第47卷第4期 大连理工大学学报 Vol. 47, No. 4
2007年7月 Journal of Dalian University of Technology Jul. 2007

文章编号: 1000-8608(2007)04-0539-06

文化粒子群优化算法

王奕首, 艾景波, 史彦军, 滕弘飞*

(大连理工大学 机械工程学院, 辽宁 大连 116024)

摘要: 为了提高粒子群优化(PSO)算法的计算精度和计算效率,避免"早熟",给出了文化粒子群优化算法. 该算法模型将PSO纳入文化算法框架,组成基于PSO的主群体空间和知识空间,两空间具有各自群体并独立并行演化. 下层主群体空间定期贡献精英个体给上层知识空间,上层知识空间经演化后,定期贡献精英个体给下层主群体空间,于是形成"双演化双促进"机制,从而实现增加PSO的群体多样性. 在以卫星舱和印刷电路板布局设计为背景的算例中进行了数值验证,结果表明对于该算例,该方法的计算精度和计算效率比遗传算法、PSO算法高.

关键词: 演化计算; 粒子群优化; 文化算法; 布局设计

中图分类号: TP391.75 **文献标识码:** A

0 引 言

粒子群优化(particle swarm optimization, PSO)由Eberhart等[1]提出,源于对鸟群捕食的行为研究. 与遗传算法类似,PSO是一种基于群体演化的优化方法. PSO有个体数目少、计算简单、鲁棒性好等优点. 在各类多维连续空间优化问题、神经网络训练、模糊系统控制等领域上均取得了很好的应用效果[2~4]. 但PSO算法也存在易早熟的缺陷,如何提高PSO的计算精度和计算效率一直是人们追求的目标.

20世纪60年代,信息论和系统论的研究将"文化"看做人与环境相交互的系统,能够对系统中的人产生正向或负向的影响. 社会学研究人员认为文化是一种将个体的以往经验保存于其中的知识库,新的个体可以在知识库中学到其没有直接经历的经验知识. 受这些想法的启发,Reynolds[5]于1994年提出文化算法(cultural algorithm, CA),近年引起人们关注[6~8],但目前国内研究尚少.

文化算法将算法演化看做是在微观和宏观两个层面上的继承过程: 在微观层面,演化群体进行迭代求解形成知识信息;在宏观层面,知识群体保存上述知识信息,并通过与微观层面的交流,对微观层面的继续演化迭代进行指导. 在文化算法中称前者为主群体空间,称后者为知识空间,其主要思想就是保存群体可接受的知识,抛弃不接受的知识. 所以,应用文化算法解决全局优化问题,就是用可接受的全局最优点的信息来引导微观群体的演化.

本文将PSO算法纳入文化算法的模型,既利用PSO算法的快速演化能力,又利用文化算法模型中的遗传操作增加PSO群体多样性,构成文化粒子群优化算法(cultural-based PSO, CBPSO),用以求解布局设计问题.

1 文化算法

文化算法提供了一种显性的机制来获取、保存和整合微观演化群体寻优求解的知识和经验,而传统的演化计算技术只隐性地表示和保存知识信息.

文化算法是模拟文化演化的计算模型,图1给出文化算法的模型[7]. 主群体空间是算法运行进行问题求解的主空间,通过演化操作和性能评价进行自身的迭代求解;主群体空间不断产生知识信息通过接受操作保存到知识空间,知识空间

收稿日期: 2005-08-22; 修回日期: 2006-04-03.
基金项目: 国家自然科学基金资助项目(50335040; 50575031; 60674078); 国家"八六三"计划资助项目(2006AA04Z109).
作者简介: 王奕首(1978-), 男, 博士生; 滕弘飞*(1936-), 男, 教授, 博士生导师, E-mail: tenghf@dlut.edu.cn.
?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

通过自身的演化操作进行更新,并通过影响操作对主群体空间的进一步演化进行指导.

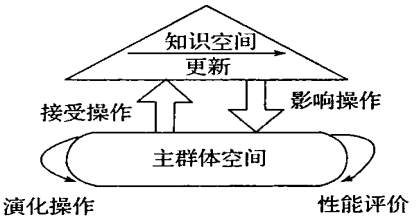


图 1 文化算法模型

Fig. 1 Cultural algorithms model

文化算法具有如下特性: (1) 支持不同算法的混合问题求解,本文方法以粒子群优化求解为主体,结合知识空间中的遗传演化操作,两者相互促进,完成整个求解过程; (2) 双重演化继承: 在主群体空间和知识解空间分别继承其父代的信息; (3) 群体空间的演化是由知识空间中保存的知识进行引导; (4) 支持主群体空间和知识空间的层次结构,支持两个空间的自适应演化,不同空间的演化可以按不同的速度进行; (5)"文化"改变的不同方式可表达于一个模型之内. 文化算法流程图如图 2所示.

