

# 一种求解混合离散优化问题的禁忌微粒群算法\*

张星会,白富生

(重庆师范大学 数学学院,重庆 400047)

摘要:微粒群优化(PSO)算法主要用于求解全局连续变量优化问题。利用罚函数处理离散变量,将混合离散优化问题  $\min f(x)$  s. t.  $g_k(x) \leq 0 \quad k=1, 2, \dots, l$ ,  $x_i^l \leq x_i \leq x_i^u \quad i=1, 2, \dots, m$ ,  $x_{m+j} \in D_j, D_j = (d_{j,1}, d_{j,2}, \dots, d_{j,q_j}), j=1, 2, \dots, n$  转化为连续变量优化问题  $\min F(x)$  s. t.  $x_i^l \leq x_i \leq x_i^u \quad i=1, 2, \dots, m$ ,  $d_{j,1} \leq x_{m+j} \leq d_{j,q_j} \quad j=1, 2, \dots, n$ 。为了解决标准 PSO 可能陷入局部最优解而存在早熟收敛的问题,本文构造微粒的邻域结构,利用禁忌搜索(TS)算法具有较强的“爬山”能力的特点,设计了一种兼具搜索惯性又能在搜索时跳出局部最优解转向解空间的其它区域的禁忌微粒群算法(TS-PSO)。求解 Rosenbrock's 测试函数和压力管设计问题的数值实验表明,该算法能较好地跳出局部最优,获得全局最优解。

关键词:离散变量;罚函数;微粒群优化;禁忌搜索

中图分类号:O221.4

文献标识码:A

文章编号:1672-6693(2011)02-0005-06

在工程、军事、国防等很多领域,经常遇到设计变量只能取离散值的情况。求解包含离散变量优化问题的典型方法包括分支定界法和对偶问题转换。也有方法把离散变量转化成连续变量,通过数学规划找到连续最优解,然后在连续最优解附近取离散值,或者直接应用整数规划获得最优解。文献[1]综述了处理离散变量的典型方法。

PSO 算法主要是针对连续变量进行搜索,寻找全局最优解的一个群智能方法<sup>[2]</sup>。目前,已有研究把 PSO 算法应用到包括离散变量和连续变量的混合变量问题中<sup>[3-4]</sup>。这些研究直接将离散变量作为连续变量进行搜索,然后在最优解附近选定离散值或者通过舍入法获得离散值。或者使用罚函数将混合变量问题的离散变量转化为连续变量,并在此基础上构造可微的增广目标函数,使得增广目标函数在原问题可行域范围内的离散点的值与原目标函数的值相等。然后通过求解包含增广目标函数的连续优化问题,从而获得原混合变量问题的最优解。

为了有效地解决大量的包含离散变量的结构优化设计问题,本文利用罚函数处理离散变量,然后将禁忌搜索与微粒群算法相结合设计了求解混合离散优化问题的禁忌微粒群算法。目前,将禁忌搜索和微粒群算法相结合,用于求解优化问题的研究还比较少。文献[5]用结合禁忌搜索和微粒群算法的混

合算法求解连续变量优化问题。文献[6]用结合禁忌搜索和微粒群算法的混合算法求解离散变量优化问题。先用罚函数处理离散变量,再用结合禁忌搜索和微粒群算法的混合算法来求解混合离散优化问题的文献还尚未见到。

## 1 混合离散优化问题

### 1.1 混合离散优化问题表述

本文研究的混合离散优化问题可表述为

$$\begin{aligned} \min f(x) \\ \text{s. t. } g_k(x) \leq 0 \quad k=1, 2, \dots, l \quad (1) \\ x_i^l \leq x_i \leq x_i^u \quad i=1, 2, \dots, m \\ x_{m+j} \in D_j, D_j = (d_{j,1}, d_{j,2}, \dots, d_{j,q_j}), j=1, 2, \dots, n \quad (2) \end{aligned}$$

