

基于禁忌搜索遗传混合算法的装配线平衡

余晓光, 严洪森

(东南大学 复杂工程系统测量与控制教育部重点实验室, 江苏 南京 210096;
东南大学 自动化学院, 江苏 南京 210096)

摘要:针对装配线平衡问题(ALBP),文中提出了一种禁忌搜索遗传混合算法。在混合算法中,遗传算法部分采用特殊的遗传变异操作算子(双点交叉和移位插入变异),使算法只在可行作业序列子空间中进行搜索,有效减小了搜索范围,提高了算法运行效率;禁忌搜索部分是在每代遗传操作完成以后,随机选择一些个体进行禁忌搜索操作,来增强算法的搜索能力。最后以经典问题的求解验证了禁忌搜索遗传算法在收敛性能和计算效率上较使用单纯的遗传算法高。

关键词:装配线平衡;遗传算法;禁忌搜索;混合算法

中图分类号:TP391;TH166

文献标识码:A

文章编号:1673-629X(2010)05-0005-04

Assembly Line Balancing Based on Tabu Search and Genetic Hybrid Algorithm

YU Xiao-guang, YAN Hong-sen

(Ministry of Education Key Laboratory of Measurement and Control of CSE,

Southeast University, Nanjing 210096, China;

School of Automation, Southeast University, Nanjing 210096, China)

Abstract: A tabu search and genetic hybrid algorithm is presented to solve assembly line balancing problem (ALBP). In its genetic algorithm part, special genetic operators (two point crossover and insert mutation) is adopted to scan only in the subspace of those feasible task sequences, which reduces search area and improves algorithm efficiency. In its tabu search part, after evolution of each generation, some random selected individuals execute tabu search, which can improve algorithm search ability. In the end of this paper, the calculating results of several classic problems prove that the tabu search and genetic hybrid algorithm is better than the genetic algorithm in search ability and convergence performance.

Key words: assembly line balancing; genetic algorithm; tabu search; hybrid algorithm

0 引言

装配线这种生产组织方式起始于1914年美国福特公司的T型车生产线。在装配线上,工件从一个工位向另一个工位流动。每个工位固定完成一项或几项装配任务。随着生产的发展,装配线广泛应用于车辆、电器、机械设备等生产领域。

从Bryton和Salveson提出装配线平衡问题(Assembly Line Balancing Problem ALBP)后,逐步有了解析法、线性规划法、启发式算法和人工智能算法等方法来

解决这类问题^[1]。由于ALBP是一类NP-hard问题,所以使用启发式算法和人工智能算法可以在较短时间内取得满意解^[2]。

根据优化目标不同,装配线平衡问题主要可以分为两类^[1],第I类:给定生产节拍,最小化需要的工作站数量;第II类:给定工作站数量,最小化生产节拍。文中主要研究第一类装配线平衡问题(ALBP-I)。

文献[3]介绍了几种使用禁忌搜索算法求解ALBP-I问题的算法,但求得的解不够理想。文献[4]提出了ALBP-I的混合遗传算法,采用遗传算法结合局部爬山法,增强了算法的搜索能力。总结相关文献报道,在相关研究中一般都没有给出算法的时间比较,对于装配线平衡问题混合算法的研究还不多。针对目前研究的不足,文中采用禁忌搜索和遗传算法结合的混合算法^[5],对装配线平衡问题进行研究。

收稿日期:2009-09-14;修回日期:2009-12-30

基金项目:国家863计划资助项目(2007AA04Z112);国家自然科学基金资助项目(50875046,60934008)

作者简介:余晓光(1984-),男,江苏南京人,硕士研究生,研究方向为智能优化算法与仿真;严洪森,教授,博士生导师,研究方向为生产计划与调度、知识化制造、并行工程等。

1 装配线平衡问题描述

第一类装配线平衡问题(ALBP-I)可以描述为在装配优先关系的约束下,单一产品的直线型装配线,在给定生产节拍下,最小化工作站数。其数学模型如下:

$$\min m \Leftrightarrow \max \frac{\sum_{i=1}^m (t(S_i)/CT)^2}{m^2} \quad (1)$$

$$\text{s.t. } S_i \cap S_j = \emptyset \quad \text{其中: } i \neq j \quad (2)$$

$$\bigcup_{i=1}^m S_i = E \quad (3)$$

$$t(S_i) \leq CT \quad (4)$$

$$i, j = 1, 2, \dots, m \quad (5)$$

其中: m 为工作站数量, CT 为生产节拍, S_i 为工作站 i 的作业元素集合, $t(S_i)$ 为工作站 i 的作业时间, E 为作业元素全集。

作业的优先关系约束可以用优先矩阵表示,优先矩阵是 $n \times n$ 的方阵,其元素定义如下:

$$p_{ij} = \begin{cases} 1 & i \text{ 作业是 } j \text{ 作业的先导} \\ 0 & i \text{ 作业不是 } j \text{ 作业的先导} \end{cases} \quad (6)$$

