文章编号: 1001-0920(2008)10-1098-05

一种求解混合整数规划的混合进化算法

李 宏[a,b], 焦永昌[b], 张 莉[a]
(西安电子科技大学 a. 理学院, b. 电子工程学院, 西安 710071)

摘 要: 提出一种基于正交试验设计的混合进化算法,用于求解混合整数规划问题. 进化算法中采用一种混合启发式的变异算子,将正交试验设计作为杂交算子. 为了增加种群的多样性,引入一种迁移算子. 仿真实验结果表明,与已有的一些算法相比,所提出的求解混合整数规划的混合进化算法能快速收敛到问题的最优解,并且算法的计算量小,解的精度高.

关键词: 混合整数规划; 正交试验设计; 混合进化算法
中图分类号: O221.4 **文献标识码**: A

Hybrid evolutionary algorithm for mixed-integer programming problems

LI Hong[a,b], *JIAO Yong-chang*[b], *ZHANG Li*[a]
(a. School of Science, b. School of Electronic Engineering, Xidian University, Xi'an 710071, China. Correspondent: LI Hong, E-mail: lihong@mail.xidian.edu.cn)

Abstract: A hybrid evolutionary algorithm based on orthogonal experimental design is developed to deal with the mixed-integer programming problems. This hybrid evolutionary algorithm contains the heuristic mutation operator, the crossover operator as which the orthogonal experimental design serves, and the migration operator to keep the population's diversity. Numerical results show that, compared with other available algorithms, the proposed algorithm can converge at optimal solutions of all test problems with higher speed, and it has a smaller number of function evaluations and yields more accurate optimal function values.

Key words: Mixed-integer programming; Orthogonal experimental design; Hybrid evolutionary algorithm

1 引 言

混合整数规划是指一类同时包含整数变量和连续变量的优化问题,其一般数学表达如下:

$$\begin{aligned}&\min\ f(x,y),\\&\text{s.t.}\ g_i(x,y)\le 0,\ i\in I=\{1,2,\cdots,K\},\\&\qquad L_1\le x\le U_1,\ L_2\le y\le U_2,\\&\qquad x\in R^n,\ y\in Z^m.\end{aligned}\tag{1}$$

其中: x 是 n 维连续变量, y 是 m 维整数变量. 变量的上界和下界为

$$L_1=(l_1^1,l_2^1,\cdots,l_n^1),\ U_1=(u_1^1,u_2^1,\cdots,u_n^1);$$
$$L_2=(l_1^2,l_2^2,\cdots,l_m^2),\ U_2=(u_1^2,u_2^2,\cdots,u_m^2).$$

定义 1 如果 $x\in R^n$, $y\in Z^m$, 且满足约束条件 $g_i(x,y)\le 0$, $i\in I$, $L_1\le x\le U_1$, $L_2\le y\le U_2$, 则称 (x,y) 是可行的.

整数规划[1,2]和混合整数规划[3,4]是 NP-完全问题,其求解算法具有指数复杂性. 基于梯度的优化方法仅能处理特殊形式的问题,如需要问题具有连续性、可微性和凸性等要求,或问题需要有特殊的数学结构. 大量使用的随机算法(包括进化算法)是一种自适应随机搜索算法,已成功地应用于求解混合整数规划问题[3,4].

2 混合进化算法

2.1 初始种群

本文采用实数和整数的混合编码,将向量

$$(x,y)=(x_1,x_2,\cdots,x_n,y_1,y_2,\cdots,y_m)\tag{2}$$

作为一个染色体,表示问题(1)的一个解. 其中: $x_i\in R$, $y_j\in Z$, $i=1,2,\cdots,n$, $j=1,2,\cdots,m$. 由如下算法随机产生 M 个染色体(M 称为种群规模),构成

收稿日期: 2007-06-22; **修回日期**: 2007-09-21.
基金项目: 国家自然科学基金项目(60171045, 60374063).
作者简介: 李宏(1972—),男,陕西丹凤人,讲师,硕士,从事进化算法、智能信息处理的研究; 焦永昌(1964—),男,山西芮城人,教授,博士,从事优化算法及应用、天线设计与测量等研究.

一个初始种群.

算法 1

Step1: 设 $\alpha = \mathrm{diag}(\alpha_1, \alpha_2, \cdots, \alpha_n)$, $\beta = \mathrm{diag}(\beta_1, \beta_2, \cdots, \beta_m)$, 对角线元素 $\alpha_1, \alpha_2, \cdots, \alpha_n, \beta_1, \beta_2, \cdots, \beta_m$ 都是区间 $(0,1)$ 中的随机数.

