

文章编号: 1001-0920(2007)08-0931-04

## 用混合遗传算法求解虚拟企业生产计划

高 阳, 江资斌

(中南大学 商学院, 长沙 410083)

**摘 要:** 针对虚拟企业生产计划的特点, 以各成员企业承担的生产任务为对象, 以快速响应市场为目标, 建立了生产任务计划的数学模型, 并基于该模型, 提出一种基于遗传算法与模拟退火算法混合的求解算法, 充分发挥了遗传算法良好的全局搜索能力和模拟退火算法有效避免陷入局部极小的优点, 从而提高了算法的全局寻优能力. 数值仿真计算表明了该算法的良好收敛性和有效性.

**关键词:** 虚拟企业; 生产计划; 遗传算法; 模拟退火

**中图分类号:** TP301

**文献标识码:** A

### Using HGA to solve production planning in VE

GAO Yang, JIANG Zi-bin

(School of Business, Central South University, Changsha 410083, China. Correspondent: GAO Yang, E-mail: gaoyangcsu@163.com)

**Abstract:** Aiming at the special feature of virtual enterprises (VE), taking into account the information of tasks that partners undertake and considering quickly responding to market as optimizing objective, a mathematical model is proposed. Based on this model, a hybrid genetic algorithms (HGA) is presented. It makes full use of the excellent whole search ability of genetic algorithm (GA), and the advantage of simulated annealing algorithm (SA) can avoid getting into part minimum efficiently, thus the global searching ability of HGA is improved. The numerical simulation shows the good convergence and effectiveness of this algorithm.

**Key words:** Virtual enterprises; Production planning; Genetic algorithm; Simulated annealing

### 1 引 言

虚拟企业 (VE) 作为敏捷制造的有效组织形式, 其主要目的是为了迎合快速变化的市场机遇<sup>[1-3]</sup>. 在组建虚拟企业时, 产品的生产通常要根据所需核心资源情况粗略地分解成若干个生产任务<sup>[4]</sup>; 然后, 核心企业通过招标方式为每个生产任务寻求合适的合作伙伴, 并签订协议, 完成任务分配. 当每项招标任务都找到合适的合作伙伴后, 就建立了虚拟企业. 由于各企业所获得的都是比较粗略的生产任务, 而他们在承担这些任务的同时, 还有自身的生产活动, 各企业需根据其获得的任务的要求及自身生产能力和可用资源情况, 将任务进一步分解为一系列生产活动或作业, 并完成相应的生产计划, 然后反馈给核心企业. 核心企业再将各个任务的生产计划综合集成为虚拟企业的总体生产计划. 为了快速响应市场, 核心企业要在成员企业制定的生产任务计划的基础上

进行整体优化, 优化目标最重要的是使产品生产延续时间最短. 文献[5, 6]研究了基于 HMS (Holonc Manufacturing System) 模型的虚拟企业生产计划问题, 其模型主要是研究虚拟企业生产计划的层次概念. 尽管 HMS 模型可进一步转换为 Petri 网模型<sup>[7]</sup>, 但随着图的规模增大, 确定优化方案需耗时很多, 甚至无法得到解. 文献[8]建立了虚拟企业生产计划的数学模型, 并提出了一个启发式的后向计划算法, 由于不同启发式规则适用的问题和环境不同, 使用它一般得不到最优的方案.

实际上, 虚拟企业生产任务的全局优化类似于作业车间优化问题, 而关于作业车间优化问题, 已有丰富的研究成果<sup>[9]</sup>. 但由于虚拟企业任务之间的约束关系比较复杂, 如某些任务可能存在多个紧前或紧后任务, 同时需考虑成员企业地理位置分散的特点, 已有的解决 Job Shop 问题的算法需要进一步扩

收稿日期: 2006-04-28; 修回日期: 2006-07-07.

基金项目: 国家自然科学基金项目 (70572060); 教育部高等学校博士点专项科研基金项目 (20040533057).

作者简介: 高阳 (1943 →), 男, 湖南望城人, 教授, 博士生导师, 从事计算机集成制造系统、智能管理系统等研究;  
江资斌 (1976 →), 男, 湖南浏阳人, 博士生, 从事虚拟企业生产运作管理、智能管理系统的研究.

展才能应用于虚拟企业的生产计划. 本文针对虚拟企业生产计划的特点, 建立了生产任务计划的数学模型. 为求解模型, 采用基于任务编号编码, 并构造约束矩阵, 有效解决了任务之间复杂的约束关系, 设计并给出一种混合遗传算法, 充分发挥遗传算法所具有的良好全局搜索能力和模拟退火算法能有效避免陷入局部极小的优点, 以提高算法的速度和全局收敛性.