2 禁忌搜索遗传算法(TSGA)

遗传算法(Genetic Algorithm, GA)是一种模拟自然界生物进化过程的仿生算法。它通过模仿生物的进化过程来优化解的集合,其基本操作是通过选择、交配和变异搜索解的空间。其对空间的搜索具有隐含并行性,它不同于随机搜索(解空间中的枚举搜索),它的选择机制可以大大减小搜索空间,因而可以在有限时间内得到一个最优解或较优解,并且遗传算法还是一种具有全局优化能力的搜索方法。

禁忌搜索(Tabu Search, TS)算法是局部邻域搜索算法的推广,通过引入禁忌搜索规则,以一定的条件接受劣解,拓展搜索区域,从而跳出局部最优解。

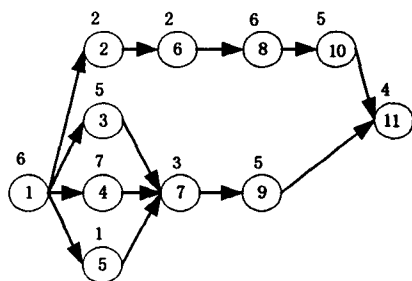
禁忌搜索遗传算法(TSGA)结合了遗传算法和禁忌搜索算法的优点。禁忌搜索遗传算法可以克服遗传算法容易出现早熟和收敛性差的缺点,并提高禁忌搜索算法的效率^[5,6]。

3 算法设计

文中设计的禁忌搜索遗传算法采用非标准操作算子,在算法运行过程中始终保证只对可行作业序列子空间进行搜索,减小搜索范围,提高搜索效率。

3.1 编码

算法中采用基于作业的自然编码方式。将实际的作业元素序号减1可以得到算法中的编码。以Jackson11问题(见图1)为例说明。



作业次序:	1	2	3	4	5	6	7	8	9	10	11
染色体:	0	1	5	4	2	3	7	9	6	8	10
作业序号:	1	2	6	5	3	4	8	10	7	9	11

图1 Jackson 11问题装配顺序图和染色体编码

根据装配优先关系约束生成的装配作业顺序为: 1、2、6、5、3、4、8、10、7、9、11,其染色体如图1所示。

3.2 译码

译码是根据编码将作业元素在生产节拍限制下分配到工作站。以图1中的染色体为例说明。在给定生产节拍12下,工作站1: {任务1、2、6、5,工时11,空闲1}。这时无法容纳下个作业3,开启工作站2。同理译码如下,工作站2: {任务3、4,工时12,空闲0},工作站3: {任务8、10,工时11,空闲1},工作站4: {任务7、9、11,工时12,空闲0};至此工作分配完毕,工作站数为4,总空闲时间为2。

3.3 初始化种群

算法中采用在装配优先关系约束下随机生成初始种群。这种方法的好处是可以在一个较广的范围内进行搜索,避免因为生成规则的限制搜索不到最优解的区域。具体流程描述为:

- 1) 根据装配优先关系初始化优先矩阵;
- 2) 随机选择自由元素,自由元素是优先矩阵中列为0的元素;
- 3) 更新优先矩阵,将选择元素列中不为0元素设置为0;
- 4) 如果作业分配完毕结束,否则转2。

3.4 选择

算法中采用根据个体适应度大小按比例选择的轮盘赌法,选择个体进行复制操作。

3.5 交叉操作

为了保证解的可行性,算法中采用两点法进行交叉。具体步骤为:

- 1) 随机生成交叉起点 $pos1 (1 \leq pos1 \leq n)$;
- 2) 随机生成交叉长度 $len (1 \leq len \leq n - pos1)$;
- 3) 将两染色体基因从 $pos1$ 位置开始的 len 个元素交叉同时保证交叉后仍满足装配优先关系约束,具体操作如图2所示。