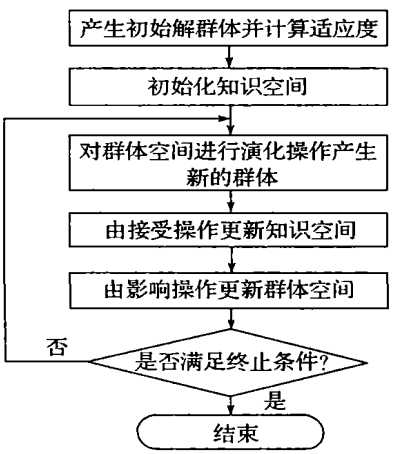


图 2 文化算法流程图

Fig. 2 Flowchart of cultural algorithms

文化算法可看做上下两层空间框架或模型. 下层的主群体空间和上层知识空间各自保存自己的群体,并各自独立并行演化. 一般而言,下层空间定期贡献精英个体给上层空间,上层空间不断进化自己的精英群体,反过来影响(或控制)下层空间群体,最终形成"双演化双促进"机制. 由此可见,这种文化算法模型或框架针对不同具体问题可以采用不同内涵的主群体空间和知识空间,具有广泛的应用前景.

2 基于粒子群优化的文化算法

本文基于文化算法的模型,建立粒子群的主群体空间(下层空间)与知识空间(上层空间),构成文化 PSO算法.

2.1 粒子群优化算法

粒子群算法的粒子 i的信息可以用 D维向量表示,位置表示为 $\boldsymbol{X}_i = (x_{i1} \quad x_{i2} \quad \cdots \quad x_{iD})^{\mathrm{T}}$,速度表示为 $\boldsymbol{V}_i = (v_{i1} \quad v_{i2} \quad \cdots \quad v_{iD})^{\mathrm{T}}$. 粒子 i根据式(1)和式(2)更新自己的速度和位置:

$$v_{id}^{k+1} = v_{id}^{k} + c_1 r_1 (p_{\mathrm{b},id}^{k} - x_{id}^{k}) + c_2 r_2 (g_{\mathrm{b},id}^{k} - x_{id}^{k}) \tag{1}$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \tag{2}$$

其中: $i = 1,\cdots,m$; $d = 1,\cdots,n$. m为粒子群中粒子的个数; n表示解向量的维数; v_{id}^{k}和 v_{id}^{k+1}分别表示第 i个粒子在 d维方向上的当前速度和修正后的速度; x_{id}^{k}和 x_{id}^{k+1}分别为第 i个粒子在 D维方向上的当前坐标和修正后的坐标; c_1、c_2是加速系数(或称学习因子); r_1、r_2是 [0, 1] 间的随机数; $p_{\mathrm{b},id}^{k}$是粒子 i在第 d维的个体极值点的位置; $g_{\mathrm{b},id}^{k}$是到第 k代为止所有粒子在迭代中找到的最好极值点的位置. 迭代中止条件根据具体问题一般选为最大迭代次数或(和)粒子群迄今为止搜索到的最优位置满足预定阈值.

文献 [9]对式(1)作了如下的改进:

$$v_{id}^{k+1} = w v_{id}^{k} + c_1 r_1 (p_{\mathrm{b},id}^{k} - x_{id}^{k}) + c_2 r_2 (g_{\mathrm{b},id}^{k} - x_{id}^{k}) \tag{3}$$

其中 w为非负数,称为惯性因子(inertia weight),或称为动量系数,控制着前一速度对当前速度的影响,w较大时,前一速度的影响较大,全局搜索能力比较强; w较小时,前一速度的影响较小,局部搜索能力比较强. 通过调整动量系数 w的大小,可以调整其大小跳出局部极小值.

本文采用的 PSO算法为式(3)的惯性因子 PSO算法,采用文献 [10]所推荐的参数,加速因子 $c_1 = c_2 = 1.49$, 惯性因子 $w = 0.729$.