Step2: 令

$$\begin{cases} x = L_1 + (U_1 - L_1)\alpha, \\ y = \mathrm{INT}[L_2 + (U_2 - L_2)\beta], \end{cases} \tag{3}$$

其中 $\mathrm{INT}[a]$ 表示四舍五入到最接近实数向量 a 的整数向量(下同).

Step3: 检查产生的染色体 (x, y) 的可行性. 如果它是不可行的,则抛弃这个染色体;否则,保存它.

Step4: 重复以上操作,直到产生 M 个初始可行的染色体.

2.2 适应度函数

所讨论的问题(1) 是最小化问题,因此设定:适应度越小其结果越好. 设适应度函数

$$F(x,y) = f(x,y) + \Theta \sum_{i=1}^{K} \max\{0, g_i(x,y)\}, \tag{4}$$

其中 Θ 是罚因子,它是合适的正数.

2.3 变异算子

设到目前为止种群中的最好染色体为 $(x,y)^b = (x^b, y^b)$. 在当前种群中,以变异概率 p_m 随机选择一个染色体,记为 $(x,y)^l = (x^l, y^l)$. 如果满足

$$\|(x,y)^b - (x,y)^l\| \geq \varepsilon_1, \ \varepsilon_1 = 10^{-4}, \tag{5}$$

则按以下方式产生一个变异染色体:

$$(x,y)^m = (x^m, y^m). \tag{6}$$

其中

$$\begin{cases} x^m = x^b + (x^b - x^l)\lambda, \\ y^m = \mathrm{INT}[y^b + (y^b - y^l)\chi], \end{cases} \tag{7}$$

λ 和 χ 分别是 n 维和 m 维对角矩阵,对角线上的元素都是 $(0,1)$ 区间的随机数.

若不满足式(5),则在当前种群中随机选择两个染色体 $(x,y)^p$ 和 $(x,y)^q$,按式(6) 的方式产生一个变异染色体,其中

$$\begin{cases} x^m = x^l + (x^b - x^l)\lambda + (x^p - x^q)\mu, \\ y^m = \mathrm{INT}[y^l + (y^b - y^l)\chi + (y^p - y^q)\kappa], \end{cases} \tag{8}$$

λ 和 μ 是 n 维对角矩阵,χ 和 κ 是 m 维对角矩阵,对角线上的元素都是 $[0,1]$ 区间的随机数.

2.4 杂交算子

正交试验设计是一种统计试验设计方法,它是从全面试验中选出部分有代表性的点进行试验,这些代表点具有均匀和整体的特点. 正交试验设计是部分因子设计的主要方法,具有很高的效率[5-10].

本文采用两水平正交表,设有 $N-1$ 个因素,且每个因素有 2 个水平,记为 $L_N(2^{N-1})$. 其中:N 代表试验次数,$N-1$ 是因素个数. 如果 $n+m = N-1$,则直接利用标准正交表 $L_N(2^{N-1})$;如果 $n+m < N-1$,则利用正交表 $L_N(2^{n+m})$. 本文使用的正交表取自文献[5],文献[6] 提供了产生各种正交表的算法.

以杂交概率 p_c 从变异后代中随机选择两个染色体 $c_1 = (x^1, y^1)$ 和 $c_2 = (x^2, y^2)$,每个染色体有 $n+m$ 个分量. 按以下方式作凸组合,产生染色体 c_1' 和 c_2':

$$c_1' = (x^1 Y_1 + x^2(I_1 - Y_1), \quad \mathrm{INT}[y^1 T_2 + y^2(I_2 - T_2)]); \tag{9}$$

$$c_2' = (x^1(I_1 - Y_1) + x^2 Y_1, \quad \mathrm{INT}[y^1(I_2 - T_2) + y^2 T_2]). \tag{10}$$

其中:I_1 和 I_2 分别是 n 维和 m 维单位矩阵,Y_1 和 T_2 分别是 n 维和 m 维对角矩阵,对角线上的元素都是 $[0,1]$ 区间的随机数.

