

文章编号:1001-9081(2006)04-0857-04

基于遗传和禁忌搜索算法求解车间调度优化问题

梁迪^{1,2}, 谢里阳¹, 隋天中¹, 陶泽¹

(1. 东北大学 机械工程与自动化学院, 辽宁 沈阳 110004; 2. 沈阳大学 机械工程学院, 辽宁 沈阳 110004)

(ldcai@sohu.com)

摘要:针对柔性生产环境下的车间调度问题,在考虑遗传算法早熟收敛问题和禁忌搜索法自适应优点的基础上,将遗传算法和禁忌搜索法结合起来,提出了基于遗传和禁忌搜索的混合动态优化调度算法,并用实例对该算法进行了仿真研究。结果表明,此算法有很好收敛精度,是可行的,并且能够在扰动发生后提供新的调度计划,与传统的调度算法相比较,体现了明显的优越性。

关键词:遗传算法;禁忌搜索算法;车间调度;组合优化

中图分类号: TP311 **文献标识码:** A

Research on shop scheduling optimization based on genetic and tabu search hybrid algorithm

LIANG Di^{1,2}, XIE Li-yang¹, SUI Tian-zhong¹, TAO Ze¹

(1. School of Mechanical Engineering and Automation, Northeastern University, Shenyang Liaoning 110004, China;

2. School of Mechanical Engineering, Shenyang University, Shenyang Liaoning 110044, China)

Abstract: In order to avoid premature convergence and to balance the exploration and exploitation abilities of simple GA, a hybrid algorithm was proposed to solve dynamic scheduling problem in flexible production environment. It is using GA excellent whole search ability and self-adaptive merit of Tabu Search(TS), and the convergence of research was improved. It is capable of generating alternative schedule after uncertain disturbance takes place on a job shop. After using crossover and mutation operations, a best or second best scheduling plan can be found. The result of the test shows that the method is feasible and efficient.

Key words: genetic algorithm; tabu search; job shop scheduling; combination optimization

0 引言

生产调度是计算机集成制造系统(Computer Integrated Manufacturing System, CIMS)研究领域生产管理的核心内容和关键技术,车间调度问题(Job shop Scheduling Problem, JSP)是最困难的约束组合优化问题和典型的 NP 难问题,其特点是没有一个有效的算法能在多项式时间内求出其最优解^[1,2]。

到目前为止,已有一些传统的启发式算法求解 JSP 优化调度问题^[3,4]。在文献[3]中,作者先是用 Petri 网对 JSP 进行建模,然后应用 L1 算法搜索最优解,L1 算法其本质是分枝定界法和动态规划法的混合运用。文献[4]是用遗传代码先确定各 Job 类型的调度先后顺序,然后用传统的人工智能中的 FBS(Filtered Beam Search)搜索方法对各个 Job 的每道工序按顺序进行安排调度。这种 FBS 搜索方法是宽度优先搜索法的一种改进方法。

上述两种方法都是基于人工智能中的搜索方法进行求解的,其主要缺点是运行时间长,且需要很大的存储空间,搜索到的解也不一定是最优解或次优解。为此,本文利用遗传算法良好的全局搜索能力和禁忌搜索具有记忆能力的全局逐步优化特性,将两种算法进行混合,针对车间流控制系统中的调

度问题进行研究,这样有利于丰富优化过程中的搜索行为,增强全局和局部意义下的搜索能力和效率,与单纯的遗传算法和禁忌搜索算法相比有明显的优越性。

1 问题描述

2.1 JSP 的数学模型建立

JSP 研究 n 个工件在 m 台机器上加工,已知各操作的加工时间和各工件在各机器上的加工次序约束,要求确定与工艺约束条件相容的各机器上所有工件的加工开始时间或完成时间或加工次序,使加工性能指标达到最优。

