

解并行机提前/拖后调度问题的一种新并行遗传算法

何桂霞¹, 于明远²

(1. 浙江工业大学 之江学院, 浙江 杭州 310024; 2. 浙江工业大学 信息学院, 浙江 杭州 310032)

摘要: 针对交货期窗口非等同并行机提前/拖后调度问题, 设计了一个基于向量组编码的新的遗传算法. 此算法特点是编码方法简单, 能有效地反映实际调度方案, 即清楚反映出每台机器加工工件的代号和顺序, 并能保证交叉和变异后个体自动满足约束条件, 收敛速度快. 同时为了更好地适应调度实时性和解大型此类问题的需要, 在基于遗传算法自然并行性特点的基础上, 实现了主从式控制网络模式下并行遗传算法. 仿真结果表明, 此算法是有效的, 优于普通的遗传算法, 具有较高的并行性, 并能适用于解大型这类调度问题.

关键词: 提前/拖后; 并行遗传算法; 交货期窗口; 调度问题

中图分类号: TP301.6

文献标识码: A

文章编号: 1006-4303(2008)01-0062-05

A new parallel genetic algorithm for parallel multi-computers in advance/tardiness scheduling problem

HE Guixia¹, YU Mingyuan²

(1. Zhijiang College, Zhejiang University of Technology, Hangzhou 310024, China;
2. college of Information Engineering, Zhejiang University of Technology, Hangzhou 310024, China)

Abstract: In order to solve parallel multi-computers in advance/tardiness scheduling problem with due windows, a new genetic algorithm based on a vector group coding method is proposed. The genetic algorithm shows the following characteristics: its coding method is simple and can effectively reflect the virtual scheduling policy, which can vividly reflect the numbers and sequences of these produced jobs from every machine, and ensure the individuals generated by crossover and mutation to automatically meet the constraint conditions. Meanwhile, based on the natural parallelism, parallel hybrid genetic algorithm is applied under the mode of master-slave control networks in order to adapt to this kind of larger scale and real-time scheduling problems. The computational results show that it is much effective, higher parallelism and better than ordinary genetic algorithms. The method can be applied to solve larger scale in advance/tardiness scheduling problems with due windows.

Key words: in advance/tardiness; parallel genetic algorithm; due date windows; scheduling problem

0 引言

并行机提前/拖后调度问题是生产调度中的一

个重要问题. 近些年来, 由于 JIT (Just in time) 思想的引入, 人们将注意力由从追求以机器利用率为目标并行多机调度问题转到基于准时制的并行多机调度问题上. 在实际应用中发现交货期窗口的调度

收稿日期: 2007-05-11

基金项目: 浙江省教育厅基金资助项目 (Y200702346)

作者简介: 何桂霞 (1974—), 女, 河南潢川人, 讲师, 硕士, 研究方向为企业信息化管理及工业工程.

问题比点交货期调度问题更加贴近实际,因此对交货期窗口调度问题的研究成为人们研究的热点问题.交货期窗口的并行机调度问题,已证明是NP难题,对于此类问题,早期的研究主要是一些启发式算法^[1,2],然而随着问题规模的扩大,求的解质量往往不令人满意.

遗传算法是一种随机搜索算法,以其近优、快速及易实现的特点在组合优化领域获得了成功的应用,在解决工业生产调度问题中,也显示出了很大的优越性,并且在解此类问题时,也取得了一定成果^[3,4].对此类问题的遗传算法,编码是关键,不好的编码不仅不能很好地反映调度方案,即使确定了每台机器上加工的工件代号及其加工顺序,也会在交叉或变异时产生大量的非法解,这样需额外对染色体进行可行性检验,降低了运行速度和寻优效果.为此,人们在研究解此类问题的遗传算法时,对其编码方法进行了专门的设计,诸如文献[5]提出了分段编码方法,文献[6,7]提出了扩展排列的编码方法等等.纵观这些编码设计方法,虽然都能够很好地反映调度方案,但存在两个明显的缺点:第一编码及其对应的解码过程复杂;第二如采取简单的交叉或变异方法将产生大量的非法解,为此,需要特殊的交叉或变异方法.此两个过程大大降低了遗传算法的运行速度和寻优效果.

