的杂交、变异算子很容易破坏原加工顺序。

(2)死锁问题也是一个重要问题,如果处理不当,死锁的出现是无法预料的。

(3)收敛性及收敛速度问题,应用 GA 解 Job – Shop 调度问题时很少有人考虑这两个问题,所以得到的结果与最佳值的接近程度无理论保证。

4　车间作业调度遗传算法的关键步骤

为处理起来更加直观,我们把 Job – Shop 调度问题的工时工序表用两个矩阵表示 $M(i,j)=[(i,k)]_{n\times m}(i=1,2,\cdots,n;j,k=1,2,\cdots,m)$,工件 J_i 的第 j 道工序在机器 k 上加工;$T(i,j)=[T_{ij}]_{n\times m}(i=1,2,\cdots,m;j=1,2,\cdots,n)$ 表示加工时间矩阵。另外 $N(k,j)=[(i,k)]_{n\times m}(i,j=1,2,\cdots,n;k=1,2,\cdots,m)$ 表示机器 k 上各工件的加工顺序,机器 k 上第 j 个加工的工件是 J_i。例如有一个 6×6 的 M、N 矩阵如下:

$$M(i,j)=\begin{bmatrix}(0,2)&(0,0)&(0,1)&(0,3)&(0,5)&(0,4)\\(1,1)&(1,2)&(1,4)&(1,5)&(1,0)&(1,3)\\(2,2)&(2,3)&(2,5)&(2,0)&(2,1)&(2,4)\\(3,1)&(3,0)&(3,2)&(3,3)&(3,4)&(3,5)\\(4,2)&(4,1)&(4,4)&(4,5)&(4,0)&(4,3)\\(5,1)&(5,3)&(5,5)&(5,0)&(5,4)&(5,2)\end{bmatrix}$$

$$N(k,j)=\begin{bmatrix}(0,0)&(3,0)&(2,0)&(5,0)&(1,0)&(4,0)\\(1,1)&(3,1)&(5,1)&(4,1)&(0,1)&(2,1)\\(0,2)&(2,2)&(1,2)&(4,2)&(3,2)&(5,2)\\(2,3)&(5,3)&(3,3)&(0,3)&(1,3)&(4,3)\\(1,4)&(4,4)&(3,4)&(5,4)&(0,4)&(2,4)\\(2,5)&(5,5)&(1,5)&(0,5)&(4,5)&(3,5)\end{bmatrix}$$

$$T=\begin{bmatrix}1&3&6&7&3&6\\8&5&10&10&10&4\\5&4&8&9&1&7\\5&5&5&3&8&9\\9&3&5&4&3&1\\3&3&9&10&4&1\end{bmatrix}$$

4.1　*Job – Shop* 调度问题的数学模型

设有 m 台机器 M_1、M_2、…、M_m 和 n 个工件 J_1、J_2、…、J_n,工件 J_j 的第 j 道工序的加工时间为 T_{ij}。满足如下条件:

(1)一台机器一次只加工一个工件。

(2)每个工件不能同时在多台机器上加工。

(3)每个工件的加工顺序预先确定,工件按序加工。

(4)每个工件在每台机器上只加工一次。

(5)工件在加工时不允许中断。

设 $C(i,j)(i=1,2,\cdots,n;j=1,2,\cdots,m)$ 表示工件 J_j 的第 j 道工序的加工机器号,$E_{i,j,C(i,j)}$ 表示 $C(i,j)$ 机器上工件 J_j 的第 j 道工序的完工时间,$B_{i,j,C(i,j)}$ 表示 $C(i,j)$ 机器上工件 J_j 的第 j 道工序的开工时间。则有:

$$E_{i,j,C(i,j)}>0$$
$$B_{i,j,C(i,j)}\geqslant 0$$
$$E_{i,j,C(i,j)}=B_{i,j,C(i,j)}+T_{ij}$$
$$B_{i,j,C(i,j)}=\max\{E_{i,j-1,C(i,j-1)},E_{k,l,C(i,j)}\}$$