1) 机器顺序阵 $J_M, J_M(i, j)$ 表示加工 i 工件的第 j 个操作的机器号, $J_M(i, \cdot)$ 表示 i 工件的所有操作按优先顺序加工的各机器号的排列。

2) 加工时间阵 $T, T(i, j)$ 为 j 工件在机器上的加工时间。

3) 工件排列阵 $M_j, M_j(i, j)$ 为 i 机器上第 j 次加工的工件号, $M_j(i, \cdot)$ 表示 i 机器上依次加工的各工件的排列。

机器顺序阵即为技术约束矩阵,与加工时间阵一样,也是事先已知的。工件排列阵则是调度的一种表示形式,可用于构造 Gantt 图。

s. t.

收稿日期:2005-10-12;修订日期:2005-12-26 基金项目:国家 863 计划资助项目(2001AA412020)

作者简介:梁迪(1971-),女,湖南长沙人,讲师,博士研究生,主要研究方向:工作流建模与仿真、生产计划与调度的智能优化方法;谢里阳(1962-),男,安徽岳西人,教授,博士生导师,主要研究方向:先进生产方式、可靠性分析;隋天中(1957-),男,山东牟平人,副教授,硕士,主要研究方向:逆向工程、CAD;陶泽(1977-),女,辽宁沈阳人,博士研究生,主要研究方向:工作流建模与仿真、Petri 网及性能分析。

$$C_{ij} - T_{ij} \geq C_{ik}, i = 1, 2, \dots, n, k = 1, 2, \dots, m$$

(1) 机器 k 先于机器 j 加工工件 i

$$C_{hj} - C_{ij} \geq T_{hj}, i, h = 1, 2, \dots, n, j = 1, 2, \dots, m$$

(2) 工件 i 先于工件 h 在机器 j 上加工

$$C_{ij} \geq 0, i = 1, 2, \dots, n, k = 1, 2, \dots, m \quad (3)$$

将工件完工时间最小化作为目标,其目标函数(即 Makespan)为:

$$\min \max_{1 \leq j \leq m} \{ \max_{1 \leq i \leq n} C_{ij} \} \quad (4)$$

式(1)~(3)表示工艺约束条件决定的各工件的各操作先后加工顺序以及加工各个工件的各机器先后顺序。工件和机器满足:一台机器在某时刻只能同时加工一个工件的一道工序;不同工件的工序之间没有先后的约束,相同工件工序满足先后约束关系。

2 基于遗传和禁忌搜索的优化调度算法

2.1 遗传算法

遗传算法(GA)^[5-9]的提出,在一定程度上解决了传统的基于符号处理机制的人工智能方法在知识表示、信息处理和解决组合爆炸等方面所遇到的困难,其自组织、自适应、自学习和群体进化能力使其适应于大规模复杂优化问题。GA是基于“适者生存”的一种高度并行、随机和自适应优化算法,它将问题的求解表示成“染色体”的适者生存过程,通过“染色体”群的一代代不断进化,包括复制、交叉和变异等操作,最终收敛到“最适应环境”的个体,从而求得问题的最优解或满意解。

2.2 禁忌搜索算法

禁忌搜索(Tabu Search, TS)^[10-12]从广义上讲,是一种通过使用自适应的记忆结构来引导局域搜索的技术,对变动的排序在其可行解的所有空间进行搜寻。通过设置禁忌表,避免陷入局部最优或重复过去的搜索,利用中、长期的存储机制进行强化的多样化搜索,对 JSP 问题 Laguna 等提出三个基于简单移动的 TS 排序策略。