把染色体 c_1 和 c_2 排成一个 2 行 $n+m$ 列的表,c_1 所在的第 1 行看作水平 1,c_2 所在的第 2 行看作水平 2. 每一列代表一个因素,每个因素有 2 个水平. 参照正交表 $L_N(2^{n+m})$,如果正交表中的元素 L_{ij} (第 i 行第 j 列) 为 1,则选择第 j 个因素对应的水平 1 的值,该位置上的 1 用水平 1 对应的值来替换;如果正交表中的元素 L_{ij} 为 2,则选择第 j 个因素对应的水平 2 的值,替换该位置上的 2. 从而得到 N 个组合(替换后的正交表的每一行是一个组合),每个组合就是一个子代染色体.

完成上述操作后,还需对其进行因素分析[7,8]. 因素分析用来评估每个因素的不同水平对适应度值的影响,通过因素分析可找到更优的染色体. 本文采用文献[7]中的因素分析法,设 F_i 表示第 i 次试验中因素组合的适应度值,$i = 1,2,\cdots,N$. 把因素 j 的第 k 个水平的影响效果记为 E_{jk},则有

$$E_{jk} = \sum_{i=1}^{N} F_i d_i, \ k = 1,2, \quad j = 1,2,\cdots,n+m. \tag{11}$$

在第 i 次试验中,如果因素 j 的水平是 k,则 $d_i = 1$;否则,$d_i = 0$. 问题(1) 是最小化问题, 如果 $E_{j1} < E_{j2}$,说明因素 j 的水平 1 好于水平 2,则因素 j 选择水平 1;否则,选择水平 2. 如果 $E_{j1} = E_{j2}$,则说明水平 1 和水平 2 相当. 在所有因素都选择了较好的水平之后,便可得到一个新的优于父代的染色体.

2.5 迁移算子

为了提高种群的多样性, 防止出现早熟收敛,本文采用一种类似于文献[4] 中的迁移算子. 设到

目前为止种群中的最好染色体为 $(x, y)^b$,设 p_{mi} 是迁移概率,在区间(0,1)上产生一个随机数 r,如果 $r < p_{mi}$,则随机选择染色体 $(x, y)^b$ 的两个基因 x_i^b 和 y_j^b,按下面的方式产生两个新基因,分别记为 $\overline{x_i^b}$ 和 $\overline{y_j^b}$:

$$\overline{x_i^b} = \begin{cases} x_i^b + \rho_1 (l_i^1 - x_i^b), \ \gamma_1 < \dfrac{y_i^b - l_i^1}{u_i^1 - l_i^1}; \\ x_i^b + \rho_1 (u_i^1 - y_i^b), \ 否则. \end{cases} \tag{12}$$

其中:ρ_1 和 γ_1 都是区间[0,1]上的随机数,$i \in \{1,2,\cdots,n\}$,$j \in \{1,2,\cdots,m\}$.

$$\overline{y_j^b} = \begin{cases} y_j^b + \mathrm{INT}[\rho_2 (l_j^2 - y_j^b)], \ \gamma_2 < \dfrac{y_j^b - l_j^2}{u_j^2 - l_j^2}; \\ y_j^b + \mathrm{INT}[\rho_2 (u_j^2 - y_j^b)], \ 否则. \end{cases} \tag{13}$$

其中 ρ_2 和 γ_2 都是区间[0,1]上的随机数.

用这两个新基因 $\overline{x_i^b}$ 和 $\overline{y_j^b}$ 分别代替原来的两个基因 x_i^b 和 y_j^b,其余基因不变.这样便产生一个不同于 $(x, y)^b$ 的新染色体.

2.6 选择算子

将父代及其所有产生的后代染色体的适应度值从小到大排序,选取前 M 个染色体作为下一代种群,并保留每一代的最好染色体.

2.7 终止规则

如果算法执行了最大进化代数 NUG,则停止,输出保留的最好染色体,作为近似全局最优解.

3 混合进化算法的步骤

Step1:参数设置.种群规模 M,变异概率 p_m,杂交概率 p_c,迁移概率 p_{mi},罚因子 Θ,最大进化代数 NUG.

Step2:初始化.用算法 1 产生初始种群.

Step3:变异.

1) 求出当前种群中的最好染色体 $(x, y)^b$.

2) 在当前种群中,以变异概率 p_m 随机选择一个染色体 $(x, y)^l$.

3) 如果满足式(5),则按式(6)和(7)产生一个变异染色体,否则,按式(6)和(8)产生一个变异染色体,并计算它的适应度值.

4) 重复 Step3 之 2) 和 3) M 次,产生 M_m 个变异后代,构成临时种群.

Step4:杂交.

1) 选择适当的正交表 $L_N(2^{N-1})$.