2.2 文化粒子群优化算法

本文根据文化算法模型,结合粒子群优化,给出文化粒子群优化算法(CBPSO). 在 PSO演化过程中,粒子跟踪两个"目标": 一个是个体历史最优解(p_{b}),另一个是全局历史最优解(g_{b}),而全局历史最优解的代代相传,自然地形成全局最优解的轨迹,将这些最优解视为迭代搜索的全局知识

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

信息不断地保存并更新知识空间,并对知识解进行演化,同时对 PSO群体的进一步求解进行指导,通过 PSO空间 (主群体空间) 和知识空间的双重演化和互相影响,使求解具有更好的全局搜索能力.

文化粒子群优化算法构造如下:

(1) 主群体粒子群的设计

a. 确定编码方案. 采取十进制编码方案.

b. 确定适应度函数

$$Q(\boldsymbol{X}_i,\lambda) = \exp\left[F(\boldsymbol{X}_i) + \sum_{i=1}^{n} \lambda^{(i)} f_i(\boldsymbol{X}_i) u_i(f_i)\right] \tag{4}$$

$$u_i(f_i) = \begin{cases} 0, 若\ f_i(X_i) \leqslant 0 \\ 1, 若\ f_i(X_i) > 0 \end{cases};\ i \in I \tag{5}$$

式中: $F(\boldsymbol{X}_i)$ 是目标函数; $f_i(\boldsymbol{X}_i)$ 是约束条件; λ 是罚因子,适应度函数值越小,所得解越优.

(2) 知识空间的设计

a. 知识解个体形式. 采用与粒子群个体相同的形式编码,用以保存主群体的 g_b. 初始化时知识个体编码的基因位为零.

b. 知识解群体规模. 一般取为等于粒子群体规模的 40%.

c. 知识解群体自身的演化操作. 采用选择、交叉和变异操作: 交叉操作用两点交叉来使知识解群体共享知识信息; 当粒子群陷入局部最优时,以概率 p_m 进行变异操作以扩大搜索空间使算法具有更好的全局搜索能力,用赌轮选择保存较优个体.

(3) 接受操作

在主群体空间的粒子群演化过程中,每运行 *AcceptStep* 代时,用当前 g_b 替换知识空间中群体的最差个体. 本文 *AcceptStep* = 10.

(4) 影响操作

主群体空间的粒子群演化每运行 *AffectStep* 代时将知识空间群体中适应值较好的一部分 (一般取 5个) 个体替换粒子群中适应值较劣的同样数目个体,其计算公式为

$$AffectStep = N_1 + \frac{EndStep - k}{EndStep} \times N_2 \tag{6}$$

式中: *EndStep* 为粒子群预先设定的粒子群最大演化代数; k 为粒子群演化当前代数; N_1 和 N_2 为常数,这里分别取为 15和 100,这样在粒子群演化的初始阶段,知识解对其影响较小,使其能够保证快速演化. 在粒子群演化的后期,知识解对其影响逐渐加大,使其能够更多地接受知识空间的引导,同时扩大搜索空间,具备更好的全局搜索能力. CBPSO算法流程图如图 3所示.

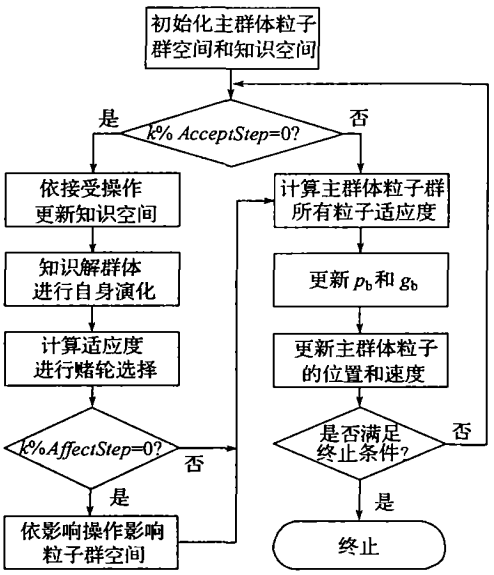


图 3　CBPSO算法流程图

Fig. 3　Flowchart of CBPSO

3　布局算例

算例 **1**　引自文献 [11]构造的一个已知最优解的布局设计问题,它的背景是返回式卫星舱布局设计问题. 在半径为 R = 75 mm的圆容器中放置 N = 9个圆形待布物,设各待布物的圆心坐标、半径、质量分别为 (x_i, y_i), r_i, m_i, 其中 r_i = 30 mm, i = 2, 3, 4, 5; r_i = 30($\sqrt{2}$ − 1) mm, i = 1, 6, 7, 8, 9,设 m_i 取值与 r_i 相等. 要求待布物在圆容器内摆放时满足下述条件: (1) 待布物之间以及待布物与容器之间不发生干涉; (2) 静不平衡量小于其许用值; (3) 待布物尽量向圆心聚集,即所有待布物的外包络圆半径 R_s^* 最小,具体数学模型见文献 [11].