其中 $E_{k,l,C(i,j)}$ 为机器 $C(i,j)$ 上加工工件 J_j 之前所加工工件 J_k 的第 l 道工序的完工时间。

选择机器的最长完工时间最小为目标函数,则目标函数可具体表示为:

$$\min\{E_{i,j,C(i,j)}\}\qquad c(i,j)=1\cdots m$$

4.2　编码

Job – Shop 调度问题就是在不破坏加工顺序的前提下,怎样在机器上安排工件以达到目标函数式的要求,即各工件在机器上的最佳排列组合,而这种排列组合关系有其内在的序关系,如果假定工件号按由小到大的自然顺序排列,则各台机器上任意两个不同工件之间只有两种序关系(顺序或逆序),这种序关系用二进制表示,顺序为 1,逆序为 0。例如对第 k 台机器上的工件 J_i 和工件 J_j(设 $i<j$),若 J_i 在 J_j 前,则序值为 1 若 J_j 在 J_i 前,则序值为 0。以第 k 台机器上 n 个工件的序关系所对应的序值为基因码,可以得到一个长度为 $n(n-1)/2$ 染色体子串,故 m 台机器的染色体串总长为 $mn(n-1)/2$。

按算法 N 矩阵中的染色体子串分别为：

$a[0]=11111,0010,011,11,0$

$a[1]=01000,1111,000,10,0$

$a[2]=10111,0111,111,01,1$

$a[3]=10010,0010,111,10,0$

$a[4]=00000,1111,001,01,1$

$a[5]=00100,0110,111,00,0$

4.3　解码

编码方法提供了一种将原问题转化为更易于求解的手段，但是二进制的染色体码并不能求出适应函数值，这就需要将二进制转换回十进制。问题空间 Ω 的基数为 $\frac{(m\times n)!}{(m!)^n}$，而二进制编码空间 $\wedge$ 的基数为 $2^{\frac{mn(n-1)}{2}}$，有 $|\Omega|<|\wedge|$，所以解码方法不能再是编码方法的简单逆过程。为保证每个二进制编码对应一个解，需建立一个解码规则，使之是一个满射关系，其基本思想为：将染色体划分成 m 个长度为 $n(n-1)/2$ 的子串，再对每个子串依次截取长度为 $n-1, n-2, \cdots, 2, 1$ 的小串，统计第 i 个小串中 0 的个数 r，则 J_i 应排在第 $r+1$ 个空白处（未被占用的位置）。

例：设某台机器上的染色体子串为 10010,0010,111,10,0，求该台机器上的加工顺序。

解　根据算法 3.2.2 知，工件 i 应在第 $r(i)+1$ 个空位，

$i=0$ 时，10010 中 0 的个数 $r=3$，所以工件 0 在第 4 位；

$i=1$ 时，0010 中 0 的个数 $r=3$，所以工件 1 在第 $4+1=5$ 位；（因为工件 0 已经占据了 1 位）

$i=2$ 时，111 中 0 的个数 $r=0$，所以工件 2 在第 1 位；

$i=3$ 时，10 中 0 的个数 $r=1$，所以工件 3 在第 $2+1=3$ 位；

$i=4$ 时，0 中 0 的个数 $r=1$，所以工件 4 在第 $2+4=6$ 位；（因为在第 2 个空位之后已有 4 个已被占有）

剩下的一位应给工件 5。

所以，该台机器上的加工顺序为：2,5,3,0,1,4

4.4　约束问题 – 死锁的判断

本文中，由 M、N 两矩阵可以判断死锁是否发生。两个指针数组 pointern(n)、pointerm(m) 分别指向 M、N 矩阵中第 1 列各元素，pointern(i) $(i=0,1,\cdots,n-1)$ 指向的是当前要加工的各工件及其所在的机器，pointerm(j) $(j=0,1,\cdots,m-1)$ 指向的是当前各台机器可以加工的工件。如果 pointern(i) 指向的所有元素和 pointerm(j) 指向的所有元素有相同的 (i,j)，则表明该工件的下一道工序可以在机器 k 上加工，机器 k 上下一个可以加工的工件是 J_i，加工可以继续下去，否则就会产生死锁。加工完 (i,j) 后，pointern$(i)\leftarrow$pointern$(i)+1$，pointerm$(j)\leftarrow$ pointerm$(j)+1$，直到所有元素都判断完毕。