其中  $x$  是由连续变量和离散变量构成的混合变量向量。 $x_i$  和  $x_{m+j}$  表示变量向量元素  $m$  和  $n$  分别表示连续变量和离散变量的个数,  $x_i^l$  和  $x_i^u$  分别表示连续变量的下界和上界。(2)式中  $D_j$  是第  $j$  个离散变量的离散值集,  $q_j$  表示离散值的个数,  $d_{j,q_j}$  是第  $j$  个离散变量的第  $q_j$  个离散值。一般来说,对于每个离散变量可利用的离散值的个数  $q_j$  可能是不同的。 $f(x)$  是所求问题的目标函数,  $g_k(x)$  是行为约束,  $l$  表示行为约束的个数。

\* 收稿日期:2010-07-11

资助项目:国家自然科学基金(No.10626058)

作者简介:张星会,女,硕士研究生,研究方向为最优化理论与算法;通讯作者:白富生, E-mail: fsbai@cqu.edu.cn

## 1.2 离散变量的处理

由于PSO是一个主要针对连续变量问题寻优的方法,所以在本文中采用(3)式,即由Shin D K<sup>[4]</sup>提出的罚函数来处理离散变量,使所有的离散变量都可看做连续变量去寻优。

$$\phi(x) = \sum_{j=1}^n \frac{1}{2} \left[ \sin \frac{2\pi \{x_{m+j}^C - 0.25(d_{j,q+1} + 3d_{j,q})\}}{d_{j,q+1} - d_{j,q}} + 1 \right] \quad (3)$$

其中 $d_{j,q}$ 和 $d_{j,q+1}$ 表示离散变量 $x_{m+j}^C$ 是 $d_{j,q}$ 和 $d_{j,q+1}$ 之间的连续变量。由罚函数(3)式和行为约束(1)式构造增广目标函数

$$F(x) = f(x) + s\phi(x) + r \sum_{k=1}^l \max[0, g_k(x)] \quad (4)$$

此处的 $s$ 和 $r$ 分别是(3)式和行为约束(1)式的罚参数。当 $s$ 充分大,离散值都是等步长移动时, $F(x)$ 波谷之间距离非常接近,近似等于离散值的步长,且在各个满足行为约束的离散点 $d_{j,q}$ , $F(x)$ 的函数值和原目标函数 $f(x)$ 的函数值相等。通过(4)式这种转换,所有变量都可看作连续变量,从而求解混合离散优化问题转化为求解下列问题。

$$\min F(x) \quad (5)$$

$$\text{s. t. } x_i^L \leq x_i \leq x_i^U, i = 1, 2, \dots, m \quad (6)$$

$$d_{j,l} \leq x_{m+j} \leq d_{j,q}, j = 1, 2, \dots, n \quad (7)$$

(6)、(7)式中的变量均为连续变量, $d_{j,l}$ 和 $d_{j,q}$ 分别为 $x_{m+j}$ 的下界和上界。

## 2 禁忌微粒群算法

### 2.1 基本PSO算法

PSO算法<sup>[7]</sup>是一种模拟鸟群觅食行为而设计的群体智能优化算法,由美国的Kennedy博士和Eberhart博士于1995年提出。PSO算法中每个微粒即为解空间的一个解,它根据自己的飞行经验和同伴的飞行经验来调整自己的飞行。