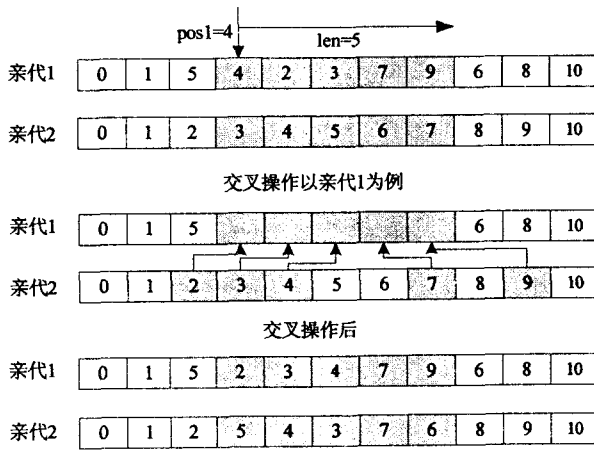


图2 两点法交叉操作

3.6 变异操作

变异操作可以增加解的多样性,增大遗传算法的搜索区域从而跳出局部最优。为了保证解的可行性,算法采用移位插入变异法。随机生成需变异元素的序号后,通过搜索变异元素所在优先矩阵的行列确定可以插入范围,并随机选择一个位置进行插入操作。

3.7 评价函数

ALBP-I模型中的最小化工作站数的性能指标对序列变化不敏感,造成过多的染色体评价价值相同的后果,故文中研究的算法中采用如(7)式的评价函数。求其最大值,该函数可以保证作业向繁忙的工作站集中,从而逐步减少工作站数目。

$$\max f = \frac{\sum_{i=1}^m (t(S_i)/CT)^2}{m^2} \quad (7)$$

3.8 精英保留策略

精英保留策略是保留算法运行过程中最优秀染色体的方法,可以保证在每代个体中都存在以往的最优解,并且最后输出的结果是历史上的最优解。

3.9 邻域生成

ALBP-I的染色体编码的邻域的大小是不固定的,所以禁忌搜索的邻域是在每一次禁忌搜索循环前动态生成的。方法是采用堆栈的方法将搜索到的邻域进行压栈操作,在每次禁忌搜索结束后再对堆栈进行清空,准备进行下一步禁忌搜索邻域生成。另外需要注意的是在元素插入时不插入紧邻的前一个元素之前,因为这与前一个元素的邻域重复。邻域生成的方法是将每一个可能的变异都实现一次。

3.10 禁忌搜索策略

禁忌搜索的策略为:

- 1)生成初始解的邻域;
- 2)判断领域的最优解是否满足藐视规则(Aspiration Rule),是:保存最优解并存入禁忌表,进行步骤1,

否:进行步骤3;

3)根据禁忌表,剔除被禁忌元素;

4)如果所有元素都被禁忌,结束禁忌搜索,否则进入下一步;