2.3 GATS 混合算法

禁忌搜索算法对于解混合最优问题 COP(Combinatorial Optimization Problems)是非常有效的,它在邻域中重复地搜索准则,快速而高效率地向好的方向移动。但它存在一个问题,即在算法中必需调整不同的参数,从这点看禁忌搜索没有很好的鲁棒性,因为参数的选取对最后得到的解有着直接的影响。由于遗传算法只需调整种群的几个参数而不是单个的解,因而遗传算法是禁忌搜索方法的一个补充。图1所示是 GATS 算法中遗传算法和禁忌搜索法的混合结构。具体地讲,就是将遗传算法中的交叉和变异操作产生的新个体(多个),看作是禁忌搜索法中当前解 X_n 的邻域 $V(X_n)$,然后,搜索 $V(X_n)$ 中的每一个个体。



图1 GATS算法中的GA和TS的混合结构

本文的 GATS 算法可以从邻域 $V(X_n)$ 中选择多个解,只要选择出的这些解都能通过 GATS 算法的禁忌结构的筛选即可,这正体现了遗传算法是基于群体而不是个体进行演变的

这一重要特征。图1中,若将TS模块去掉,则整个结构即为典型的遗传算法结构。GA模块的作用相当于构造了禁忌搜索法的邻域结构,以及为其提供了较好的初始解。由图1可看出,这种混合结构由于融入了禁忌搜索法的思想,使得那些只有通过禁忌检验的个体,才能真正地被作为新的个体所接收,这一方面使得那些有效基因缺失、但适宜度不高于其父代的个体被禁忌掉,另一方面,也使得那些包含有效基因、但适宜度较低的个体有更多的机会参加交叉和变异,从而延缓或避免了早熟收敛的发生,也提高了遗传算法的爬山能力。

3 GATS 混合算法的设计

本文的 GATS 算法是先用 GA 进行全局搜索,使群体中的个体比较稳定地分布在解空间的大部分区域,再从群体中每个个体开始,用 TS 算法进行局部搜索,改善群体的质量,这样可减少调用 TS 算法的次数,也就减少了计算时间。混合策略有效结合了 GA 并行的大范围搜索能力和 TS 的局部搜索能力,力图在算法的全局收敛性能和避免局部极小方面有较大改善。该算法的最大优点是满足收敛条件,这是其他混合算法无法比拟的优势。下面给出混合策略的计算过程:

步骤1:给定初始参数,包括最大迭代次数 T ,群体规模 m ,交换概率 p_c 和变异概率 p_m 。

步骤2:确定编码方式,令 $t = 0$ 。

步骤3:随机产生初始群体,其中有 m 个个体 x_1^1, \dots, x_m^1 。

步骤4:计算群体中每个个体的适应值 $f(x_1^1), \dots, f(x_m^1)$ 。

步骤5:从群体中选择 m 个个体作为下一代的父本点,每个个体被选择的概率为 $P_i = \frac{f(x_i^t)}{\sum_{j=1}^m f(x_j^t)}$, $i = 1, 2, \dots, m$ 。

选择过程中使用保优原则,即上一代最优的个体以概率 1 保存至下一代。

步骤6:根据交换概率,对群体中的个体进行交换运算。

步骤7:根据变异概率,对群体中的个体进行变异运算,产生新一代群体。

步骤8:调用TS搜索过程,对新一代群体中的每一个个体进行局部搜索,改进群体点的质量,设改进后的群体点为 $x_{t+1}^1, \dots, x_{t+1}^m$ 。

步骤9:如果 $t < T$,令 $t = t + 1$,转步骤4;否则转步骤10。

步骤10:停止运算, $x_{t+1}^1, \dots, x_{t+1}^m$ 中目标函数值最优的点作为最终计算结果。

4 GATS 混合算法模拟实验

4.1 静态调度

仿真试验中,以10个零件、10台机器的调度问题为例^[13]。下面给出加工时间阵、机器顺序阵,以及 GATS 算法计算10次所得的最优工件排序阵及其对应的最优加工 Gantt 图(图2),最优解为87个单位时间。同时,该调度问题的 Gantt 图也明显地反映了此问题的求解复杂性。通过 Gantt 图可以清晰地了解到在每台机器上加工的工件及其各工序加工的起始时间和终止时间。如果调度的最佳解不唯一,则该调度算法同样也可以将它们全部搜索到,该算例的最佳调度结果就不是唯一的,图2只是其中的一个最佳调度的 Gantt 图。