假设在一个 $D$ 维搜索空间中,由 $N_{pop}$ 个微粒组成的种群为 $X = (x_1, x_2, \dots, x_{N_{pop}})$ ,其中第 $i$ 个微粒的位置为 $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})^T$ ,其速度为 $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})^T$ 。微粒 $x_i$ 在 $D$ 维空间中将按照(8)、(9)式来更新微粒的速度和位置,即

$$v_{iD}^{k+1} = wv_{iD}^k + c_1r_1(p_{iD}^k - x_{iD}^k) + c_2r_2(p_{gD}^k - x_{iD}^k) \quad (8)$$

$$x_{iD}^{k+1} = x_{iD}^k + v_{iD}^{k+1} \quad (9)$$

(8)式中 $p_{iD}^k$ 是微粒 $x_i$ 直到第 $k$ 次迭代的最优解, $p_{gD}^k$ 是在第 $k$ 次迭代中整个种群的最优解, $c_1, c_2$ 为加速常数,它使每个微粒向 $p_i$ 和 $p_g$ 位置加速运动,常取值为 $c_1 + c_2 = 4$ , $r_1, r_2$ 为 $[0, 1]$ 区间的随机数, $w$ 为

惯性因子,随着迭代过程按(10)式线性减小。

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{k_{\max}} \times k \quad (10)$$

(10)式中 $k_{\max}$ 是迭代的最大次数, $w_{\max}$ 和 $w_{\min}$ 是惯性因子的最大值和最小值,在实验中通常取 $w_{\max} = 0.9$ 和 $w_{\min} = 0.4$ 。

在用PSO求解问题(5)式的过程中,第 $i$ 个微粒的初始位置 $x_i$ 被任意设定,其对应的罚参数 $s_i$ 由(11)式决定。

$$s_i = 1 + \phi(x_i), i = 1, 2, \dots, N_{pop} \quad (11)$$

(4)式中的罚参数 $s$ 为 $s_i$ 之间的最小值

$$s = \min\{s_1, s_2, \dots, s_{N_{pop}}\}$$

同时,在迭代过程中使用(12)式更新罚参数<sup>[8]</sup>。

$$s^{k+1} = s^k \times \exp(1 + \phi(p_g^k)) \quad (12)$$

(12)式中的 $p_g^k$ 表示第 $k$ 次迭代中整个微粒群的最优解。此方法主要是为增广目标函数提供多峰性,并在离散值附近产生极值。

### 2.2 禁忌搜索

禁忌搜索<sup>[9]</sup>是Glover在1986年提出的一种人工智能算法,是局部算法的扩展。它的一个重要思想是标记已得到的局部最优解或求解的过程,并在进一步的迭代中避开这些局部最优解或过程,以此来跳出局部最优解。

### 2.3 应用TS-PSO求解混合离散优化问题

2.3.1 TS-PSO的设计思想 PSO算法具有全局寻优、实现简单的优点,但是算法运行过程中易陷入局部最优而导致早熟,求解准确度不高。为保持群体中微粒的多样性,扩大搜索空间,避免算法陷入局部最优导致早熟。本文利用多峰增广目标函数 $F(x)$ 波谷之间的距离近似相等的特点构建禁忌表,记录PSO算法搜索到的当前最优位置作为禁忌对象。通过结合禁忌搜索和PSO构建TS-PSO算法,在禁忌搜索中通过PSO产生新的微粒种群构成邻域解。在搜索过程中,选择候选解集中未被禁忌或被禁忌但满足特赦准则的最佳候选解替代当前解进行迭代。禁忌搜索终止后,以禁忌表中存储的最优微粒作为算法的最优解输出。

2.3.2 TS-PSO的算法流程 首先对TS-PSO算法中用到的概念进行说明。

定义1 邻域解 PSO算法更新过程中,对于某一个微粒,取和它同代生成的所有其他微粒为其邻域解。

定义2 适应度函数。取经过罚函数处理离散

变量后构造的增广目标函数  $F(x)$  为适应度函数。

定义 3 禁忌对象和禁忌表长度。将每次迭代得到的未被禁忌的最好解放入禁忌表中,作为禁忌对象。禁忌表长度指禁忌表中禁忌对象的个数。

定义 4 候选解集。取邻域解中未被选择的微粒组成候选解集。

定义 5 特赦准则。如果一个微粒被禁忌,但其优于禁忌表中对应的禁忌对象,则该微粒满足特赦准则。

定义 6 终止准则。当迭代次数大于特定值时,满足终止准则。

TS-PSO 算法的流程框图如图 1 所示,算法一般步骤可描述如下。

步骤 1 初始化。设定 PSO 算法的参数( $w, c_1, c_2$  等)随机产生初始种群,并对每个微粒的速度和位置进行初始设定,计算初始种群中每个微粒的适应度函数值,设定每个微粒的初始历史最优解,求解种群当前最优解,设定 TS 的参数,置禁忌表为空;

