第 34 卷　第 5 期　　　　计　算　机　工　程　　　　2008 年 3 月
Vol.34　　*No.5*　　　　Computer Engineering　　　　March 2008

·人工智能及识别技术·　　文章编号：1000—3428(2008)05—0179—03　　文献标识码：A　　中图分类号：TP301.6

基于文化粒子群算法的约束优化问题求解

高丽丽，刘　弘，李同喜

(山东师范大学信息科学与工程学院，济南 250014)

摘　要：提出一种基于文化算法的粒子群优化算法(PSO)。该算法在群体空间采用基于高斯概率分布和柯西概率分布的改进 PSO 算法，在信念空间根据形势知识和规范化知识指导种群的进化，充分利用优秀个体所包含的信息，提高了算法的进化速度。实验表明，该算法的优化性能和效率优于基本 PSO 算法。

关键词：粒子群优化算法；文化算法；约束优化问题

Particle Swarm Based on Cultural Algorithm for Solving Constrained Optimization Problems

GAO Li-li, LIU Hong, LI Tong-xi

(Dept. of Information & Engineering, Shandong Normal University, Jinan 250014)

【Abstract】 A Particle Swarm Optimization(PSO) based on cultural algorithm for solving constrained optimization problems is proposed. This algorithm employs PSO using Gaussian and Cauchy probability distributions in population space, uses situational knowledge and normative knowledge in belief space to guide the evolution of the population. In this way, it exploits the information sufficiently that the optimum individual carries and speeds up the evolutionary process. Experimental results prove the algorithm is superior to basic PSO in quality and efficiency.

【Key words】 particle swarm optimization; cultural algorithm; constrained optimization

大多数求解约束优化问题的算法都基于梯度概念，要求目标函数和约束条件可微，且一般只能求得局部最优解。粒子群优化算法具有易于实现、不要求目标函数和约束条件可微等优点，目前已成功应用在很多优化问题中。但文献[1]指出，PSO 算法不是全局收敛算法，容易陷入局部最优，出现“早熟”现象。为了增强粒子群算法的全局搜索能力和跳出局部最优能力，并加快算法的计算速度和精度，本文提出一种基于文化算法的粒子群算法(PSO-CA)来求解约束优化问题。

1　粒子群优化算法

粒子群算法是一种基于群体的演化算法，将每个个体看作 1 个没有体积、没有重量的微粒，在搜索空间中以一定速度飞行。迭代时，粒子x_i通过跟踪个体历史最优解 p_i^t 和全局历史最优解 p_g^t 更新位置和速度。基本PSO算法中，速度更新公式和位置更新公式分别如式(1)和式(2)所示。

$$v_i^{t+1} = w \cdot v_i^t + c_1 \cdot rand() \cdot [p_i^t - x_i^t] + c_2 \cdot Rand() \cdot [p_g^t - x_i^t] \quad (1)$$

$$x_i^{t+1} = x_i^t + \Delta t \cdot v_i^t \quad (2)$$

其中，c_1和c_2为加速正常数；w为惯性权重。

w 决定了先前速度对当前速度的影响程度，能够平衡算法的全局搜索和局部搜索能力。为使算法前期有较高的探索能力，后期有较高的开发能力，本文设 w 随时间线性减少。

rand()和 *Rand*()是两个产生随机数的统一概率分布，文献[2]指出：使用其他概率分布可以改善算法性能，如柯西分布(Cauchy probability distribution)能增强算法逃出局部最优的能力，高斯分布(Gaussian probability distribution)能加速局部搜索收敛的速度。本文用柯西分布产生认知部分的随机数，用高斯分布产生社会部分的随机数，两者都取[-1,1]区间，然后映射到[0,1]区间。式(1)变为

$$v_i^{t+1} = w^t \cdot v_i^t + c_1 \cdot cd() \cdot [p_i^t - x_i^t] + c_2 \cdot Gd() \cdot [p_g^t - x_i^t] \quad (3)$$

2　文化粒子群算法

2.1　文化算法(CA)