针对上面遗传算法在编码方面的这些缺点,这里提出了一种基于向量组编码新的遗传算法.编码不仅简单,能很好地反映调度方案,而且交叉和变异简单,不会产生非法解.同时为满足调度实时性和大规模调度问题的需要,在基于遗传算法自然并行性特点的基础上,实现了主从式控制网络模式下并行遗传算法.仿真实验和实际应用数据实验表明此算法是有效的,具有较高的并行性.

1 问题的描述

交货期窗口并行机提前/拖后调度问题描述:有 n 个相互独立的工件, m 台机器设备($m \geq 2$,且各台机器不一定有相同的加工周期),每个工件只有一道工序,且可由此 m 台机器中的任意一台完成加工任务,每个工件的交货期窗口不同但确定,提前和拖后都受惩罚,在窗口内完成不受惩罚.要找一最优调度方案,即确定每台机器上加工的工件代号及其加工顺序,使加工完所有工件后提前/拖后惩罚的总

和最小.这里假设每机器每时刻只能加工一个工件,且一个工件一旦确定被某机器加工就不能中断直至加工完成.

假设 $N = \{1, 2, \dots, n\}$ 为工件集合, $M = \{1, 2, \dots, m\}$ 为机器设备的集合, t_{ij} 表示工件 i 由机器 j 加工所用的加工时间,其中 $i \in N, j \in M$,具体是如下的矩阵:

$$(t_{ij}) = \begin{bmatrix} t_{11} & \dots & t_{1m} \\ \vdots & & \vdots \\ t_{n1} & \dots & t_{nm} \end{bmatrix}$$

c_i 是工件 i 的完工时间, $[d_i, e_i]$ 表示工件 i 的交货期窗口,其中 d_i, e_i 分别表示是工件 i 的最早和最晚交货期.如果工件 i 在其交货期窗口内完工,则不受惩罚,否则对工件 i 进行提前/拖后惩罚,问题的目标就是找到一个调度使下面的式子最小化.

$$\min F(\theta) = \min \left\{ \sum_{i=1}^n h_i \text{esign}(i) + \sum_{i=1}^n w_i \text{tsign}(i) \right\} \quad (1)$$

式中: θ 为一个调度方案; h_i, w_i 分别是工件 i 的单位提前/拖后惩罚数.

$$\text{esign}(i) = \begin{cases} 1, & d_i > c_i \\ 0, & d_i \leq c_i \end{cases} \quad \text{tsign}(i) = \begin{cases} 1, & e_i < c_i \\ 0, & e_i \geq c_i \end{cases}$$

2 求解问题的串行遗传算法

根据问题的特征,笔者采用了基于向量组的编码方法,保证了编码方式能够很好地反映调度方案,同时对其初始种群、复制、交叉和变异方法也进行了研究.设 $P(t)$ 和 $O(t)$ 分别表示在第 t 代中的父代和子代,则算法总的流程如下:

Begin

$t \leftarrow 0$

产生初始种群 $P(t)$

评价 $P(t)$

Repeat

以交叉概率随机地选择两个个体进行交叉操作,得到的子代加入到子代群体 $O(t)$ 中

以变异概率随机地选择一个个体进行变异操作,得到的子代加入到子代群体 $O(t)$ 中

评价 $P(t)$ 和 $O(t)$

从 $P(t)$ 和 $O(t)$ 中选择 $P(t+1)$

$t \leftarrow t+1$

Until{终止条件满足/}

End

2.1 编码设计

根据问题的特征,提出了基于向量组的编码方法.对每个工件 $k_i(i \in N)$,如加工的机器为 e_i ,则定义基因为二维向量 $g_i = \begin{bmatrix} k_i \\ e_i \end{bmatrix}$,于是染色体的基因编码为 $[g^1, g^2, \dots, g^n] = \begin{bmatrix} k_1 & k_2 & \dots & k_n \\ e_1 & e_2 & \dots & e_n \end{bmatrix}$,其长度等于待加工工件的总数 n .

如何对编码进行译码,用下面过程来说明:

(1) 令 $i = 1$.

(2) 读取基因 $g_i = \begin{bmatrix} k_i \\ e_i \end{bmatrix}$.