步骤 2 判断 TS 终止准则是否满足,若是,则转步骤 8,否则转下一步;

步骤 3 根据进化方程(8)、(9)式更新每个微粒的速度和位置,重新计算各微粒的适应度函数值,选择种群中的最优微粒为当前解,更新邻域解和候选解集,转下一步;

步骤 4 判断当前解是否被禁,若是,则转下一步,否则,转步骤 6;

步骤 5 判断当前解是否满足特赦准则,若是,转步骤 7,否则选候选解集中的最优解为当前解,返回步骤 4;

步骤 6 判断当前解是否优于禁忌表中的某个禁忌对象,若是,则转下一步,否则,返回步骤 2;

步骤 7 更新禁忌表,返回步骤 2;

步骤 8 输出最优解,结束搜索。

### 3 数值实验

为验证 TS-PSO 算法的优化性能,使用 Rosenbrock's 测试函数和压力管设计问题来对算法进行测试。

#### 3.1 Rosenbrock's 测试函数

考虑下面的 Rosenbrock's 测试函数<sup>[10]</sup>。该函数的两个变量  $x_1$  和  $x_2$  均为离散值,并且以 0.55 的步长移动。

$$\min f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

$$\text{s. t. } -0.55 \leq x \leq 4.95$$

Rosenbrock's 测试函数的目标函数  $f(x)$  和增广目标函数  $F(x)$  的图形如图 2 所示。该问题的离散最优解是  $x_{opt}^D = (1.65, 2.75)^T$ ,连续最优解是  $x_{opt}^C = (1, 1)^T$ 。将所有离散值用圆点表示绘制连续最优解和离散最优解位置关系,如图 3 所示。这是一个离散最优解不可能通过连续最优解的舍入法或就近原则找到的典型问题。

微粒个数设为 50,搜索的最大迭代次数  $k_{max}$  设为 200。采用 Matlab 进行编程,用 TS-PSO 算法和基本 PSO 算法求解 Rosenbrock's 测试函数,采用随机初始种群进行 100 次实验,结果见表 1 和表 2。