文化能使种群以一定的速度进化并适应环境，这个速度超越了单纯依靠基因遗传生物的进化速度[3]。在种群进化过程中，个体知识的积累和群体内部知识的交流在另一个层面促进群体的进化。受这些思想的启发，文献[3]于 1994 年提出文化算法，其重要思想是从进化的种群中获取待解决问题的知识，并反馈这些知识来指导搜索过程，从而提高搜索效率。文化算法的基本框架如图 1 所示。

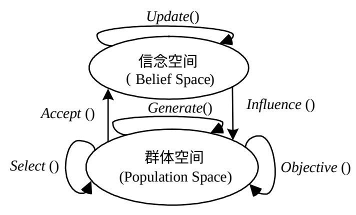

图 1　文化算法框架

文化算法由群体空间和信念空间组成，这 2 个空间通过一组由接收函数 *accept*()和影响函数 *influence*()组成的通信协议联系在一起。*accept*()用来收集种群中所选个体的经验知识，信念空间利用更新函数 *update*()进行调整；而 *influence*()

基金项目：国家自然科学基金资助项目(69975010，60374054)；山东省自然科学基金资助项目(Y2003G01，Z2006G09)

作者简介：高丽丽(1982－)，女，硕士研究生，主研方向：进化计算，协同设计；刘　弘，教授、博士；李同喜，硕士研究生

收稿日期：2007-09-21　**E-mail：**snowgao@163.com

能利用信念空间中待解决问题的经验知识来指导种群空间的进化。和其他进化算法相比，文化算法更能准确地反映社会的进化过程，具有很好的应用前景。

2.2 文化粒子群算法(PSO-CA)

任何一种基于种群的进化算法都可以为文化算法的群体空间提供种群，如GA, EC, EP等，最近又提出了基于PSO的群体空间[4]。对于信念空间，相关人士定义了 5 种对决策制定非常有用的知识，其中，最常用的 2 类是规范化知识和形势知识。本文采用PSO作为群体空间，并在信念空间中使用形势知识和规范化知识来引导群体空间中种群的进化。

2.2.1 约束条件的处理

约束优化问题的一般形式为

$$\min f(\boldsymbol{x})$$
$$\text{s.t.}\begin{cases} g_j(\boldsymbol{x}) \le 0 & j=1,2,\cdots,q \\ h_p(\boldsymbol{x})=0 & p=1,2,\cdots,m \\ x_i^l \le x_i \le x_i^u & i=1,2,\cdots,n \end{cases} \tag{4}$$

其中，$\boldsymbol{x}=(x_1,x_2\cdots,x_n)\in R^n$ 是n维实向量；$f(\boldsymbol{x})$为目标(适应值)函数；g_j表示第j个不等式约束；h_p表示第p个等式约束；$[x_i^l,x_i^u]$是变量x_i的区间约束。

约束条件的处理是约束优化问题处理过程中 1 个重要环节。目前，最广泛的处理方法是罚函数法，即根据约束的特点，构造某种惩罚函数，把它加到目标函数中，将约束优化问题转化为无约束问题。本文将式(4)转化为如下无约束优化问题:

$$F(x)=f(x)+R\cdot(\sum_{j=1}^{q}\max(0,g_j(x))+\sum_{p=1}^{m}\left|h_p(x)\right|) \tag{5}$$

其中，R 为罚系数，取值为很大的正数。不可行解被赋予很大的适应值，且离约束边界越远，不满足的约束条件越多，受到的惩罚就越重。

对于变量的区间约束[x_i^l,x_i^u]，当某粒子x_i不满足此区间约束时，采用式(6)进行修正。

$$\begin{cases} x_i=x_i+w\cdot rand[0,1]\cdot[x_i^u-x_i^l] & x_i<x_i^l \\ x_i=x_i-w\cdot rand[0,1]\cdot[x_i^u-x_i^l] & x_i>x_i^u \end{cases} \tag{6}$$

其中，$w\in[0,1]$，具体取值视具体问题而定，本文取 0.01。

2.2.2 信念空间中知识的获取和更新

PSO 演化过程中，粒子跟踪 2 个极值：个体历史最优解和全局历史最优解，这些最优解对迭代搜索是很有用的知识信息，用这些信息不断地更新信念空间，并指导群体空间的进一步求解，通过群体空间和信念空间的相互影响，使求解具有更好的全局搜索能力。