本算例运行于 Celeron 850 MHz CPU, 256 MB内存的 PC上. 分别采用 GA(采用 GA算法库求解工具 GALib)、粒子群优化算法 (PSO)、本文方法 (CBPSO) 求解并进行比较. 本实验初始参数设定如下: 群体规模为 50, CBPSO中知识解群体演化采用两点交叉、基本位变异. 交叉、变异概率分别为 0.90、0.10. 初时群体随机产生,满足终止条件时结束 (运行时间达到 160 s). 对此算例分别用 3种算法进行 30次计算,表 1、2给出了最

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

好的一组布局结果;表 3给出 30次计算中成功所得结果的平均值,计算成功的判断条件为静不平衡量ΔJ小于 3.4 g·mm,且干涉量ΔS小于 0.25 mm^2.

由表 2可得,CBPSO与 GA法和 PSO法相比,所得最优解外包络圆半径分别减少 3.22%、0.06%;由表 3数据知:CBPSO与 GA和 PSO相比,所得平均外包络圆半径分别减少 4.30% 和增加了 0.20%;CBPSO与 GA法和 PSO法相比,成功率分别降低 6.7% 和提高了 3.3%.算例 1的最优布局结果如图 4所示.

表 1 算例 1 GA、PSO、CBPSO布局优化方案结果

Tab. 1 The optimal results of GA, PSO, CBPSO for Example 1

待布物序号	GA		PSO		CBPSO	
	x_i /mm	y_i /mm	x_i /mm	y_i /mm	x_i /mm	y_i /mm
1	50.71	-34.66	-42.33	42.89	-58.72	-13.33
2	8.78	-40.22	-0.24	-0.15	0.07	-0.30
3	-43.54	-10.52	-42.86	-41.87	-13.65	58.60
4	-9.72	43.73	43.28	41.93	58.70	13.40
5	44.01	8.33	42.27	-42.95	13.62	-58.60
6	-53.35	32.12	-42.68	0.55	-22.56	-36.10
7	-31.14	-53.79	-0.15	-42.68	-36.10	22.50
8	36.38	50.99	42.69	-0.50	22.43	36.20
9	-1.37	2.14	0.10	42.69	36.22	-22.50

表 2 算例 1优化布局性能指标

Tab. 2 Performance index of the optimal results for Example 1

计算方法	R_s^* /mm	ΔJ /(g·mm)	ΔS /mm^2
GA	75.063	0.869	1.012
PSO	72.688	0	0
CBPSO	72.645	0	0

注:静不平衡量 < 0.20 g·mm和干涉量 < 0.20 mm^2视为 0

表 3 算例 1平均优化布局性能指标

Tab. 3 Average performance index of all the solutions for Example 1

计算方法	R_s^* /mm	ΔJ /(g·mm)	ΔS /mm^2	成功率 /%
GA	77.021 7	0.545 6	0.670 0	16.7
PSO	73.565 1	0.019 4	0.087 8	6.7
CBPSO	73.711 3	0.001 2	0.044 2	10.0

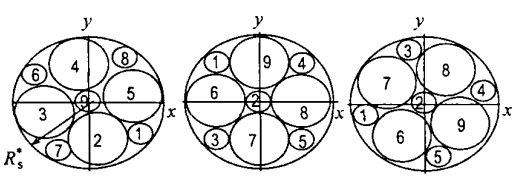


图 4 算例 1三种算法的布局优化图

Fig. 4 View of the layout results from GA, PSO, CBPSO for Example 1

算例 2 引自文献[12](文献[12]算例是在文献[13]算例基础上,给出考题数据),以印刷电路板为背景的布局问题.在二维空间放置 15个圆形待布物,待布物间的权值为 w_{ij}(表示待布物在位置空间上的亲密度),要求待布物摆放时满足下述要求:(1)待布物间无干涉;(2)具有较大权重值的待布物应相互聚集,即 $C=\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} d_{ij}w_{ij}$ 的数值最小,其中 d_{ij}为两待布物间的距离;(3)待布物的外包络矩形面积最小.数学模型可表述如下:

$$\begin{aligned} &\text{求 } \boldsymbol{X}=(x_i \quad y_i)^{\mathrm{T}}, i\in\{1,2,\cdots,15\} \\ &\min F(\boldsymbol{X}_i)=S+\lambda C \\ &\text{s.t. } \mathrm{int}A_i\cap \mathrm{int}A_j=\varnothing \\ &\qquad i\neq j, i,j\in(1,2,\cdots,n) \end{aligned} \tag{7}$$