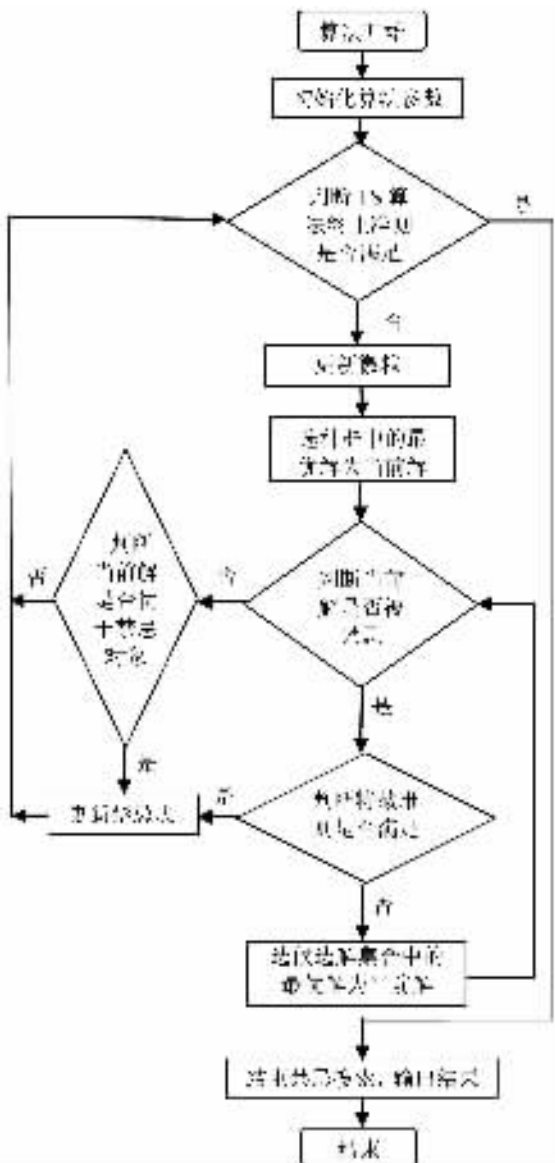


图 1 TS-PSO 算法流程

Fig. 1 The flow chart of TS-PSO

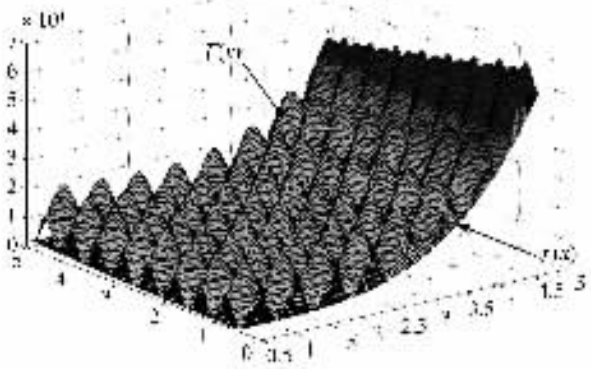


图 2 Rosenbrock's 函数  
Fig. 2 Rosenbrock's function

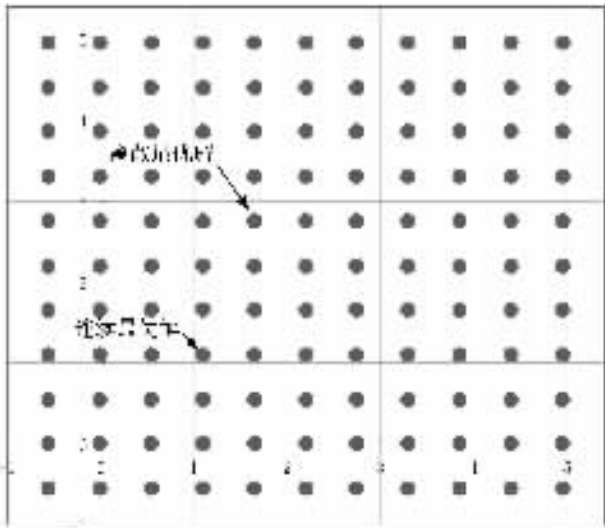


图 3 连续最优解和离散最优解  
Fig. 3 Discrete and continuous optimum

表 1 100 次实验结果

Tab. 1 Statistical results

方法	搜索到最优解的次数	时间/s
PSO	23	7.703 1
TS-PSO	94	9.031 25

表 2 PSO 和 TS-PSO 算法的统计结果

Tab. 2 Statistical results of PSO and TS-PSO methods

方法	最优值	平均值	最差解	标准偏差
PSO	0.498 125	1.325 7	8.528 1	1.033 1
TS-PSO	0.498 125	0.528 2	1.000 0	0.119 2

从表 1 可见,TS-PSO 算法比 PSO 算法计算时间要多,但是搜索到最优解的几率比 PSO 算法高。从表

2 可见,两种算法都找到了唯一的最优解,但是 TS-PSO 算法所求得平均值、最差值和标准偏差均优于 PSO 算法,由此可见 TS-PSO 算法的总体性能优于 PSO 算法。