信念空间主要使用形势知识(SK)和规范化知识(NK)，即*BLF*=<*SK*,*NK*>。形势知识由从种群中选择出来的优秀个体组成，是其他个体行为的“典范”，形势知识的数据结构如图 2 所示。

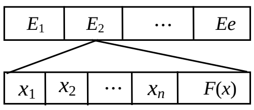

图 2 形势知识的数据结构

在图 2 中，列表的每个优秀个体包含 2 类值，即每个变量的值和该个体所对应的适应度值。其中，$F(E_1)\ge F(E_2)\ge\cdots\ge F(E_e)$。

初始化形势知识时，取初始种群中前 e($e=\alpha\%popsize$)个最优个体，形势知识的更新取决于每代的最优个体 x_{best}^t，若 x_{best}^t 优于当前列表中的最差个体 E_1^t，则去掉 E_1^t，并将 x_{best}^t 插入到 E_i^t 和 E_{i+1}^t 之间($F(E_i^t)\ge F(x_{best}^t)>F(E_{i+1}^t)$)。

规范化知识由各决策变量区间组成，这些区间由当前种群中的优秀个体集合(即形势知识)决定，给出了最优解最可能出现的范围，即个体行为的标准。迭代时根据粒子所在的具体位置决定粒子下一次的飞行速度，能更快地找到最优解。规范化知识的数据结构如图 3 所示。

l_1	u_1	l_2	u_2	$\cdots$	l_n	u_n
L_1	U_1	L_2	U_2	$\cdots$	L_n	U_n

图 3 规范化知识的数据结构

在图 3 中，l_i和 u_i 分别表示第i个决策变量的下边界和上边界；L_i和U_i分别表示这 2 个边界对应的适应度值。初始化时，l_i和 u_i 设为变量的定义域区间，对于最小化问题，L_i和U_i设为 $+\infty$，在演化过程中，这 4 个值会根据优秀个体集合所在区间不断地调整。通过*Accept*()函数在群体空间内选择能直接影响当前信念空间的粒子。本文选择当前群体空间中适应度最低的个体来重新修正信念空间中的形势知识，即更新优解集合，然后通过*Update*()函数更新信念空间中的规范化知识。假设第i个粒子影响第j个决策变量的下限，第k个个体影响j的上限，则通过式(7)更新规范化知识。

$$l_j^{t+1}=\begin{cases} x_{i,j}^t & \text{if } x_{i,j}^t\le l_j^t \text{ or } F(x_i^t)<L_j^t \\ l_j^t & \text{otherwise} \end{cases}$$
$$L_j^{t+1}=\begin{cases} obj(x_i) & \text{if } x_{i,j}^t\le l_j^t \text{ or } F(x_i^t)<L_j^t \\ L_j^t & \text{otherwise} \end{cases}$$
$$u_j^{t+1}=\begin{cases} x_{k,j}^t & \text{if } x_{k,j}^t\ge u_j^t \text{ or } F(x_k^t)<U_j^t \\ u_j^t & \text{otherwise} \end{cases}$$
$$U_j^{t+1}=\begin{cases} obj(x_k) & \text{if } x_{k,j}^t\ge u_j^t \text{ or } F(x_k^t)<U_j^t \\ U_j^t & \text{otherwise} \end{cases} \tag{7}$$

2.2.3 规范化知识对群体空间的影响

文化算法设计中如何利用信念空间中的知识影响群体空间中下一代种群的生成很重要。本文信念空间通过*Influence*()函数对群体空间的粒子施加影响，根据粒子的当前位置确定下一步的速度，式(3)变为式(8)。