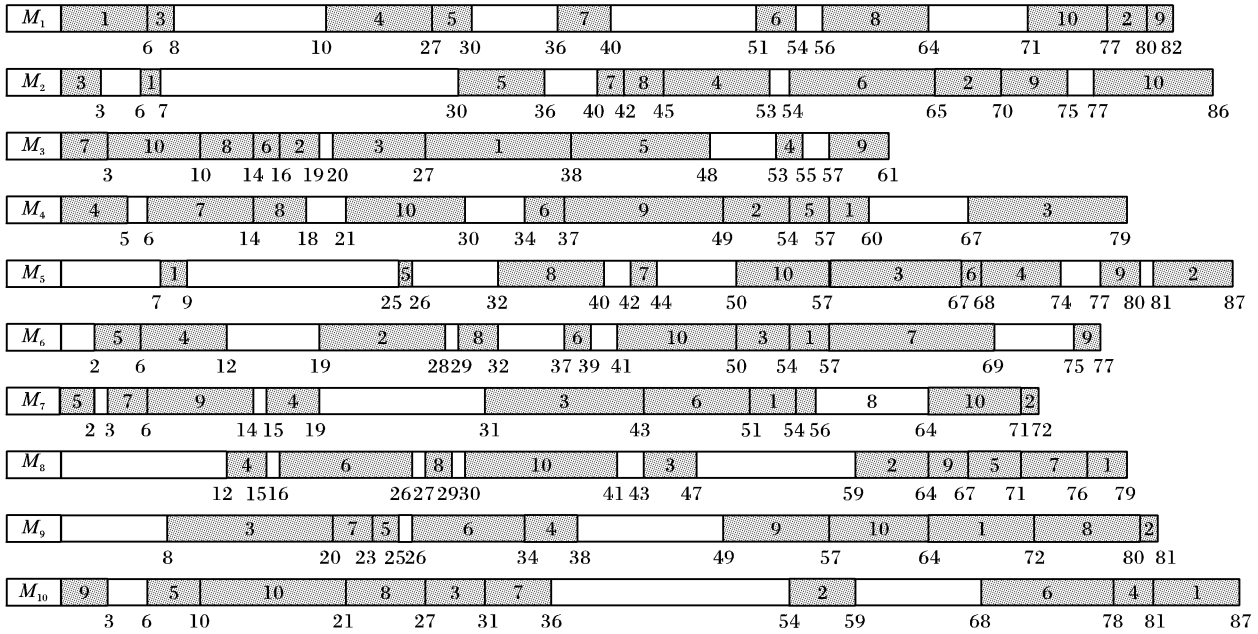


图2 最优调度结果 Gantt 图

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

$$J_M = \begin{bmatrix} 1 & 2 & 5 & 3 & 7 & 6 & 4 & 9 & 8 & 10 \\ 3 & 6 & 4 & 10 & 8 & 2 & 7 & 1 & 9 & 5 \\ 2 & 1 & 9 & 3 & 10 & 7 & 8 & 6 & 5 & 4 \\ 4 & 6 & 8 & 7 & 1 & 9 & 2 & 3 & 5 & 10 \\ 7 & 6 & 10 & 9 & 5 & 1 & 2 & 3 & 4 & 8 \\ 3 & 8 & 9 & 4 & 6 & 7 & 1 & 2 & 5 & 10 \\ 3 & 7 & 4 & 9 & 10 & 1 & 2 & 5 & 6 & 8 \\ 3 & 4 & 10 & 8 & 6 & 5 & 2 & 7 & 1 & 9 \\ 10 & 7 & 4 & 9 & 3 & 8 & 2 & 6 & 5 & 1 \\ 3 & 10 & 4 & 8 & 6 & 5 & 9 & 7 & 1 & 2 \end{bmatrix}$$