式中:$\mathrm{int}A_i$表示内部;S为外包络矩形的面积;λ为参数 C相对于 S的权重,本文取 $\lambda=0.8$;A_i、A_j为待布物 i、j的面积.待布物间的权重矩阵见文献[12].

用 GA、PSO、CBPSO分别对此算例进行 30次计算,求得的最好一组布局结果见表 4和表 5,表 6给出计算成功所得结果的平均值.计算成功的判断条件为干涉量小于 0.35 mm^2.本实验初始参数设定同算例 1,初始群体随机产生,满足终止条件时结束(运行时间达到 300 s).计算结果表明,利用 CBPSO比 GA和 PSO所得最优解的 f_{sc}($f_{sc}=S+0.8C$)分别减小 1.20% 和 7.05%;所得平均 f_{sc}相对 GA、PSO分别增加 2.71% 和减

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

小 10.18%; CBPSO与 GA和 PSO相比,成功率降低 10%. 算例 2的最优布局结果如图 5所示.

表 4　算例 2 GA PSO CBPSO布局优化方案结果

Tab. 4　The optimal results of GA, PSO, CBPSO for Example 2

待布物序号	r /mm	GA		PSO		CBPSO	
		x_i /mm	y_i /mm	x_i /mm	y_i /mm	x_i /mm	y_i /mm
1	12	14.93	-4.81	-13.05	-22.76	17.14	-14.64
2	3	9.17	20.59	-27.83	15.68	-8.92	11.37
3	12	23.83	17.47	-12.77	14.72	3.65	24.08
4	3	8.93	-18.48	-8.90	-0.87	1.18	-12.58
5	9	35.60	-1.49	3.43	0.95	-10.16	-16.65
6	10	-12.60	-29.35	24.87	-24.84	22.06	-36.10
7	7	5.25	11.46	1.59	27.19	-15.82	4.13
8	8	-7.97	4.58	20.18	-2.80	-29.70	-1.49
9	4	15.54	-20.77	-7.07	-7.61	-19.15	-7.16
10	12	31.56	-22.09	18.00	17.20	22.32	8.93
11	6	2.81	-25.05	-17.69	-3.58	-25.32	-15.16
12	10	-20.14	17.83	-36.48	5.99	1.24	0.65
13	9	-21.53	-5.65	-31.38	-12.29	-16.80	-33.45
14	9	-4.72	-12.07	7.73	-16.52	1.88	-30.45
15	10	-2.23	26.72	-35.29	26.65	-18.37	21.86

表 5　算例 2优化布局性能指标

Tab. 5　Performance index of the optimum results for Example 2

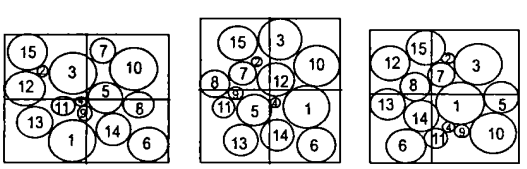
计算方法	S /mm²	C /mm	ΔS /mm²
GA	5 674.67	76 676.0	0.541 2
PSO	5 816.14	81 767.5	0
CBPSO	5 918.97	75 360.6	0

注: 干涉量小于 0.20 mm²视为 0

表 6　算例 2平均优化布局性能指标

Tab. 6　Average performance index of all the solutions for Example 2

计算方法	S /mm²	C /mm	ΔS /mm²	成功率 /%
GA	6 405.29	82 794.5	0.835 1	93.3
PSO	6 622.04	93 725.6	0.282 8	93.3
CBPSO	6 561.99	83 416.9	1.138 2	83.3