## 2 问题描述和模型建立

考虑具有以下特点的虚拟企业生产计划问题:

- 1) 生产由一些作业时间已知的任务组成;
- 2) 每个任务的开始时间依赖于其他一些任务的完成(时序约束), 生产任务在执行期间不中断;
- 3) 任务执行的企业已由任务分配阶段确定, 由于各企业在承担虚拟企业任务的同时, 还有自身的生产活动, 考虑到其生产能力和资源约束, 企业一次只执行一个虚拟企业的生产任务;
- 4) 计划的目标是最小化生产延续时间.

设  $N = \{1, 2, \dots, n\}$  为需计划的生产任务编号集,  $M = \{1, 2, \dots, m\}$  为企业编号集,  $E = \{E_1, E_2, \dots, E_m\}$  为承担生产任务的企业集,  $T = \{T_1, T_2, \dots, T_n\}$  为生产任务集,  $t_{ei}$  表示生产任务  $T_i$  在企业  $E_e$  中作业时的完成时间,  $t_i (i \in N)$  为生产任务  $T_i$  的作业时间,  $t_{pi} (i \in N)$  为生产任务  $T_i$  的计划开始时间,  $H_i$  为生产任务  $T_i$  的紧前任务集,  $B_i$  为生产任务  $T_i$  的紧后任务集.

设  $S_{ki}$  为  $k \in H_i$  时, 生产任务  $k$  完成后到生产任务  $i$  作业点的运输时间,  $PT_{et}$  为企业  $E_e$  在时间  $t$  作业的生产任务,  $WT_{et}$  为企业  $E_e$  在时间  $t$  等待作业的生产任务集.

假设生产在零时间开始, 根据上述符号, 虚拟企业的生产计划问题可归结为对于每项生产任务  $T_i$  确定  $t_{pi}$ , 使整个产品生产所需时间最短. 该问题的数学模型描述为

$$\min J = \max_{1 \leq e \leq m} \max_{1 \leq i \leq n} t_{ei}, \tag{1}$$

$$\text{s. t. } t_{ei} = t_{pi} + t_i, i \in N, e \in M; \tag{2}$$

$$t_{ePT_{et}} - \min(t_{pWT_{et}}) \leq 0, \tag{3}$$

$$PT_{et}, WT_{et} \in N; \tag{3}$$

$$t_{pi} + t_i + \max(S_{ib}) - \min(t_{pb}) \leq 0, \tag{4}$$

$$b \in B_i; \tag{4}$$

$$\max(t_{pk} + t_k + S_{ki}) - t_{pi} \leq 0, \tag{5}$$

$k \in H_i$ .  
式(1)表示目标函数, 即制造周期, 指明所计划的最后一个生产任务完成时间尽可能早; 式(2)为生产任务开始时间与完成时间的关系; 式(3)为同

一个企业内部任务正在作业的任务完成时间与下一个将作业的任务时间约束关系; 式(4)为生产任务到达时间和紧后任务开始时间关系; 式(5)为生产任务的开始时间和紧前任务的到达时间关系. 由于模型的目标函数是非连续可微的, 不能用一般的数学规划方法进行求解. 可以看出该问题是一个组合优化问题, 是 NP-hard 难题<sup>[10]</sup>.

## 3 混合遗传算法求解模型

### 3.1 编码

采用基于任务编号的染色体表示方法. 将每个企业承担的生产任务编号排成一个子顺序, 然后将所有的企业生产任务编号串起来形成染色体, 如图 1 所示, 染色体的每个基因为生产任务的数值编号.

企业:  $E_1 \quad E_2 \quad E_3 \quad E_4 \quad E_5 \quad E_6$   
染色体: 31 27 22 5 4 1 30 26 21 11 6 ... ..

图 1 染色体编码

### 3.2 启发式规则产生初始种群

启发式算法由于运用了经过检验的优先规则, 所生成的基因串具有较高的适应度. 算法初始种群的个体由两部分组成: 一部分基于优先规则, 如作业时间最短优先、作业时间最长优先、任务编号大者优先和任务编号小者优先等产生, 启发式规则用于产生初始种群, 可得到适应度较高的个体; 另一部分则随机产生, 随机产生的个体可保证初始个体的多样性.