5)选择未被禁忌的最优解存入禁忌表,并以此解为初始解,准备进行下次禁忌搜索。

6)判断是否达到禁忌搜索循环步数,否:进行步骤1,是:结束禁忌搜索。

3.11 算法流程图

文中设计的禁忌搜索遗传算法流程如图3所示。

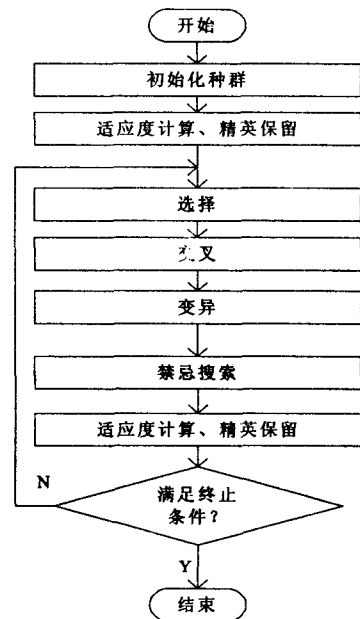


图3 禁忌搜索遗传算法流程

4 仿真实验

文中设计的禁忌搜索遗传算法使用 VC++ 7.0 编写,在 Window XP 平台下使用配置为 CPU: Intel E4500,内存:2GB 的机器运行。使用文献[7]中算例对算法进行测试。下面以 Tonge 问题为例说明。Tonge 问题(见图4)有70个作业元素,最大作业元素时间为156,最小作业元素时间为1,总时间为3510。设置生产节拍为175,算法参数:种群大小为50,演化代数80,交叉概率为0.8,变异概率为0.2,个体参与禁忌搜索的概率为0.1,禁忌搜索步数为50,禁忌表长度为10,求得最优化工作站数为21,总空闲时间为165。具体工作站安排为: $S_1 = \{5, 9, 15\}$, $S_2 = \{10, 16, 30\}$, $S_3 = \{18, 1\}$, $S_4 = \{17, 19, 20, 21\}$, $S_5 = \{2, 3, 4\}$, $S_6 = \{11, 6\}$, $S_7 = \{7, 8, 12\}$, $S_8 = \{24, 70, 14\}$, $S_9 = \{22, 13\}$, $S_{10} = \{23\}$, $S_{11} = \{25, 57\}$, $S_{12} = \{31, 32, 26, 27\}$, $S_{13} = \{33, 34, 29\}$, $S_{14} = \{28, 35, 51, 52, 62\}$, $S_{15} = \{58, 61, 56, 44, 45\}$, $S_{16} = \{59, 63\}$, $S_{17} =$

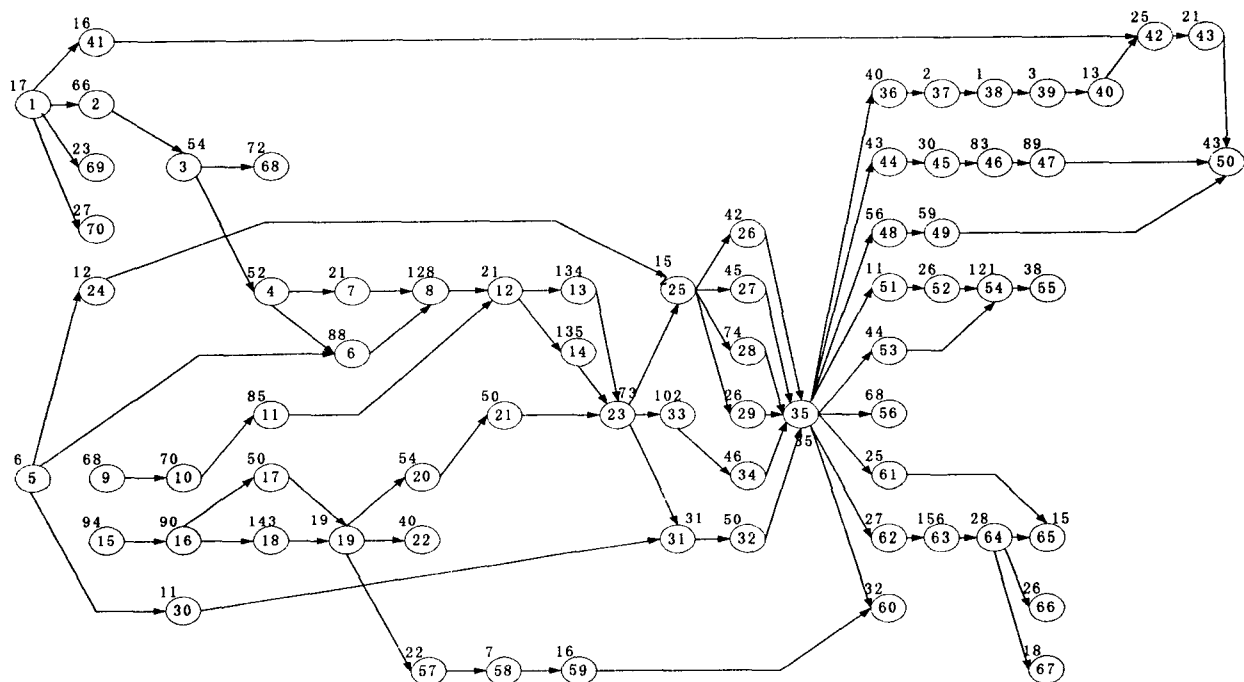


图 4 70 作业的 Tonge 问题工序图

$\{36, 37, 38, 39, 46, 53\}$, $S_{18} = \{69, 41, 48, 64, 67, 60\}$,
 $S_{19} = \{54, 65, 55\}$, $S_{20} = \{40, 47, 66, 42, 43\}$, $S_{21} =$
 $\{68, 49, 50\}$ 。