$$v_{i,j}^{t+1}=\begin{cases} w^t\cdot v_{i.j}^t+c_1\cdot cd()\cdot[p_{i,j}^t-x_{i,j}^t]+c_2\cdot Gd()\cdot[p_{g,j}^t-x_{i,j}^t] \\ \qquad \text{if } x_{i,j}^t<l_j^t \text{ and } x_{i,j}^t<p_{g,j}^t \\ w^t\cdot v_{i.j}^t-c_1\cdot cd()\cdot[p_{i,j}^t-x_{i,j}^t]-c_2\cdot Gd()\cdot[p_{g,j}^t-x_{i,j}^t] \\ \qquad \text{if } x_{i,j}^t>u_j^t \text{ and } x_{i,j}^t>p_{g,j}^t \\ w^t\cdot v_{i.j}^t\pm c_1\cdot cd()\cdot[p_{i,j}^t-x_{i,j}^t]\pm c_2\cdot Gd()\cdot[p_{g,j}^t-x_{i,j}^t] \\ \qquad \text{otherwise} \end{cases} \tag{8}$$

其中,“ ± ”号表示按相同概率取“ + ”或“ - ”。不同的粒子，当前位置、状态的不同，使它们受到的影响也不同，这体现了知识对各粒子不同的指导作用，使算法更合理有效。

2.2.4 文化粒子群算法流程

求解约束优化问题的 PSO-CA 算法流程如下：

(1)t=0；

(2)在群体空间中，初始化一群微粒 $pop(t)$，包括位置和速度；

(3) 初始化信念空间中的形势知识和规范化知识 *BLF*(t)=<*SK*,*NK*>；

(4)评价每个粒子的适应度值；

(5)更新信念空间；

(6)在信念空间知识的指导下，按式(2)、式(8)更新当前种群中粒子的速度和位置；

(7)t=t+1；

(8)选择新的子代个体 $pop(t\text{-}1)$；

(9)返回步骤(4)直到满足终止条件。

3 仿真实验

为验证算法有效性，本文选择 4 个典型优化问题进行性能测试，并与基本粒子群算法(bPSO)进行了比较。比较结果如表 1 所示。实验中，bPSO 算法参数设置如下：群体规模 N=40; $c1$=$c2$=2; Δt =1; w 由 0.7 线性减少到 0.4。对于 PSO-CA 算法，R=50000, $\alpha\%=15\%$。其中，$f_1(x)$ 是一个线性规划优化问题，已知全局最优解为 $f_1(x^*)=-9.0$。

$$\min f_1(x)=-4x_1-3x_2$$

$$\text{s.t.}\begin{cases}2x_1+3x_2-6\le 0\\-3x_1+2x_2-3\le 0\\2x_1+x_2-4\le 0\\0\le x_i\le 2, i=1,2\end{cases}$$

$f_2(x)$既有不等式约束，又有等式约束，已知全局最优解为 $f_2(x^*)=1.393$。

$$\min f_2(x)=(x_1-2)^2+(x_2-1)^2$$

$$\text{s.t.}\begin{cases}x_1-2x_2+1=0\\\dfrac{x_1^2}{4}+x_2^2-1\le 0\end{cases}$$

$f_3(x)$，$f_4(x)$ 的约束条件全部为非线性不等式约束，$f_4(x)$ 较为复杂，已知全局最优解 $f_3(x^*)=-5.507\,9$，$f_4(x^*)=$ 680.6300573。

$$\min f_3(x)=-x_1-x_2$$

$$\text{s.t.}\begin{cases}-2x_1^4+8x_1^3-8x_1^2+x_2-2\le 0\\-4x_1^4+32x_1^3-88x_1^2+96x_1+x_2-36\le 0\\0\le x_1\le 3\\0\le x_2\le 4\end{cases}$$

$$\min f_4(x)=(x_1-10)^2+5(x_2-12)^2+x_3^4+3(x_4-11)^2+10x_5^6+7x_6^2+x_7^4-4x_6x_7-10x_6-8x_7$$

$$\text{s.t.}\begin{cases}-127+2x_1^2+3x_2^4+x_3+4x_4^2+5x_5\le 0\\-282+7x_1+3x_2+10x_3^2+x_4-x_5\le 0\\-196+23x_1+x_2^2+6x_6^2-8x_7\le 0\\4x_1^2+x_2^2-3x_1x_2+2x_3^2+5x_6-11x_7\le 0\\-10\le x_i\le 10, i=1,2,\cdots,7\end{cases}$$