设  $C$  为生产任务之间的作业次序约束矩阵, 若编号为  $i$  的生产任务在  $j$  之前完成, 则  $C_{ij} = 1$ ; 若编号为  $i$  的生产任务在  $j$  之后完成, 则  $C_{ij} = -1$ ; 若它们之间没有次序约束, 则  $C_{ij} = 0$ , 同时约定  $C_{ii} = 0$ . 不论用何种方法产生的染色体, 其基因子顺序均应满足  $C_{ij} = 1$  或  $C_{ij} = 0$ .

### 3.3 个体评价

因为采用总的生产延续时间最短作为计划问题的评价指标, 而遗传算法通常进化的方向是适应度函数增大的方向, 所以要通过以下变换先将目标函数映射成适应度函数. 设  $F_{\max}$  为初始种群中最大的任务完成时间, 第  $i$  个计划方案的任务完成时间为  $F_i$ , 其适应度评价函数为

$$\begin{cases} f_i = F_{\max} - F_i, & F_i < F_{\max}; \\ f_i = 0, & \text{otherwise.} \end{cases} \tag{6}$$

### 3.4 遗传算子

#### 3.4.1 选择算子

采用轮盘赌选择算子与精英保留策略相结合, 首先按轮盘赌选择方法执行遗传算法的选择操作; 然后将当前群体中适应度最高的个体结构完整地复

制到下一代群体中。

### 3.4.2 交叉算子

为使交叉满足模型的约束及提高交叉的效率，采用多重顺序交叉 (MOX)。即先产生一个随机数，表示发生交叉的基因段，每一基因段为同一企业所承担的任务编号串，随机选择同样数量的基因段，对每个基因段进行顺序 (OX) 交叉<sup>[10]</sup>。执行交叉后，利用约束矩阵  $C$  对每个基因段进行合法性检查，若排在前面的任务和排在后面的任务不满足  $C_{ij} = 1$  或  $C_{ij} = 0$ ，则将它们进行交换，直到所有基因段和所有的任务均满足约束条件。

### 3.4.3 变异算子

在基于任务编号编码的染色体中，普通的变值变异方法不适用，可采用移动与交换作为变异。移动变异将一个随机选择的基因向左或向右移动一个随机的位置。交换变异是随机选择两个基因进行变换。通过约束矩阵检查，若  $C_{ij} = 0$ ，则可进行移动或交换。

### 3.5 培 育

遗传操作每进行  $B_r$  次后，选择其中的  $S_e$  个最优个体，对其进行模拟退火操作，使它们获得更优良的特性，加快遗传算法的收敛，并将培育后的优良染色体返回原来的进化过程中，继续进行遗传进化。SA 的初始启动温度一次比一次低，以此来模拟混合算法的全局冷却。每次 SA 运行将在其退火过程中降低初始温度。

### 3.6 仿真算法

在基于任务编号编码的染色体个体中，任务的最终完成时间无法直接从染色体求得，即个体的适应度评价函数值无法通过解码得到。为此，本文设计了以下仿真算法来得到生产任务的制造周期。

Step1: 初始化任务数  $n$ ，计划没有紧后任务的生产任务，令其完成时间为零，其开始时间为完成时间减去作业时间，设置其计划状态为真，设置  $F$  为其开始时间， $n = n - 1$ 。

Step2: 选择基因段  $e$ 。

Step3: 按顺序选择任务编号，看其计划状态是否为真，如果为真转 Step4；否则看其紧后任务的计划状态是否为真。如果为假转 Step4；如果为真，该任务的完成时间  $t_{ei}$  为其紧后任务开始时间减去该任务到紧后任务的运输时间之差和  $E_e$  前一个已计划任务的开始时间中最小者。置  $t_{pi} = t_{ei} - t_i$ ，并置其计划状态为真， $n = n - 1$ ，比较  $F$  与  $t_{pi}$ ，如果  $F < t_{pi}$ ，则令  $F = t_{pi}$ 。

Step4: 如果搜索到基因段尾，转 Step5；否则转 Step3。

Step5: 如果  $n = 0$ ，转 Step6；否则考虑下一个基因段， $e = e + 1$ 。如果  $e > m$ ， $e = 1$ ，转 Step2。

Step6: 将所有任务的开始时间加上  $|F|$ ， $|F|$  即制造周期，通过式 (6) 将其转换为适应度，算法结束。

### 3.7 混合遗传算法流程

模型求解整个混合遗传算法步骤如下：

Step1: 初始化遗传算法参数。设置种群大小 pop-size，杂交概率  $p_c$ ，变异概率  $p_m$ ，最大繁殖代数 max\_gen 和初始繁殖代数  $gen = 0$ ；设置培育参数  $B_r$  和  $S_e$ 。