### 3.2 压力管设计问题的求解结果

压力管设计问题来自文献[10],压力管结构如图 4 所示。优化目标是求满足一系列约束条件的 4 个设计变量  $T_h$ ( $x_1$ , 圆柱形管子的厚度)、 $T_h$ ( $x_2$ , 半球形盖子的厚度)、 $R$ ( $x_3$ , 圆柱形管子的内径)和  $L$ ( $x_4$ , 圆柱形管子的长度,不包括两端的盖子),使得包括材料、焊接、铸造等费用在内的总费用最少。其中  $T_h$  和  $T_h$  均为 0.062 5 的整数倍,  $R$  和  $L$  是连续变量。

该问题的数学模型描述如下。

$$\begin{aligned} \min f(x) &= 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + \\ & 3.1661x_1^2x_4 + 19.84x_1^2x_3 \\ \text{s. t. } & 0.0625 \leq x_1, x_2 \leq 99 \\ & 10 \leq x_3, x_4 \leq 200 \\ g_1(x) &= -x_1 + 0.0193x_3 \leq 0 \quad (13) \\ g_2(x) &= -x_2 + 0.00954x_3 \leq 0 \quad (14) \\ g_3(x) &= -\pi x_3^2x_4 - 4\pi x_3^3/3 + 1296000 \leq 0 \quad (15) \\ g_4(x) &= x_4 - 240 \leq 0 \quad (16) \end{aligned}$$

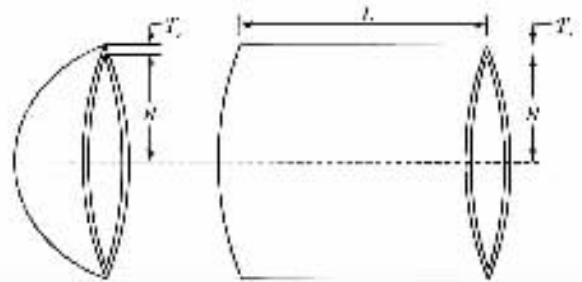


图 4 压力管设计问题

Fig. 4 Pressure vessel

利用罚函数构造如(4)式形式的增广目标函数  $F(x)$ 。其中,行为约束(13)~(16)式的罚参数  $r$  设为  $r = 1.0 \times 10^8$ 。使用 50 个微粒迭代 5 000 次。采用随机初始种群进行 30 次实验,最优结果如表 3 的最后一列所示。

TS-PSO 算法与如下算法进行比较,包括遗传自适应搜索法[11]、扩张拉格朗日乘子法[10]、分支定界法[12]、基于协进化模型的 GA[13]、基于支配选择机制的 GA[14]、基于协进化 PSO 算法[15]。表 3 列出了各种算法求得的最优解,表 4 则给出了统计结果。

表 3 不同算法求解压力管设计问题所得的最优解

Tab.3 Comparison of different algorithms for pressure vessels design problem

设计变量	文献 11 ]	文献 10 ]	文献 12 ]	文献 13 ]	文献 14 ]	文献 15 ]	TS-PSO
$x_1( T_s )$	1. 125 0	1. 125 0	0. 937 5	0. 812 5	0. 812 5	0. 812 5	0. 812 5
$x_2( T_h )$	0. 625 0	0. 625 0	0. 500 0	0. 437 5	0. 437 5	0. 437 5	0. 437 5
$x_3( R )$	47. 700 0	58. 291 0	48. 329 0	40. 323 9	42. 097 4	42. 091 3	42. 098 4
$x_4( L )$	117. 701 0	43. 690 0	112. 679 0	200. 000 0	176. 654 1	176. 746 5	176. 636 6
$g_1( x )$	- 0. 204 4	0. 000 02	- 0. 004 8	- 0. 034 3	- 0. 000 02	- 0. 000 1	- 0. 000 0
$g_2( x )$	- 0. 169 9	- 0. 068 9	- 0. 038 9	- 0. 052 8	- 0. 035 9	- 0. 035 9	- 0. 035 88
$g_3( x )$	54. 226 0	- 21. 220 1	- 3 652. 876 8	- 27. 105 8	- 27. 886 1	- 116. 382 7	- 0. 327 0
$g_4( x )$	- 122. 299 0	- 196. 310 0	- 127. 321 0	- 40. 000 0	- 63. 346 0	- 63. 253 5	- 63. 363 4
$f( x )$	8 129. 103 6	7 198. 042 8	6 410. 381 1	6 288. 744 5	6 059. 946 3	6 061. 077 7	6 059. 714 3