对每个优化问题，在相同的运行环境下，使用 2 种方法分别独立求解 100 次，所求最优解和平均最优解以及取得最优解时的平均运行时间见表 1。可以看出，PSO-CA 算法能在较短时间内收敛，大约能缩短 1/3 的运行时间。并且求得最优解的成功率得到极大提高，搜索的结果优于 bPSO 算法，由于该算法充分利用了优秀个体所蕴含的知识，并对不同的个体加以不同的引导，因此能够提高算法的收敛速度和精度。

表 1 实验结果比较

f	方法	最优解	平均最优解	平均运行时间/s	成功率/(%)
f_1	bPSO	-8.999 700	-8.999 500	8.085	100
	PSO-CA	-8.999 900	-8.999 800	5.231	100
f_2	bPSO	1.393 434	1.394 006	9.299	98
	PSO-CA	1.393 431	1.393 439	5.345	100
f_3	bPSO	-5.507 000	-5.506 800	12.894	64
	PSO-CA	-5.507 800	-5.507 500	7.252	98
f_4	bPSO	680.633 800	680.634 800	23.211	72
	PSO-CA	680.632 100	680.632 500	17.637	96

测试实验结果表明，本文提出的 PSO-CA 算法对于约束条件没有要求，不等式约束、等式约束、线性约束和非线性约束，都比使用基本 PSO 算法具有更好的性能。

参考文献

[1] Van den Bergh F. Analysis of Particle Swarm Optimizers[D]. South Africa: University of Pretoria, 2001.

[2] Coelho L S, Krohling R A. Predictive Controller Tuning Using Modified Particle Swarm Optimization Based on Cauchy and Gaussian Distributions[C]//Proc. of the 8th Online World Conference on Soft Computing in Industrial Applications. Dortmund, Germany: [s. n.], 2005.

[3] Robert R G. An Introduction to Cultural Algorithms[C]//Proc. of the 3rd Annual Conf. on Evolution Programming. [S. l.]: World Scientific Publishing, 1994: 131.

[4] Iacoban R, Reynolds R G, Brewster J. Cultural Swarms: Modeling the Impact of Culture on Social Interaction and Problem Solving[C]//Proc. of IEEE Swarm Intelligence Symposium. Indianapolis, USA: [s. n.], 2003: 205.

（上接第 175 页）

60)，采用改进遗传算法和遗传算法分别进行了仿真。仿真实验的结果如表 1 所示。

表 1 本文算法与其他算法的实验结果比较

	第 1 组	第 2 组	第 3 组	第 4 组
简单遗传算法	0.256 6	0.240 3	0.239 7	0.239 7
文献[4]遗传算法	0.082 1	0.032 4	0.008 2	0.000 6
改进的遗传算法	0.081 0	0.028 9	0.007 8	0.000 4

仿真实验分为 4 组，每组都独立执行 5 次，每组迭代的次数分别为 1 000, 2 000, 3 000, 4 000, 5 000 次。

算法的标准为：相对距离＝(本次进化的最优解－已知最优解)，其中，已知最优解为 15 404 km。每组的相对距离为独立执行 5 次所得到的平均值。

从实验结果可以看出，采用了改进的遗传算法的综合控制策略在收敛速度和解的质量上有明显的提高，得到了已知的最优解，并且与文献[3]的结果相比更优。因此，基于改进的遗传算法对解决 TSP 问题具有好的稳定性、收敛性和精确性。

参考文献

[1] Jong-Hwan Kim, Hong-Kook Chae, Jeong-Yull Jeon, et al. Identification and Control of Systems with Friction Using Accelerated Evolutionary Programming[J]. IEEE Control System, 1996, 16(4): 38-47.

[2] Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs[M]. Beijing: Science Publishing House, 2000.

[3] 陈 斌, 徐华中. 一种改进遗传算法及其在 TSP 问题中的应用[J]. 计算机工程, 2002, 28(9): 90-93.

[4] 刘 海, 郝志峰. 改进遗传交叉算子求解 TSP 问题[J]. 华南理工大学学报: 自然科学版, 2002, 30(12): 70-73.