Step2: 基于任务编号编码，生产约束矩阵  $C$ ，利用启发式规则和随机方法产生合法的初始化种群。

Step3: 启动仿真算法，得到各染色体的适应值和计划方案。

Step4: 如果最优解满意或终止条件  $gen \leq \max\_gen$  满足，转 Step8。

Step5: 如果  $gen \% B_r = 0$  且  $gen > 0$ ，则选择  $S_e$  个优良染色体；否则转 Step7。

Step6: 对  $S_e$  中每个染色体用 SA 进行培育操作：1) 初始化 SA 参数，确定初温  $t_{p0}$  和温度下降速率；2) 执行 SA，得到优良染色体，插入当前种群。

Step7: 遗传操作，执行选择、交叉、变异算子， $gen = gen + 1$ ，转 Step3。

Step8: 输出最优解和计划方案，算法终止。

### 4 数值仿真

现以某大型设备的生产为例。企业  $E_1$  无法单独完成其生产，将其生产与装备过程分解为 31 个生产任务，并选择了另外 5 家企业一起组建虚拟企业，由  $E_1$  执行任务  $\{T_1, T_4, T_5, T_{22}, T_{27}, T_{31}\}$ ， $E_2$  执行任务  $\{T_6, T_{11}, T_{21}, T_{26}, T_{30}\}$ ， $E_3$  执行  $\{T_2, T_{10}, T_{14}, T_{25}, T_{29}\}$ ， $E_4$  执行  $\{T_7, T_{12}, T_{16}, T_{17}, T_{28}\}$ ， $E_5$  执行  $\{T_3, T_9, T_{13}, T_{19}, T_{24}\}$ ， $E_6$  执行  $\{T_8, T_{15}, T_{18}, T_{20}, T_{23}\}$ 。各个任务作业次序约束有向图如图 2 所示，其中圆角矩形框下的数字为各企业任务计划反馈的执行时间，箭头上标注的为运输时间。如果当前任务和其紧后任务在同一个企业内加工，忽略其运输时间。 $T_0$  为虚设任务，以便于仿真算法的搜索。

利用 Visual C++ 编程，以任务为对象，利用其承担企业、执行时间、紧前任务、紧后任务、运输时间等信息，实现前述混合遗传算法 HGA。HGA 种群大小为 20，染色体长度为 31，最大进化代数为 30，交叉概率为 0.8，变异率为 0.05，每进化 5 代就启动一次 SA 培育过程，SA 的每次初始温度  $t_{p0}$  为 GA 当前最好的制造周期乘以  $I$ ， $I = 5, 4, 3, 2, 1, 0.5$ ，对应于进化代数  $gen = 5, 10, 15, 20, 30$ 。SA 温度下降速率为