(3) 将工件 k_i 分配到机器 e_i 上.

(4) $i = i + 1$.

(5) 如果 $i \leq n$,则转向(2).

(6) 分配到同一个机器上的工件根据其在染色体中的排列位置依次加工,计算每个工件 i 的完工时间 c_i ,从而依据式(1)计算出目标函数值.

例1:假设 $n = 7, m = 3$,则染色体的译码过程如图1所示.

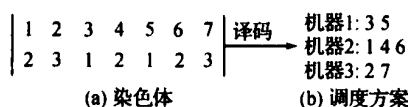


图1 染色体的译码过程

Fig. 1 Decoding procedure of chromosome

2.2 种群的初始化

对于遗传算法来说,初始种群的大小和优劣对算法的执行效果有明显的影响.为保证群体的多样性,在初始化种群时,这里采取如下的初始化方法.

随机产生 Pop_Size (初始种群尺寸)个 n 维向量 (k_1, k_2, \dots, k_n) 分别作为初始种群中 Pop_Size 个染色体的第一行,其中 $k_i(i \in N)$ 为互不相同的自然数.对每一个向量 (k_1, k_2, \dots, k_n) 中的元素 $k_i(i \in N)$,从加工机器集合中随机选择一个机器代码作为其加工机器,不妨记作为 e_i ,按此办法就生成每个染色体的第二行向量 (e^1, e^2, \dots, e^n) .

2.3 适应值的计算

定义适应值 $\text{Eval}(\theta) = \alpha \cdot e^{-\beta F(\theta)}$

式中: α 和 β 为大于零的常数.

2.4 遗传算子

2.4.1 复制

在遗传算法中,按照适应值在亲代种群中选择优良个体的常用选择算法,有轮盘赌选择法、局部选

择法、随机遍历抽样法、截断选择法等.由于在仿真实验中发现种群内的较多个体的适应度值彼此非常接近,因此,这里使用排序选择机制来获得个体被选中的概率,即首先将种群(包括父代和子代)内的个体按适应值大小排序,其中适应值最大的排第一位;然后对所有排序号以某种规则选取,使排位越靠前的个体被选中的机会越大.笔者采用线性选择机制,被选中个体的序号定义为

$$\text{index} = S \cdot (\delta - \sqrt{\delta^2 - 4(\delta - 1) \cdot \text{random}()}) / 2 / (\delta - 1)$$

式中: $\delta = 2$ 为偏置量; S 为种群的规模.适应度最大的个体被选中的概率大致等于正中间个体的两倍.

2.4.2 交叉

该操作目的是产生新的个体,从而增加搜索到更优个体的机会.在交叉时采用简单的单点交叉和两点交叉即可,不会产生非法的染色体.

例2:下面举例来说明整个交叉过程.假设两个父代个体 A 和 B 分别为

$$\begin{bmatrix} 1 & 3 & 2 & 4 & 7 & 5 & 6 \\ 2 & 1 & 3 & 2 & 1 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 1 & 2 & 1 & 2 & 3 \end{bmatrix}$$

则交叉后产生的个体为

$$\begin{bmatrix} 1 & 3 & 3 & 4 & 5 & 5 & 6 \\ 2 & 1 & 1 & 2 & 1 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 2 & 2 & 4 & 7 & 6 & 7 \\ 2 & 2 & 3 & 2 & 1 & 2 & 3 \end{bmatrix}$$

2.4.3 变异

启发式变异:为使变异的染色体有较好的特征,这里采用了启发式插入变异的方式,即随机地选择要变异的基因 g_i ,采用邻接搜索的方法,将集合中所有机器号(但不包括要变异的基因号)都插入到此一次,覆盖原有的基因,这样产生 $m - 1$ 个染色体,选择其中最好的作为变异产生的后代.

很容易看出,此种变异方式不仅改变了工件的加工顺序,而且也能较好地保证了种群的多样性、变异后个体质量.

3 求解问题的并行遗传算法

并行遗传算法是基于主从式控制网络,具体方法如下:将其中一个子群体设置成中心,其他子群体均与中心子群通讯.中心子群体中始终保存当前最优个体,其它子群体通过“引进”中心子群体中的最优个体来加速收敛速度.