2) 在临时种群中,以概率 p_c 随机选择两个变异染色体,作凸组合并进行正交试验,产生 N 个杂交后代,并计算它们的适应度值.

3) 计算各因素的影响效果,确定最优水平.

4) 根据选定的各因素的最优水平,生成一个新的更好的染色体,并计算它的适应度值.

5) 重复 Step4 之 2)～4) M_m 次,直到生成 M_c 个染色体,构成中间种群.

Step5:迁移:

1) 求出到目前为止最好的染色体.

2) 产生一个随机数 $r \in (0,1)$,如果 $r < p_{mi}$,则转 Step5 之 3);否则,转 Step5 之 4).

3) 随机选取最好染色体的两个基因,按式(12)和(13)产生一个新染色体,并计算它的适应度值.

4) 重复 Step5 之 2) 和 3) M_c 次,直到生成 M_{mi} 个染色体.

Step6:选择.按选择策略选出较好的 M 个染色体,构成下一代种群,并保留最好染色体.

Step7:判断.停机条件是否满足,满足便输出结果,否则转 Step3.

4 数值仿真及结果比较

为了评估本文提出的算法(简称 HEA)的性能,作者测试了一些混合整数规划问题.以下算例均取自文献[3,4].

算例 1

$$\min f(x, y) = 2x + y,$$
$$\text{s.t.} \ 1.25 - x^2 - y \le 0, \ x + y - 1.6 \le 0,$$
$$0 \le x \le 1.6, \ y \in \{0,1\}.$$

全局最优解是 $(x, y) = (0.5,1)$,最优值是 2.

算例 2

$$\min f(x, y) = 2x_1 + x_2 - y,$$
$$\text{s.t.} \ x_1 - 2\exp(-x_2) = 0,$$
$$-x_1 + x_2 + y \le 0,$$
$$0.5 \le x_1 \le 1.4, \ y \in \{0,1\}.$$

全局最优解是 $(x_1, x_2, y) = (1.375, 0.375, 1)$,最优值是 2.124.

算例 3

$$\min f(x, y) = -0.7y + 5(x_1 - 0.5)^2 + 0.8,$$
$$\text{s.t.} \ -\exp(x_1 - 0.2) - x_2 \le 0,$$
$$x_1 - 1.2y - 0.2 \le 0, \ x_2 + 1.1y + 1 \le 0,$$
$$-2.22554 \le x_2 \le -1,$$
$$0.2 \le x_1 \le 1, \ y \in \{0,1\}.$$

全局最优解是 $(x_1, x_2, y) = (0.94194, -2.1, 1)$,最优值是 1.07654.

算例 4

$$\min f(x, y_1, y_2, v_1, v_2) = 7.5y_1 + 5.5y_2 + 7v_1 + 6v_2 + 5x,$$
$$\text{s.t.} \ y_1 + y_2 = 1, \ z_1 + z_2 = 10,$$
$$z_1 = 0.9[1 - \exp(-0.5v_1)]x_1,$$
$$z_2 = 0.8[1 - \exp(-0.4v_2)]x_2,$$

$x_1 + x_2 = x,\ z_1 y_1 + z_2 y_2 = 10,\ v_1 \le 10 y_1,$
$v_2 \le 10 y_2,\ x_1 \le 20 y_1,\ x_2 \le 20 y_2,$
$x_1, x_2, z_1, z_2, v_1, v_2 \ge 0,\ y_1, y_2 \in \{0,1\}.$

全局最优解是 $(x, y_1, y_2, v_1, v_2) = (13.362272, 1, 0, 3.514237, 0)$,**最优值是** 99.245209.

算例 5

$$\min f(x,y) = (y_1 - 1)^2 + (y_2 - 1)^2 + (y_3 - 1)^2 - \ln(1 + y_4) + (x_1 - 1)^2 + (x_2 - 2)^2 + (x_3 - 3)^2,$$

s.t.

$y_1 + y_2 + y_3 + x_1 + x_2 + x_3 - 5 \le 0,$
$y_3^2 + x_1^2 + x_2^2 + x_3^2 - 5.5 \le 0,$
$y_1 + x_1 - 1.2 \le 0,\ y_2 + x_2 - 1.8 \le 0,$
$y_3 + x_3 - 2.5 \le 0,\ y_4 + x_1 - 1.2 \le 0,$
$y_2^2 + x_2^2 - 1.64 \le 0,\ y_3^2 + x_3^2 - 4.25 \le 0,$
$y_2^2 + x_3^2 - 4.64 \le 0,\ x_1, x_2, x_3 \ge 0,$
$y_1, y_2, y_3, y_4 \in \{0,1\}.$

全局最优解是 $(x_1, x_2, x_3, y_1, y_2, y_3, y_4) = (0.2, 1.280624, 1.954483, 1, 0, 0, 1)$, **最优值是** 3.557463.