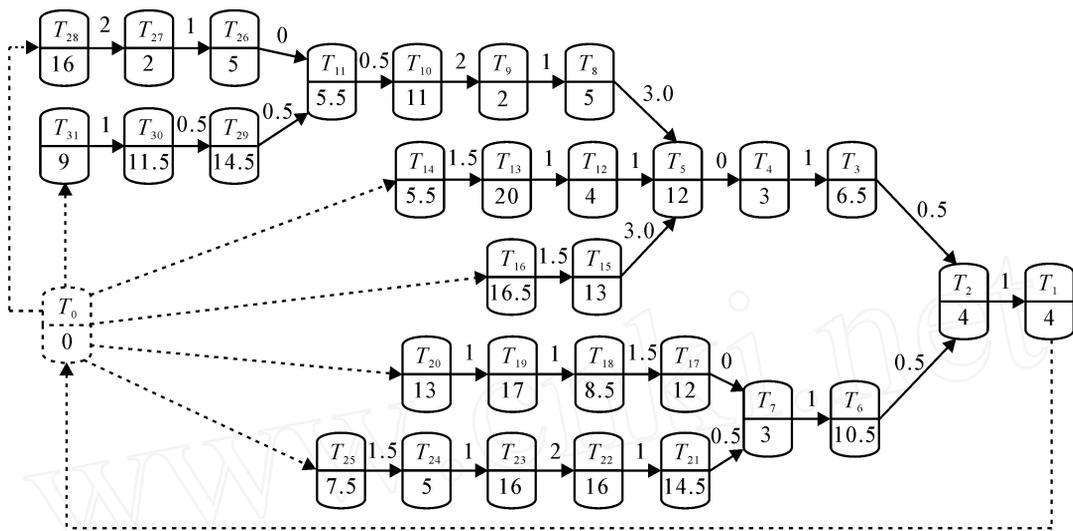


图2 任务作业次序约束的有向图表示

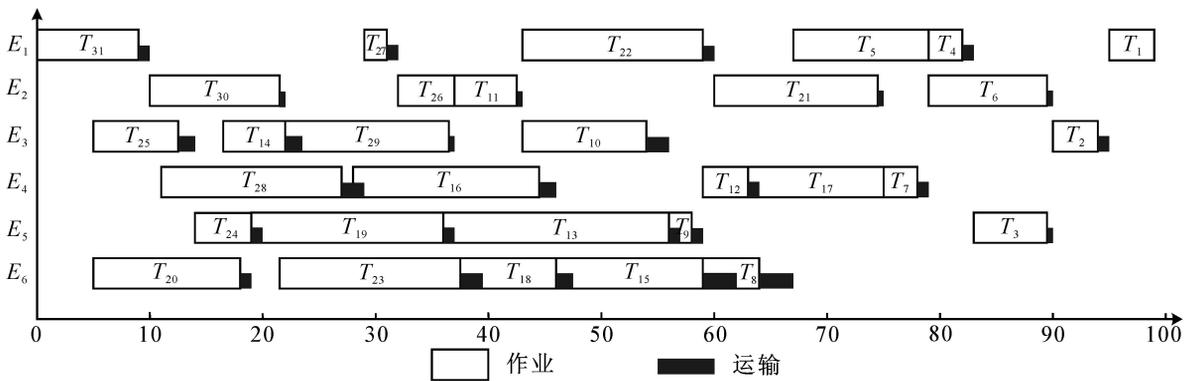


图3 混合遗传算法求得的最优计划方案

表1 本文算法与文献[9] GASA 的性能比较

Problem	n, m	GASA			HGA		
		$\bar{C}$	$C_{best}$	$\bar{t}$	$\bar{C}$	$C_{best}$	$\bar{t}$
MT06	6, 6	55	55	2.4	55	55	0.8
MT10	10, 10	930.4	930	45.3	930.2	930	25.5
MT20	20, 5	1 166.2	1 165	88.5	1 165.2	1 165	48.6

注:  $m$  为机器数,  $n$  为工件数,  $\bar{C}$  为 10 次所得解的平均值,  $C_{best}$  为得到的最优解,  $\bar{t}$  为得到最优解的平均时间。

0.9, SA 运行后把当前解和制造周期返回 GA 中。算法在 Pentium IV 机上验证, 运行 10 次, 平均耗时 6 ~ 8 s, 最终以甘特图输出最优计划方案, 如图 3 所示, 所求得的最优制造周期为 99 个时间单位。

为进一步验证算法的有效性, 将本文的 HGA 与文献[9]的 GASA 就经典问题随机运行 10 次的优化结果进行比较, 如表 1 所示。

由表 1 可以看出, 本文提出的混合遗传算法能有效地求解 Job Shop 问题。文献[9]中的 GASA 每进化一代, 选择种群大小的优良个体执行 SA 操作; 而 HGA 每进化  $B_r$  代后, 选择  $S_e$  个优良个体进行 SA 操作, 且初始启动温度变化大, 一次比一次低, 因而耗时相对文献[9]中的 SA 操作要少。另外, 作业

次序约束矩阵  $C$  的引入简化了 GA 和 SA 的操作。这均使本文算法具有更快的收敛速度。

### 5 结 语

本文提出了一种基于混合遗传算法的虚拟企业生产计划求解方法, 为解决虚拟企业生产运作问题提供了技术支持, 其算法结构简洁, 运算速度快, 具有一定的实用价值。需要指出的是, 虚拟企业的生产计划是一个反复优化的过程, 核心企业得到优化的计划方案后, 需将其传递给各个成员企业, 如果成员企业需要调整任务, 则需要重新协商, 核心企业则根据任务信息再次进行全局优化, 直至最后一致同意。

(下转第 938 页)