(a) GA布局　(b) PSO布局　(c) CBPSO布局

图 5　算例 2三种算法的布局优化图

Fig. 5　The layout results from GA, PSO and CBPSO for Example 2

4　结　语

以求解布局设计问题为应用背景,研究文化粒子群优化算法. 该方法将粒子群优化(PSO)纳入文化算法(cultural algorithm)模型,构造了基于粒子群的主群体空间和知识空间两空间的双重演化(在知识空间中本文采用遗传算法演化操作),主群体空间和知识空间具有各自的群体并各自独立并行演化,下层主群体空间定期贡献精英个体给上层知识空间群体,上层空间不断进化自己的群体,反过来影响(或控制)下层空间群体,最终形成并行计算的"双演化双促进"机制,从而实现提高 PSO的计算精度和计算效率的目的. 这是本文算法与现有混合粒子群和遗传算法在本质上的区别. 算例计算结果表明,与 GA PSO相比,本文算法的计算精度和计算效率较高. 它可能适用于解决较复杂的或规模较大的函数优化问题. 另外,这种文化算法框架针对不同具体问题可以采用不同内涵的主群体空间和知识空间,具有较广泛的应用前景.

参考文献:

[1] EBERHART R, KENNEDY J. A new optimizer using particle swarm theory [C]// Proceedings of

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

the Sixth International Symposium on Micro Machine and Human Science. Nagoya: IEEE, 1995: 39–43

[2] SHI Y, EBERHART R. Fuzzy adaptive particle swarm optimization [C]// **Proceedings of the 2001 Congress on Evolutionary Computation**. Seoul: IEEE Press, 2001: 101–106

[3] HU X H, EBERHART R C. Multi-objective optimization using dynamic neighborhood particle swarm optimization [C]// **Proceedings of the 2002 Congress on Evolutionary Computation**. Honolulu: IEEE, 2002: 1677–1681

[4] JUANG C F. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design [J]. **IEEE Trans Syst Man and Cybernetics: Part B-Cybernetics**, 2004, **34**(2): 997–1006

[5] REYNOLDS R G, MICHALEWICZ Z, CAVARETTA M. Using cultural algorithms for constraint handling in GENOCOP [C]// **Proceedings of the Fourth Annual Conference on Evolutionary Programming**. Cambridge: MIT Press, 1995: 298–305

[6] STERNBERG M, REYNOLDS R G. Using cultural algorithms to support re-engineering of rule-based expert systems in dynamic performance environments: a case study in fraud detection [J]. **IEEE Trans on Evolut Comput**, 1997, **1**(4): 225–243

[7] JIN X D, REYNOLDS R G. Using knowledge-based system with hierarchical architecture to guide the search of evolutionary computation [J]. **Int J on Artif Intell Tools**, 2000, **9**(1): 27–44

[8] PENG B, REYNOLDS R G. Cultural algorithms: knowledge learning in dynamic environments [C]// **Proceedings of Congress on Evolutionary Computation**. Portland: IEEE, 2004: 1751–1758

[9] SHI Y, EBERHART R. A modified particle swarm optimizer [C]// **Proceedings of IEEE World Congress on Computational Intelligence**. Anchorage: IEEE, 1998: 69–73

[10] CLERC M, KENNEDY J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space [J]. **IEEE Trans on Evolut Comput**, 2002, **6**(1): 58–73

[11] 钱志勤, 滕弘飞, 孙治国. 人机交互的遗传算法及其在约束布局优化中的应用 [J]. 计算机学报, 2001, **24**(5): 553–559

[12] 李广强, 滕弘飞, 霍军周. 并行混合免疫算法及其在布局设计中的应用 [J]. 机械工程学报, 2003, **39**(6): 79–85

[13] 康立山, 谢 云, 尤矢勇, 等. 非数值并行算法 [M]. 北京: 科学出版社, 1994

Cultural-based particle swarm optimization algorithm

WANG Yi-shou, AI Jing-bo, SHI Yan-jun, TENG Hong-fei*

(School of Mech. Eng., Dalian Univ. of Technol., Dalian 116024, China)

Abstract A cultural-based particle swarm optimization (CBPSO) algorithm is proposed to improve the computational accuracy and efficiency of PSO and avoid premature. This algorithm model consists of a PSO-based main population space and a knowledge space, which respectively has its own population to evolve independently and parallel. The lower level main population space (PSO population) contributes elite individuals to the upper level space (knowledge population) periodically, and the upper level space continually evolves these elite individuals and then contributes elite individuals to the lower level space. The mechanism of dual evolution and dual promotion improves the population diversity, and avoids premature. Two examples originated from the layout design of satellite module and integrated circuit show that CBPSO exhibits better computational efficiency and accuracy than GA and PSO.

Key words evolutionary computation; particle swarm optimization; cultural algorithm; layout design

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net