算例 6

$$\min f(x,y) = -5.357854 x_1^2 - 0.835689 y_1 x_3 - 37.29329 y_1 + 40792.141,$$

s.t.

$a_1 + a_2 y_2 x_3 + a_3 y_1 x_2 - a_4 x_1 x_3 - 92 \le 0,$
$a_5 + a_6 y_2 x_3 + a_7 y_1 y_2 + a_8 x_1^2 - 110 \le 0,$
$a_9 + a_{10} x_1 x_3 + a_{11} y_1 x_1 + a_{12} x_1 x_2 - 25 \le 0,$
$27 \le x_1, x_2, x_3 \le 45,\ y_1 \in \{78, 79, \cdots, 102\},$
$y_2 \in \{33, 34, \cdots, 45\}.$

全局最优解是 x_2, y_2 **任取**,$(x_1, x_3, y_1) = (27, 27, 78)$,**最优值是** 32217.4. **约束系数** $a_1, a_2, \cdots, a_{12}$ **的值详见文献**[3].

本文算法的参数设置为:种群规模 $M = 30$,**变异概率** $p_m = 0.3$,**杂交概率** $p_c = 0.8$,**迁移概率** $p_{mi} = 0.2$,**最大进化代数** NUG = 100,**罚因子** $\Theta = 10000$. **所用的两水平正交表选自文献**[5]**中附录,分别为**:$L_4(2^2)$,$L_4(2^3)$;$L_8(2^4)$,$L_8(2^5)$,$L_8(2^7)$.

每个问题独立运行 10 **次,记录** 10 **次所得的最优解、最优值和成功率**(10 **次运行中找到最优解的百分比**),**找出最优解的平均** CPU **时间和适应度函数的平均计算次数**. HEA **的统计结果如表** 1 **所示**. **将** HEA **的成功率和计算量与文献**[3]**中的** 5 **种算法**(**表** 2 **前** 5 **列**) **及文献**[4]**中算法** MIHDE(**表** 2 **第** 6 **列**) **进行比较,其结果如表** 2 **所示**. **表中**:# F **表示适应度函数的平均计算次数**,C% **表示成功率**.

由表 1 **可以看出,本文提出的** HEA **能以较大的概率快速找到所有问题的最优解,而且解的精度很高. 由表** 2 **可以看出,除了算法** $(\mu + \lambda)$ - ES,

表 1　HEA 的数值结果

问题	最优值	最　优　解	成功率 C/ %	平均 CPU 时间 / s	适应度函数的平均计算次数# F
1	2	(0.5, 1)	100	0.0470	780
2	2.124472	(1.374826, 0.374820, 1)	100	0.0766	1612
3	1.076543	(0.941937, - 2.1, 1)	100	0.5017	5250
4	99.239101	(13.427941, 1, 0, 3.514116, 0)	90	0.9249	8175
5	3.557464	(0.2, 1.280623, 1.954482, 1, 0, 0, 1)	80	1.1970	9008
6	32217.43	(27, *, 27, 78, 39)	100	0.3922	2580

*: 在[27,45]之间任取.

表 2　HEA 与文献[1,2]中其他算法的比较

问题	GA-Pen		GA-Deb		$(\mu+\lambda)$-ES		M-SIMPSA		M-SIMPSA-Pen		MIHDE (N_p = 3)		HEA	
	# F	C/ %	# F	C/ %	# F	C/ %	# F	C/ %	# F	C/ %	# F	C/ %	# F	C/ %
1	6787	100	6191	100	1518	100	607	99	16282	100	13104	100	780	100
2	13939	100	15298	100	2255	100	10582	83	14440	100	29166	100	1612	100
3	107046	90	110233	90	1749	100	b	0	38042	100	28455	100	5250	100
4	22489	100	23730	80	b	0	14738	100	42295	100	60950	100	8175	90
5	102778	60	34410	90	6710	100	22309	60[a]	63751	97[a]	12375	100	9008	80
6	37167	100	35255	100	2536	100	27410	87	33956	95	938	100	2580	100

a: 收敛到非最优解; b: 中断算法的执行.