$$M_J = \begin{bmatrix} 1 & 3 & 4 & 5 & 7 & 6 & 8 & 10 & 2 & 9 \\ 3 & 1 & 5 & 7 & 8 & 4 & 6 & 2 & 9 & 10 \\ 7 & 10 & 8 & 6 & 2 & 3 & 1 & 5 & 4 & 9 \\ 4 & 7 & 8 & 10 & 6 & 9 & 2 & 5 & 1 & 3 \\ 1 & 5 & 8 & 7 & 10 & 3 & 6 & 4 & 9 & 2 \\ 5 & 4 & 2 & 8 & 6 & 10 & 3 & 1 & 7 & 9 \\ 5 & 7 & 9 & 4 & 3 & 6 & 1 & 8 & 10 & 2 \\ 4 & 6 & 8 & 10 & 3 & 2 & 9 & 5 & 7 & 1 \\ 3 & 7 & 5 & 6 & 4 & 9 & 10 & 1 & 8 & 2 \\ 9 & 5 & 10 & 8 & 3 & 7 & 2 & 6 & 4 & 1 \end{bmatrix}$$

对仿真程序做了 10 次运行,记录下了每次得到的目标函数最小值,如表 1 所示。表 1 的结果表明这个算法在 10 次运行中有 2 次达到最小值,而且最差的值与最优值之间的偏差也只有 3,结果波动不大。

表 1 仿真试验 10 次运行结果

目标函数	次数									
	1	2	3	4	5	6	7	8	9	10
最小值	90	87	90	89	89	88	90	89	90	87
GATS	90	87	90	89	89	88	90	89	90	87
GA	103	96	100	100	96	95	99	96	99	97

为了验证算法的有效性,在交叉算法和变异算法以及一些参数设置相同的情况下,采用遗传算法重复运算 10 次得到的最优解为 95,与最差值之间的偏差为 8。同时可以比较出,禁忌搜索混合遗传算法比传统的遗传算法有较好的稳定性。

4.2 动态调度

动态调度系统考虑了重调度时各台机器的当前状态,从而可使静态调度与扰动后的调度有效衔接起来,避免了工序冲突现象的发生,提高了柔性制造系统的生产效率。目标函数和其他参数的选取同静态调度。假设 15h 后来了一个紧急零件(101)需要加工,需要改变原来的调度方案。对已加工完的工件的工序不予考虑。正在加工工件的机器不能够停下来,必须等该工件的这道工序加工完之后,才能考虑这台机器上的重新调度。例如,机器 10 在该时刻正在加工工件 10,必须等工件 10 在该机器上的工序加工完以后才能考虑该机器上工件的重新调度即 21h 后。紧急工件(101)的加工时间阵为(6 0 11 3 2 3 0 0 8 0);机器顺序阵为(3 5 1 4 9 6 0 0 0)。根据图 2 的 Gantt 图得到各台机器的起始时间点分别为 15、15、16、18、15、15、15、20、21。已经完成或正在加工的工件工序信息为((1,1)(3,2))_{M1}、((3,1)(1,2))_{M2}、((7,1)(10,1)(8,1)(6,1))_{M3}、((4,1)(7,3)(8,2))_{M4}、((1,3))_{M5}、((5,2)(4,2))_{M6}、((5,1)(7,2)(9,2))_{M7}、(4,3)_{M8}、(3,3)_{M9}、(9,1)(5,3)(10,2))_{M10}。((1,1)(3,2))_{M1} 表示为第 1 台机器上完成或正在加工的工件信息为第 1 个工件的第 1 道工序和第 3 个工件的第二道工序。工件排列阵变为和 M_i' 。得到的最优调度结果如图 3 所示,其中深色阴影部分为紧急工件到来时正在加工中的工件。矩阵中为 0 的部分表明紧急事件不在该机器上加工。