将遗传算法(禁忌搜索遗传算法减去禁忌搜索部分其余部分相同)与禁忌搜索遗传算法进行结果与时间的比较。设置算法跳出条件为找到最优解或运行 28s 以上。两种算法各运行 50 次得到结果如表 1 所示。 m^* 为最优工作站数; \bar{m} 为 50 次运算的求出的工作站数均值; P 为得到最优解的概率; T 为平均运行时间,单位为秒。从表中可以看出禁忌搜索遗传算法在时间和运算结果上都比遗传算法有比较大的提高。其他问题的禁忌搜索算法运行结果如表 2 所示。

表 1 遗传算法与禁忌搜索算法的比较结果

问题	最优解		GA			TSGA		
	CT	m^*	\bar{m}	P	T	\bar{m}	P	T
Tonge70	175	21	21.66	17/50	21.14	21.12	44/50	11.72

表 2 ALB 问题运行结果

问题	CT	m	CT	m	CT	m	CT	m
Mertens7	6	6	7	5	10	3	15	2
Jackson11	9	6	10	5	12	4	16	3
Buxey29	27	13	28	12	34	10	48	7
Hahn53	2004	8	2806	6	3507	5	4676	4
Tonge70	175	21	364	10	410	9	527	7

5 结束语

装配线平衡问题是实际生产中经常需要求解的问题。合理的平衡装配线作业可以提高生产效率,增加

企业收入。文中提出的禁忌搜索遗传混合算法求解 ALBP-I 问题取得了良好的效果。

对于装配线平衡这种 NP-hard 问题^[2],目前还没有通用的精确算法可以求解,所以目前的观点是没有免费的午餐(No Free Lunch)^[8],对于每个不同问题都有其适合的手段进行解决。在文中算法测试时可以看到在复杂问题时禁忌搜索遗传算法比简单遗传算法搜索效率高,收敛性好,但是在规模较小的问题上混合算法不具有优势。

下一阶段的工作中,可以考虑时间波动对装配时间的影响,并使用仿真软件如 Flexsim 进行装配线 3D 仿真,以更贴近装配线的生产实际。

参考文献:

- [1] Becker C, Scholl A. A survey on problems and methods in generalized assembly line balancing[J]. European Journal of Operational Research, 2006, 168(3): 694 - 715.
- [2] 周亮. 装配线平衡的最优化模型与算法研究[D]. 南京: 南京理工大学, 2005.
- [3] Chiang W C. The application of a tabu search metaheuristic to the assembly line balancing problem[J]. Annals of Operations Research, 1998, 77(1): 209 - 227.
- [4] Goncalves J F, Almeida J R. A hybrid genetic algorithm for assembly line balancing[J]. Journal of Heuristics, 2002, 8(6): 629 - 642.
- [5] 张平, 严洪森, 余晓光. 基于混合算法的知识网运算表达式优化[J]. 计算机技术与发展, 2009, 19(3): 32 - 35.

(下转第 12 页)

edTreeMiner 好很多。

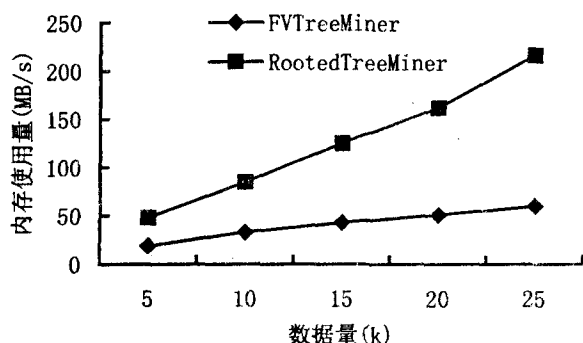


图 4 不同数据量的内存使用量

图 5 和图 6 是在支持度介于 0.5% 至 5% 时,运行时间和空间消耗的实验结果。FVTreeMiner 由于使用了频繁延伸点的概念,避免产生不必要的候选者,减少了候选者的测试,所以在空间消耗和时间消耗量上都明显少于 RootedTreeMiner。