M-SIMPSA 和 MIHDE 中的个别算例外, HEA 对适应度函数的平均计算次数明显少于其他算法.

5 结 论

本文把统计试验设计的方法 ——正交设计作为交叉算子,给出了一种混合启发式的变异算子. 为了提高种群的多样性,引入一种迁移算子,提出一种混合进化算法. 本文算法不仅能求解混合整数规划,而且能求解纯整数规划. 数值实验结果表明,所提出的混合进化算法是快速而有效的,同已有的其他算法的数值结果相比,本文算法计算量小,求得解的精度高.

参考文献(References)

[1] 孟志青, 胡奇英, 杨晓琪. 一种求解整数规划与混合整数规划非线性罚函数方法[J]. 控制与决策, 2002, 17(3): 310-314.
(Meng Z Q, Hu Q Y, Yang X Q. A method of non-linear penalty function for solving integer programming and mixed integer programming [J]. Control and Decision, 2002, 17(3): 310-314.)

[2] 肖建, 张志宏. 一种求非线性整数规划全局最小解的算法[J]. 石家庄学院学报, 2006, 8(6): 49-53.
(Xiao J, Zhang Z H. An algorithm for solving the global optimization of nonlinear integer programming [J]. J of Shijiazhuang University, 2006, 8(6): 49-53.)

[3] Lino Costa, Pedro Oliveira. Evolutionary algorithms approach to the solution of mixed integer non-linear programming problems [J]. Computers and Chemical Engineering, 2001, 25(2): 257-266.

[4] Lin Y C, Hwang K S. A mixed-coding scheme of evolutionary algorithms to solve mixed-integer nonlinear programming problems[J]. Computers and Mathematics with Applications, 2004, 47(8): 1295-1307.

[5] 方开泰, 马长兴. 正交与均匀试验设计[M]. 北京: 科学出版社, 2001.
(Fang K T, Ma C X. Orthogonal and uniform experimental design [M]. Beijing: Science Press, 2001.)

[6] Leung Y W, Wang Y. An orthogonal genetic algorithm with quantization for global numerical optimization[J]. IEEE Trans on Evolutionary Computation, 2001, 5(1): 41-53.

[7] Ho S Y, Shu L S, Chen J H. Intelligent evolutionary algorithms for large parameter optimization problems [J]. IEEE Trans on Evolutionary Computation, 2004, 8(6): 522-540.

[8] Tsai J T, Liu T K, Chou J H. Hybrid taguchi-genetic algorithm for global numerical optimization[J]. IEEE Trans on Evolutionary Computation, 2004, 8(4): 365-377.

[9] Zhang Q, Leung Y W. An orthogonal genetic algorithm for multimedia multicast routing[J]. IEEE Trans on Evolutionary Computation, 1999, 3(1): 53-62.

[10] Li H, Jiao Y C, Zhang L, et al. Genetic algorithm based on the orthogonal design for multidimensional knapsack problems [C]. Proc of 2nd Int Conf on Advances in Natural Computation. Birlin: Springer-Verlag, 2006, 1: 696-705.

(上接第 1097 页)

[7] Kennedy J, Eberhart R. Particle swarm optimization [C]. IEEE Int Conf on Neural Networks. Perth, 1995: 1942-1948.

[8] 夏蔚军, 吴智铭. 基于混合微粒群优化的多目标柔性 Job-shop 调度[J]. 控制与决策, 2005, 20(2): 137-141.
(Xia W J, Wu Z M. Hybrid particle swarm optimization approach for multi-objective flexible job-shop scheduling problems[J], Control and Decision, 2005, 20 (2): 137-141.)

[9] 彭传勇, 高亮, 邵新宇, 等. 求解作业车间调度问题的广义粒子群优化算法[J]. 计算机集成制造系统, 2006, 12(6): 911-917.
(Peng C Y, GAO L, Shao X Y, et al. General particle swarm optimization algorithm for job-shop scheduling problem [J]. Computer Integrated Manufacturing Systems, 2006, 12(6): 911-917).

[10] Fatih Tasgetiren M, Yun-Chia Liang, Mehmet Sevkli, et al. A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem [J]. European J of Operational Research, 2007, 177(3): 1930-1947.

[11] Jean-Marie Proth. Scheduling: New trends in industrial environment[J]. Annual Reviews in Control, 2007, 31(1): 157-166.