下面根据主从式控制网络并行遗传算法定义,

解交货期窗口下非等同并行机调度提前/拖后问题的并行遗传算法实现如下:

- Step1: 产生一个进程(该进程为父进程).
- Step2: 由父进程产生 $m-1$ 个子进程.
- Step3: 各子进程(包括父进程)进行串行遗传算法, 当子进程中遗传代数被 10 整除, 子进程发送最优个体至父进程.
- Step4: 父进程选择当前各子进程中最优个体(记为 good), 发送给各子进程.
- Step5: 各子进程用 good 替换自己当前代种群中适应值最低的个体.
- Step6: 如果繁殖代数大于预先设定的最大繁殖代数, 则结束, 否则转向 Step 3.

4 数值实验

在 IBM PIV 2.8 处理器, Linux9.0 操作系统, MPI 环境下的 Cluster 4 个节点机群上对上述方法(记为 PGA)进行了数值实验.

4.1 仿真实验

4.1.1 初始设置

- 实验的初始设置如下:
- (1) 取六组不同规模(表示为 $n \times m$)的问题, 即 30×6 ; 50×4 ; 60×5 ; 80×10 ; 80×20 ; 100×20 .
 - (2) 取种群规模 50, 交叉率为 0.8, 变异率为 0.25, 最大遗传代数为 50, 每个工件的提前和拖后的惩罚数分别为区间 $[0.1, 1)$ 和 $[1, 1.5)$ 内的随机数.
 - (3) 每个机器加工工件的时间随机生成, 但生成数要满足均匀分布 $U[1, m]$, 每个工件 $i(i \in N)$ 的最早交货时间和最晚交货时间也是随机生成, 但要分别满足均匀分布 $U(t_{\min}, n \times t_{\min}/m)$ 和 $U(t_{\min} \times n/m, (t_{\max} - (t_{\max} - t_{\min})/2) \times n/m)$.

4.1.2 PGA 与 GA 的比较

为验证笔者提出的算法有效性, 将该算法与文献[5]中的算法进行比较. 对每种规模的问题, 表 1 中最优计算值(提前/拖后调度惩罚最优值)及平均值(最优计算值所在代的所有个体提前/拖后调度惩罚

值的平均值)数据为运行 100 代(这里把初始设置的 50 变更为 100)的结果.

表 1 方法比较

Table 1 Comparing GA with PGA

序号	规模 $n \times m$	遗传算法		PGA	
		最优计算值	平均值	最优计算值	平均值
1	30×6	1.67	2.37	1.64	2.28
2	50×4	6.03	7.32	5.52	6.97
3	60×5	33.45	34.35	31.02	31.63
4	80×10	43.23	46.21	39.76	40.26
5	80×20	99.32	101.21	93.06	96.73
6	100×30	100.14	102.76	93.85	94.96

从表 1 中可以看出, 对于不同的规模, PGA 计算的最优计算值和平均值都优于文章^[5]设计的遗传算法, 并且随着规模的扩大, PGA 的优势更明显, 这表明 PGA 是有效的.

4.1.3 PGA 的并行效率

以两个处理器为例, 在表 2 里列出在不同规模下, 算法结束后各个处理器的通讯时间及总计算时间.

表 2 通讯时间比较

Table 2 Comparison with communication time

序号	规模	第一处理器/s	第二处理器/s	总计算时间/s
1	30×6	0.000 583 0	0.000 548 9	1.409 485 1
2	50×4	0.000 630 8	0.000 689 3	1.603 238 2
3	60×5	0.000 632 3	0.000 675 4	1.629 445 3
4	80×10	0.000 874 3	0.000 879 3	1.890 384 1
5	80×20	0.000 895 6	0.000 887 4	2.293 009 7
6	100×30	0.000 946 5	0.000 950 4	2.390 322 8

从表 2 可以看出, PGA 有着较少的通讯时间, 并且受规模的影响较小. 也就是说, PGA 具有较高的并行性.