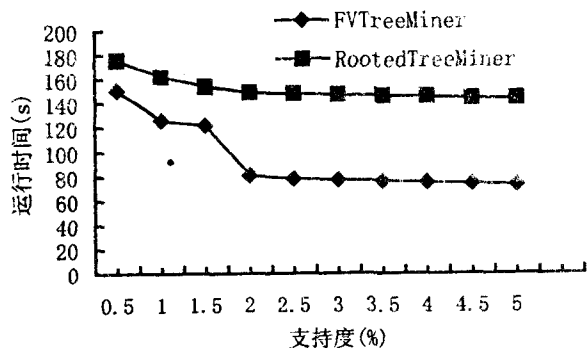


图 5 不同支持度的执行时间

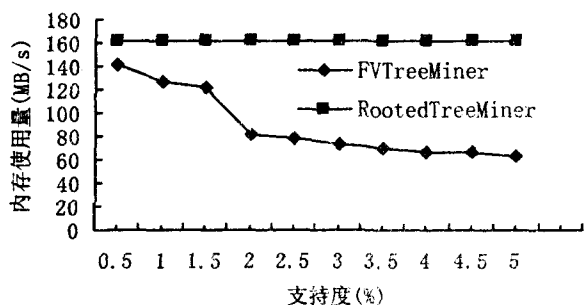


图 6 不同支持度的内存使用量

4 结束语

在庞大的树资料库中去匹配一个候选模式复杂性很高,所以为了减少匹配时间和所需的内存空间,文中提出树投影资料库的概念;在频繁子树的延伸上,采用

最右延伸的方式,同时利用延伸点频繁剔除掉部分多余的可延伸点,避免产生不必要的候选模式,减少了匹配模式和计算支持度的时间;子树频繁度的提出也有利于支持度的计算。

通过实验,可以看到,较 RootedTreeMiner,文中的这三个改进对频繁树挖掘算法的效率有明显的提高。另外,对于新的子树有部分可延伸点在前一个模式中已经找过了,所以可继承这些点,以提高算法的效率。

参考文献:

- [1] Chi Y, Yang Y, Muntz R R. HybridTreeMiner: An Efficient Algorithm for Mining Frequent Rooted Trees and Free Trees Using Canonical Forms[C]//In proceedings of the 16th International Conference on Scientific and Statistical Database Management (SSDEM'04). Washington: IEEE Computer Society, 2004.
- [2] Chi Y, Yang Y, Muntz R R. Canonical Forms for Labeled Trees and Their Applications in Frequent Subtree Mining[J]. Journal of Knowledge and Information Systems (KAIS), 2005, 8(2): 203-234.
- [3] Huang K Y, Chang C H, Lin K Z. PROWL: An efficient frequent continuity mining algorithm on event sequences[C]//In proceedings of 6th International Conference on Data Warehousing and Knowledge Discovery (DaWak). Washington: [s. n.], 2004.
- [4] Agrawal R, Srikant R. Fast algorithms for mining association rules[C]//In proceedings of 1994 International Conference. Very Large Data Bases (VLDB'94). New York: [s. n.], 1994: 487-499.
- [5] 潘 锦. Chopper: 一个高效的有序标号树频繁结构的挖掘算法[C]//第 20 届全国数据库年会 (NDBC). 长沙: [出版者不详], 2003: 303-308.
- [6] 杨 沛, 郑启伦, 彭 宏, 等. PFTM: 一种基于投影的频繁子树挖掘算法[J]. 计算机科学, 2005, 32: 206-209.
- [7] 王新宇, 杜孝平, 谢昆清. FP-growth 算法的实现方法研究[J]. 计算机工程与应用, 2004(9): 174-176.
- [8] 吉根林, 韦素云, 鲍培明. 一种基于 DOM 树的 XML 数据频繁模式挖掘算法[J]. 南京航空航天大学学报, 2006, 38(2): 206-211.
- [9] Zaki M J. Fast vertical mining using diffsets, TR01-1[R]. New York: Rensselaer Polytechnic Institute, 2001.
- [10] 朱永泰, 王 晨, 洪铭胜, 等. ESPM-频繁子树挖掘算法[J]. 计算机研究与发展, 2004, 41(10): 1720-1727.

(上接第 8 页)

- [6] 董建华, 肖田元, 赵银燕. 遗传禁忌搜索算法在混流装配线排序中的应用[J]. 工业工程与管理, 2003, 8(2): 14-17.
- [7] Scholl A. Data of assembly line balancing problems[EB/OL]. 1993. <http://www.assembly-line-balancing.de>.

- [8] Wolpert D H, Macready W G. No free lunch theorems for optimization[J]. IEEE Transactions on Evolutionary Computation, 1997, 1(1): 67-82.