通过 Gantt 图,可以清楚了解工序和机床的分配关系,其

间还标明了每台机器上加工的工件的起始时间。图中横坐标表明了这批工件加工的时间历程,纵坐标表明机床设备。从图 3 中可以看出当紧急事件到来时工件 6、工件 8、工件 3 和工件 10 正分别在机器 3、机器 4、机器 9 和机器 10 上加工;所以必须等这四个工件加工完才能考虑机器 3、4、9 和机器 10

上的重新调度。从图 3 和矩阵 M_i' 可以观察到紧急事件没有在机器 2、7、8、和机器 10 上加工。如果采用静态的车间调度策略来处理文中提到的环境变化(即有紧急事件到达时),显然是行不通的。因为静态调度无法对环境的变化做出相应的变化。从仿真的结果看,文中提出的动态调度算法完全可行。

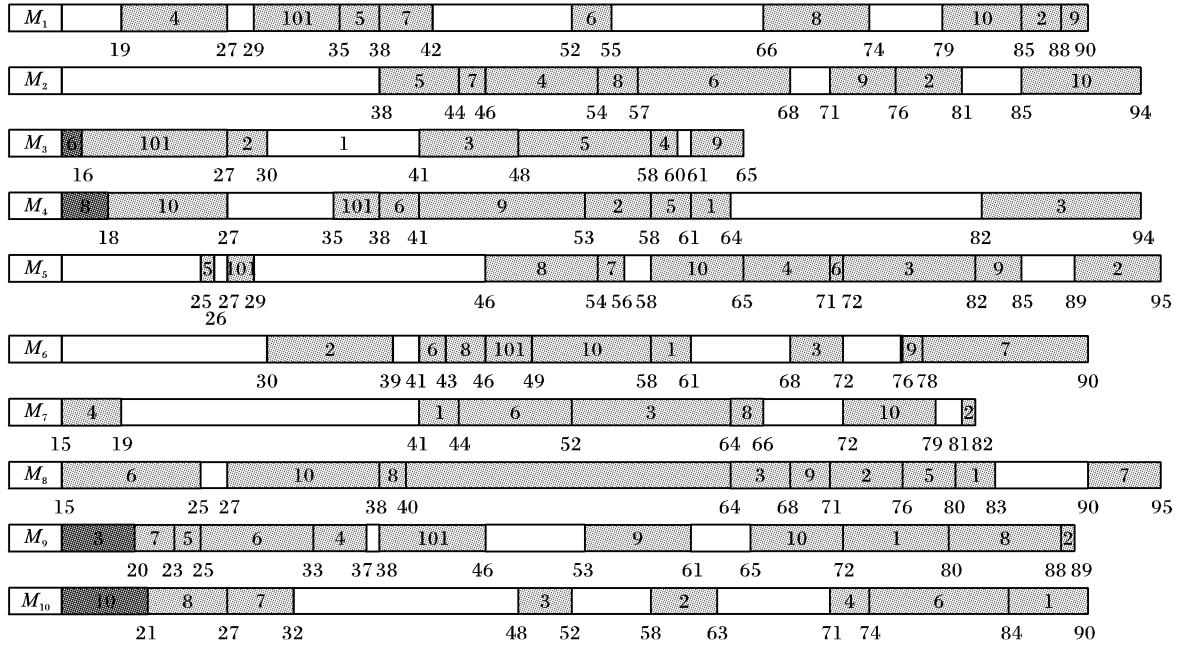


图 3 扰动后的 Gantt 图

$$M_i' = \begin{bmatrix} 1 & 3 & 4 & 101 & 5 & 7 & 6 & 8 & 10 & 2 & 9 \\ 3 & 1 & 5 & 7 & 4 & 8 & 6 & 9 & 2 & 10 & 0 \\ 7 & 10 & 8 & 6 & 101 & 2 & 1 & 3 & 5 & 4 & 9 \\ 4 & 7 & 8 & 10 & 101 & 6 & 9 & 2 & 5 & 1 & 3 \\ 1 & 5 & 101 & 8 & 7 & 10 & 4 & 6 & 3 & 9 & 2 \\ 5 & 4 & 2 & 6 & 8 & 101 & 10 & 1 & 3 & 9 & 7 \\ 5 & 7 & 9 & 4 & 1 & 6 & 3 & 8 & 4 & 2 & 0 \\ 4 & 6 & 10 & 8 & 3 & 9 & 2 & 5 & 1 & 7 & 0 \\ 3 & 7 & 5 & 6 & 4 & 101 & 9 & 10 & 1 & 8 & 2 \\ 9 & 5 & 10 & 8 & 7 & 3 & 2 & 4 & 6 & 1 & 0 \end{bmatrix}$$

5 结语

本文针对流程工业中的 job shop 调度问题,建立了相应的数学模型。在调度模型的基础上,将遗传算法与禁忌搜索算法结合起来;而且还考虑了有紧急工件加入的动态调度,根据静态调度的 Gantt 图对动态调度进行仿真。仿真试验表明了该算法不仅有效,而且提高了搜索精度。

参考文献:

[1] GAREY M, JOHNSON D, SETHO R. The complexity of flowshop and jobshop scheduling[J]. Mathematics of Operations Research, 1976, 1(1): 117 - 129.
 [2] BLAZEWICZ J, ECKER K H, SCHMIDT G, et al. Scheduling in Computer and Manufacturing Systems[M]. Second Revised Edition. Berlin: Springer-Verlag, 1996.
 [3] LEE DY, DICESARE F. Scheduling flexible manufacturing systems using Petri nets and heuristic search[J]. IEEE Transaction on Robotics and Automation, 1994, 10(2): 123 - 132.