4.2 实际应用

这小节里, 以浙江余杭一家纺企业为例, 来验证此算法在实际应用中的可行性和有效性. 假设在计划期要用 4 台机器生产 15 种产品, 每台机器生产每种产品所需的时间见表 3, 其中横排为产品种类, 纵排为机器号. 每种产品交货期窗口见表 4, 其中横排为产品种类, 纵排为每种产品对应最早和最晚交货期的值.

表 3 机器及对应产品生产时间表

Table 3 Production time of products on every machine

序号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	11.2	12.9	7	8.02	8.25	14.2	12.8	14.2	13.9	14.7	14.6	12.2	9.57	14.2	13.6
2	12.6	10.9	9.8	8.27	10.7	14.3	9.56	11.5	9.29	10.9	14.3	10.7	10.3	13.2	11
3	12.8	10.7	11.6	7.35	7.59	9.01	8.42	10.4	9.01	14.6	9.29	14	14.5	9.92	10.5
4	14.1	8.79	10.7	7.17	12.9	13.5	12	14.3	9.69	14.5	12.4	9.22	13.3	14.9	8.11

表 4 产品交货期窗口
Table 4 Due time windows of every product

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
最早	8.65	17.9	15.3	19.8	16.2	14.2	18.9	17.3	13.3	10.5	12.7	16.5	19.1	15	19.8
最晚	31.2	31.2	29.5	23.6	31.8	22.5	22.5	27.9	28.7	25.2	31.6	21.3	21.1	31.6	23.6

这里初始种群规模为 50, 交叉概率为 0.80, 变异概率为 0.25, 对每个产品, 单位提前惩罚值都为 0.25, 拖后惩罚值都为 1.25, 则利用 PGA 运算后得到的提前/拖后调度惩罚最优值为 42.165, 最优个体为

$$\begin{bmatrix} 7 & 6 & 1 & 13 & 3 & 5 & 12 & 10 & 8 & 4 & 11 & 9 & 14 & 15 & 2 \\ 3 & 3 & 1 & 1 & 1 & 1 & 4 & 2 & 2 & 4 & 3 & 2 & 3 & 4 & 4 \end{bmatrix}$$

从结果可以看出, 所有的产品几乎都是在最少生产时间的机器上生产, 也就是说, 机器都是在最大能力生产, 因此 PGA 得到的最优调度方案比较合理, 满足企业的实际调度需求.

5 结束语

针对交货期窗口下并行机提前/拖后调度问题, 提出了一个基于向量组编码的新的并行混合遗传算法. 从实验结果可以看出, 笔者提出的算法具有如下明显的优点:

- (1) 编码简单.
- (2) 算法运行速度快, 复杂度仅为 $O(n)$, 优于其它一般的遗传算法.
- (3) 算法具有较高的并行性.

实际应用结果显示该算法能很好地解决交货期窗口下并行机提前/拖后调度问题. 为更好地应用此遗传算法, 下一步工作将研究如何推广应用到带特殊工艺约束条件下成本最小、提前/拖后惩罚最小等单目标或多目标调度问题中.

参考文献:

- [1] RENATA M, VINICIUS A, ARMENTANO. A heuristic for single machine scheduling with early and tardy costs [J]. Europe Journal of Operation Research, 2001, 128(1): 129-146.
- [2] SCOTT J, JOHN W. A modified shifting bottleneck heuristic for minimizing total weighted tardiness in complex job shops [J]. Journal of Scheduling, 2001, 5(3): 247-262.
- [3] 刘民, 吴澄. 用遗传算法解决并行多机调度问题[J]. 系统工程理论与实践, 1998, 19(1): 14-17.
- [4] SUN Hongyi, GUO Qing. Parallel machines earliness and tardiness scheduling with proportional weights[J]. Computers & Operation Research, 2003, 30(5): 801-808.
- [5] 刘民, 吴澄. 解决并行多机提前/拖后调度问题的混合遗传算法方法[J]. 自动化学报, 2000, 25(2): 258-262.
- [6] 王莉, 李大卫, 王梦光. 交货期窗口下的并行机调度问题的遗传算法[J]. 系统工程学报, 2002, 17(1): 45-49.
- [7] 高家全, 王雨顺. 解并行多机提前/拖后调度问题的并行遗传算法[J]. 计算机工程与应用, 2006, 42(20): 10-12.

(责任编辑: 陈石平)