和镇定,也可作为一种异结构混沌同步手段。

### 参考文献(References)

- [1] 刘延柱,陈立群. 非线性振动[M]. 北京: 高等教育出版社, 2001.  
(Liu Yan-zhu, Chen Li-qun. Nonlinear vibrations[M]. Beijing: High Education Press, 2001.)
- [2] 王光瑞,于熙龄,陈式刚. 混沌的控制、同步与利用[M]. 北京: 国防工业出版社, 2001.  
(Wang Guang-rui, Yu Xi-ling, Chen Shi-gang. Chaos control, synchronization and application[M]. Beijing: National Defence Industry Press, 2001.)
- [3] Pecora L M, Carroll T L. Synchronization in chaotic circuits[J]. Physical Review Letters, 1990, 64(8): 821-824.
- [4] Liao T L. Adaptive synchronization of two Lorenz systems[J]. Chaos, Solitons and Fractals, 1998, 9(9): 1555-1561.
- [5] El-Gohary A. Optimal synchronization of Rossler system with complete uncertain parameters[J]. Chaos, Solitons and Fractals, 2006, 27(2): 345-355.
- [6] 李丽香,彭海朋,卢辉斌. Henon 混沌系统的追踪控制与同步[J]. 物理学报, 2001, 50(4): 629-632.  
(Li Li-xiang, Peng Hai-peng, Lu Hui-bin. Control and synchronization of the Henon chaotic system[J]. Acta Physica Sinica, 2001, 50(4): 629-632.)
- [7] 王兴元,石其江. Rossler 系统的追踪控制[J], 物理学报, 2005, 54(12): 5591-5596.  
(Wang Xing-yuan, Shi Qi-jiang. Tracking control and synchronization of the Rossler's chaotic system[J]. Acta Physica Sinica, 2005, 54(12): 5591-5596.)
- [8] Lü H, Zhou T S, Zhang S C. Controlling the Chen attractor using linear feedback based on parametric identification[J]. Chinese Physics, 2002, 11(1): 12-16.
- [9] Li Z, Han C Z. Adaptive synchronization of Rossler and Chen chaotic system[J]. Chinese Physics, 2002, 11(7): 666-669.
- [10] 宁娣,陆君安. 一个临界系统与 Lorenz 系统和 Chen 系统的异结构同步[J]. 物理学报, 2005, 54(10): 4590-4595.  
(Ning Di, Lu Jun-an. Synchronization of a critical system with Lorenz system and Chen system[J]. Acta Physica Sinica, 2005, 54(10): 4590-4595.)
- [11] Chen G R, Ueta T. Yet another chaotic attractor[J]. Int J of Bifurcation and Chaos, 1999, 9(3): 1465-1466.

(上接第 934 页)

### 参考文献(References)

- [1] Martinez M T, Fouletier P, Park K H, et al. Virtual enterprise — Organization, evolution and control [J]. Int J Production Economics, 2001, 74(1-3): 225-238.
- [2] Rajiv Kishore, Ephraim R McLean. The next generation enterprise a CIO perspective on the vision, its impacts, and implementation challenges[J]. Information Systems Frontiers, 2002, 4(1): 121-138.
- [3] David Walters. Performance planning and control in virtual business structures[J]. Production Planning and Control, 2005, 16(2): 226-239.
- [4] Wu N Q, Sun J. Grouping the activities in virtual enterprise paradigm [J]. Production Planning and Control, 2002, 13(4): 407-415.
- [5] Duncan C McFarlane, Stefan Bussmann. Developments in holon production planning and control[J]. Production Planning and Control, 2000, 11(6): 522-536.
- [6] Huang B Q, Gou H M, Liu W H. A framework for virtual enterprise control with the holonic manufacturing paradigm[J]. Computers in Industry, 2002, 49(3): 299-310.
- [7] Hsieh Fur-Shiung. Model and control holonic manufacturing systems based on fusion of contract nets and Petri nets[J]. Automatica, 2004, 40(1): 51-57.
- [8] 苏平,伍乃骥. 一种可重构制造系统的生产计划方法[J]. 计算机集成制造系统, 2003, 9(3): 189-193.  
(Su Ping, Wu Nai-qi. A method for production scheduling in reconfigurable manufacturing systems [J]. Computer Integrated Manufacturing Systems, 2003, 9(3): 189-193.)
- [9] 王凌. 车间调度及其遗传算法[M]. 北京: 清华大学出版社, 2003.  
(Wang Ling. Shop scheduling with genetic algorithms [M]. Beijing: Tsinghua University Press, 2000.)
- [10] 陈国良,王煦法,庄镇泉,等. 遗传算法及其应用[M]. 北京: 人民邮电出版社, 2001.  
(Chen Guo-liang, Wang Xu-fa, Zhuang Zhen-quan, et al. Genetic algorithm and its application [M]. Beijing: Posts and Telecommunications Publishing House, 2001.)