 [4] HOLSAPPLEW, JACOB VS, PAKATH R, et al. A genetics-based

hybrid scheduler for generating static schedules in flexible manufacturing contexts[J]. IEEE Transaction on Systems, Man and Cybernetics, 1993, 23(4): 953 - 972.
 [5] 王凌. 车间调度及其遗传算法[M]. 北京: 清华大学出版社, 2002.
 [6] MATTFELD DC, BIERWIRTH C. An efficient genetic algorithm for job shop scheduling with tardiness objectives[J]. European Journal of Operational Research, 2004, 155(3): 616 - 630.
 [7] ESQUIVEL S, FERRERO, GALLARD R, et al. Enhanced evolutionary algorithms for single and multiobjective optimization in the job scheduling problem[J]. Knowledge-Based Systems, 2003, 15(1 - 2): 13 - 25.
 [8] 吴云高, 王万良. 基于遗传算法的混合 Flowshop 调度[J]. 计算机工程与应用, 2002, 38(12): 82 - 84.
 [9] SOUSEN L, SHAOTING H. GA-based resource- constrained flowshop scheduling model for mixed precast production[J]. Automation in Construction, 2002, 11: 439 - 452.
 [10] GLOVER F, KELL YJ, LAGUNA M. Genetic algorithm and tabu Search: hybrids for optimizations[J]. Computers and Ops. Res, 1995, 22(1): 111 - 134.
 [11] LAFUNA M, BARNES JW, GLOVER F. Tabu search methods for a single machine scheduling proplem[J]. Journal of Intelligent Manufacturing, 1991, 2: 63 - 74.
 [12] LAFUNA M, BARNES JW, GLOVER F. Intelligent scheduling with tabu search: an application to jobs with linear delay penalties and sequence - dependent setup costs and times[J]. Journal of Applied Intelligence, 1993, 3: 159 - 172.
 [13] 王万良, 吴启迪. 求解作业车间调度问题的改进自适应遗传算法[J]. 系统工程理论与实践, 2004, 24(2): 58 - 62.