1）$k\leftarrow1$，pointerm$(i)\leftarrow0$，pointern$(j)\leftarrow0$

2）$k>n\times m$ 成立吗？若是则无死锁并转 5），否则转 3）

3）查询 pointerm(i) 指向的所有元素和 pointern(j) 指向的所有元素是否有相同的 (i,j)，若这样的元素不存在则死锁并结束，否则转 4）

4）在机器 j 上加工工件，根据式（3）、（4）求各台机器上当前的最早完工时间，$k\leftarrow k+1$，pointerm$(i)\leftarrow$pointern$(i)+1$，pointern$(j)\leftarrow$pointern$(j)+1$，转 2）

5）由式（5）求目标函数值，结束

5　二进制的病毒遗传算法解 Job – Shop 调度问题的测试用例

我们给出二组工时工序表和加工时间矩阵，每组按参数不同又分为 2 组不同的例子，选取群体规模为 100，遗传代数为 500。其中，m 为工时工序表矩阵，t 为加工时间矩阵，为方便起见，采用行输入法。以一组为例：

$m=(231045\ 325410\ 214503\ 425130\ 235104\ 120354)$

$t=(236541\ 589565\ 478595\ 632545\ 563254\ 478596)$

测试用例：$(pc=0.75, pm=0.15)$

相应的甘特图和迭代曲线如下图：

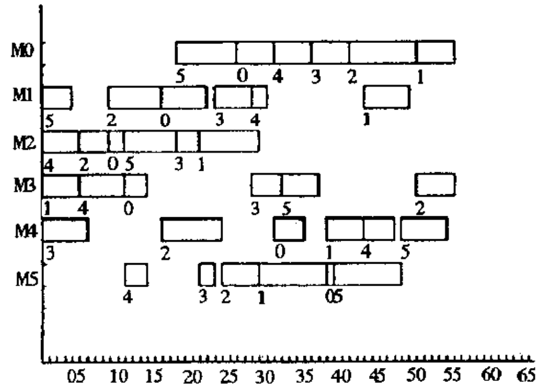

图 1　甘特图

Fig. 1　Gantt Chart

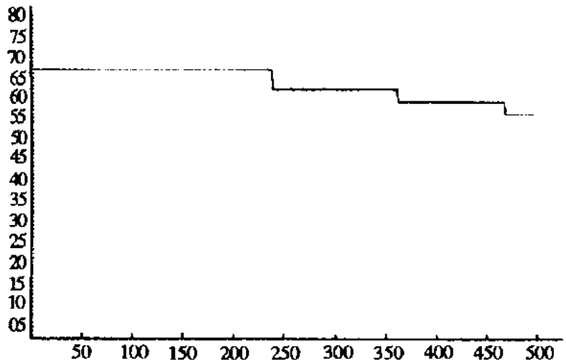

图 2 迭代曲线
Fig. 2 Iterative curve

由上面例子及我们的多组实验结果表明,对同一组测试矩阵,变异概率取得适当大一些效果能更好一点。从所举的例子看,本算法所得的最短加工时间都是比较优的结果,迭代效果也是很明显的,但由于此算法对收敛性没做什么保证,从迭代曲线上看收敛速度不是很快,在这方面仍需改进。但由此也充分说明了 JSS 遗传算法是解决 JSS。

问题的有效方法,说明了其设计的可行性与成功性。

参考文献:

[1] Guoyong Shi, A genetic algorithm applied to a classic job – shop scheduling problem[J]. International Journal of System Science,1997, 28(1):25 – 32.
[2] 王小平,曹立明. 遗传算法[M]. 西安:西安交通大学出版社,2002.
[3] 李小平. 用定解最优保存遗传算法解 Job – Shop 调度问题的研究[D],哈尔滨:哈尔滨理工大学,1999.
[4] 方剑,席裕庚. 基于遗传算法的 job – shop 静态调度算法[J]. 上海交通大学学报,1997(3):49 – 52.

A Solution of Job Shop Scheduling System Based on Genetic Algorithms

AN Jing, QIN Ke
(UGS College, Yancheng Institute of Technology, Jiangsu Yancheng 224003,China)

Abstract:The paperhas discussed a design of job shop scheduling oriented genetic algorithm to solve job – shop scheduling problems;The models presented are mainly coding, decoding, and dead lock. Finally, the conclusion drawn by a test shows that the design is applicable and successful.

Keywords:job shop scheduling;genetic algorithm;coding;decoding;dead lock