表 4 不同算法求解压力管设计问题的统计结果

Tab.4 Statistical results of TS-PSO and other algorithms for pressure vessel

方法	最优解	平均值	最差值	标准偏差
文献 11 ]	8 129. 103 6	N/A	N/A	N/A
文献 10 ]	7 198. 042 8	N/A	N/A	N/A
文献 12 ]	6 410. 381 1	N/A	N/A	N/A
文献 13 ]	6 288. 744 5	6 293. 843 2	6 308. 149 7	7. 413 3
文献 14 ]	6 059. 946 3	6 177. 253 3	6 469. 322 0	130. 929 7
文献 15 ]	6 061. 077 7	6 147. 133 2	6 363. 804 1	86. 454 5
TS-PSO	6 059. 714 3	6 063. 148 8	6 288. 677 0	53. 187 6

其中 表 4 中 N/A 表示 Not Available。

由表 3 和表 4 可见 ,TS-PSO 算法求得的最优解、平均值和最差解均优于其他方法的结果。此外 ,由表 4 可见 ,TS-PSO 算法求得的标准偏差仅次于文献 13 ] ,说明 TS-PSO 算法的稳定性较好 ,而且 TS-PSO 算法求得的最差解比文献 10-12 ]求得的最优解要好。

上述两个问题的数值实验和比较结果表明 ,TS-PSO 算法具有有效的搜索能力和很好的初值鲁棒性 ,即对初始值的依赖性较小。

## 5 结论

本文用罚函数处理混合离散优化问题中的离散变量 ,将原问题转化为目标函数为多峰函数的连续变量优化问题 ,然后利用混合 PSO 算法求解转换得到的连续变量问题。针对 PSO 算法可能陷入局部最优而出现早熟的特点 ,将 TS 算法与 PSO 算法相结合 ,构造 TS-PSO 算法。TS-PSO 算法利用 TS 算法具有的较强“爬山”能力 ,搜索时能够跳出局部最优解 ,转向解空间的其他区域。这样 ,实现了两种算法优势互补 ,提高了微粒种群的多样性和微粒搜索的遍

历性。数值实验结果表明 ,TS-PSO 算法能够提高混合离散优化问题的求解准确度 ,收敛到全局最优或尽可能靠近全局最优 ,是一种比较有效的优化算法。

## 参考文献 :

- [ 1 ] Arora J S ,Huang M W. Methods for optimization of nonlinear problems with discrete variables :a review[ J]. Structural Optimization ,1994 ,8( 2 ) :69-85.
- [ 2 ] Eberhart R ,Kennedy J. Swarm intelligence[ M ]. San Francisco :Morgan Kaufman Publishers ,2001 :33-52.
- [ 3 ] Venter G ,Sobieski J S. Particle swarm optimization[ J ]. AIAA Journal 2003 ,41( 8 ) :1583-1589.
- [ 4 ] Shin D K ,Gurdal Z ,Griffin O H. A penalty approach for nonlinear optimization with discrete design variables[ J ]. Engineering Optimization ,1990 ,16 :29-42.
- [ 5 ] 张玉芳 ,薛青松 ,熊忠阳. 基于禁忌搜索的动态粒子群算法[ J ]. 计算机工程与应用 2008 ,44( 24 ) :56-58.
- [ 6 ] Song X Y ,Chang C Y ,Cao Y. New particle swarm algorithm for job shop scheduling problems[ C ]. World Congress on Intelligent Control and Automation 2008.
- [ 7 ] Eberhart R C ,Kennedy J. A new optimizer using particle swarm theory[ C ]. Proceedings of the sixth international symposium on micro machine and human science ,Nagoya , Japan IEEE service center ,1995.
- [ 8 ] Kitayama S ,Arakawa M ,Yamazaki K. Penalty function approach for the mixed discrete non-linear problems by particle swarm optimization[ J ]. Structural and Multidisciplinary Optimization 2006 ,32( 3 ) :191-202.
- [ 9 ] Glover F. Tabu search :Part I[ J ]. ORSA Journal on Computing ,1989 ,1( 3 ) :190-206.
- [ 10 ] Kannan B K ,Kramer S N. An augmented lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design[ J ]. Transactions of the ASME ,Journal of Mechanical Design ,1994 ,116 :318-320.

- [ 11 ] Deb K ,Gene A S. A robust optimal design technique for mechanical component design[ C ]//Dasgupta D ,Michalewicz Z. Evolutionary Algorithms in Engineering Applications. Berlin Springer-Verlag , 1979 :497-514.
- [ 12 ] Sandgren E. Nonlinear integer and discrete programming in mechanical design[ C ]//Kissimee , Proceedings of the ASME Design Technology Conference , 1988 :95-105.
- [ 13 ] Coello C A C. Use of a self-adaptive penalty approach for engineering optimization problems[ J ]. Computers in Industry 2000 41 :113-127.
- [ 14 ] Coello C A C ,Montes E M. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection[ J ]. Advanced Engineering Informatics ,2002 , 16 :193-203.
- [ 15 ] He Q ,Wang L. An effective co-evolutionary particle swarm optimization for constrained engineering design problems [ J ]. Engineering Applications of Artificial Intelligence. 2006 20 89-99.

## Operations Research and Cybernetics

### A Hybrid Tabu Search and Particle Swarm Optimization Algorithm for Mixed Discrete Optimization Problems

ZHANG Xing-hui , BAI Fu-sheng

( College of Mathematics Science , Chongqing Normal University , Chongqing 400047 , China )

**Abstract :** Particle swarm optimization( PSO ) algorithm is mainly used to find global solutions of continuous variables optimization problems. In this paper , the penalty function approach to handle the discrete variables is employed , in which mixed discrete optimization problem :  $\min f(x)$  , s. t.  $g_k(x) \leq 0$  ,  $k = 1, 2, \dots, l$  ,  $x_i^L \leq x_i \leq x_i^U$  ,  $i = 1, 2, \dots, m$  ,  $x_{m+j} \in D_j$  ,  $D_j = (d_{j,1}, d_{j,2}, \dots, d_{j,n_j})$  ,  $j = 1, 2, \dots, n$  is handled as continuous one :  $\min F(x)$  , s. t.  $x_i^L \leq x_i \leq x_i^U$  ,  $i = 1, 2, \dots, m$  ,  $d_{j,1} \leq x_{m+j} \leq d_{j,n_j}$  ,  $j = 1, 2, \dots, n$ . Standard PSO algorithm will likely fall into local optimal solution and exist premature convergence. Tabu search( TS ) algorithm has good hill-climbing ability and can escape from the local optimal solution and turn to other parts of the solution space. A neighborhood structure is designed and a hybrid tabu search and particle swarm optimization( TS-PSO ) algorithm is proposed , which has memory ability and efficient hill-climbing capability. Simulation results on Rosenbrocks function and pressure vessel design show that the disadvantage of getting in the local best point of standard PSO is overcome effectively and the ability of global optimality is toned up.

**Key words :** discrete variables ; penalty function ; particle swarm optimization ; tabu search

( 责任编辑 黄 颖 )