

文章编号: 1001-0920(2008)05-0511-04

## 一种克服粒子群早熟的混合优化算法

吴敏<sup>1</sup>, 丁雷<sup>1,2</sup>, 曹卫华<sup>1</sup>, 徐辰华<sup>1</sup>

(1. 中南大学 信息科学与工程学院, 长沙 410083; 2. 吉首大学 物理科学与信息工程学院, 湖南 吉首 416000)

**摘要:** 针对粒子群优化算法在寻优时容易出现早熟现象, 提出在粒子群收敛停滞时, 从种群中随机选择粒子进行共轭梯度法计算, 通过引入共轭梯度算法计算的信息来影响粒子速度的更新, 以保持群体的活性, 从而打破群体信息陷入局部最优的状况. 不同于传统的粒子群算法, 该算法有机地结合了粒子群的全局搜索能力和共轭梯度法的强大局部搜索能力, 从而在一定程度上有效地克服了粒子群早熟的缺点. 仿真计算结果表明, 该改进粒子群的方法对于不同维数的非线性函数具有很好的寻优效果.

**关键词:** 粒子群优化; 共轭梯度法; 全局最优

中图分类号: TP18

文献标识码: A

## A kind of hybrid optimization algorithm with prevention of premature convergence of particle swarm

WU Min<sup>1</sup>, DING Lei<sup>1,2</sup>, CAO Wei-hua<sup>1</sup>, XU Chen-hua<sup>1</sup>

(1. School of Information Science and Engineering, Central South University, Changsha 410083, China; 2. School of Physics Science and Information Engineering, Jishou University, Jishou 416000, China. Correspondent: WU Min, E-mail: min@mail.csu.edu.cn)

**Abstract:** To prevent the problem of premature convergence frequently appeared in the particle swarm optimization (PSO), a method is proposed, which selects particles stochastically to perform the conjugate gradient algorithm when the PSO stagnates. The calculation information of the conjugate gradient algorithm is employed to affect the update of the particle speed so as to maintain the particle activation and avoid local optima. Unlike the existing PSO algorithms, the presented method integrates the global search ability of the PSO and the powerful local search ability of the conjugate gradient algorithm. Thus, the problem of premature convergence of the PSO algorithm is prevented. Simulation results show that the method has better performance for different dimensioned nonlinear functions.

**Key words:** Particle swarm optimization; Conjugate gradient algorithm; Global optimum

### 1 引言

优化问题是工业设计中经常遇到的问题, 为了解决各种优化问题, 人们提出了许多优化算法. 1995 年 Eberhart 和 Kennedy<sup>[1,2]</sup> 提出了一种新算法: 粒子群优化 (PSO) 算法. 该算法一经提出, 立刻引起了优化及演化计算等领域学者们的广泛关注.

粒子群优化算法是一类基于群智能随机优化算法, 因受人工生命研究结果的启发, 其优点在于简单, 同时又有深刻的智能背景, 既适合科学研究, 又适合工程应用, 目前已广泛应用于函数优化、神经网络训练、模糊系统控制等多个领域<sup>[3-5]</sup>. PSO 根据个体的适配信息进行搜索, 不受函数约束条件的限制,

具有全局优化能力. 但对于多维复杂问题, 往往会出现早熟现象, 容易陷入局部最优, 无法保证其收敛到全局最优. 共轭梯度法是一个经典的非线性寻优算法, 具有线性收敛的特点, 有二次终结性, 其局部搜索能力很强<sup>[6,7]</sup>.

针对粒子群算法的早熟现象, 本文研究如何充分利用粒子群的全局搜索能力和共轭梯度法的局部搜索能力, 达到克服粒子群算法早熟所带来的影响, 得到最佳的优化值.

### 2 粒子群优化算法和共轭梯度法分析

粒子群优化算法 (PSO) 是一种进化计算技术, 源于对鸟群捕食的行为研究. 其基本思想是通过群

收稿日期: 2007-01-13; 修回日期: 2007-05-18.

基金项目: 国家杰出青年科学基金项目 (60425310).

作者简介: 吴敏 (1963—), 男, 广东化州人, 教授, 博士生导师, 从事过程控制、鲁棒控制等研究; 丁雷 (1972—), 男, 湖南临湘人, 博士生, 从事过程控制、智能控制等研究.

体中个体之间的协作和信息共享来寻找最优解,即初始化为一群随机粒子(随机解),然后通过迭代找到最优解。PSO的优势在于算法简单,容易实现,并且没有许多参数的调节。

PSO算法在每次的迭代中,粒子通过跟踪两个“极值”(pbest, gbest)来更新自己。其中:pbest表示个体最优值,gbest表示全局最优值。在找到这两个最优值后,粒子通过下式来更新自己的速度和位置<sup>[1]</sup>:

$$\begin{cases} v_i = v_i + c_1 \cdot \text{rand}() \cdot (\text{pbest}_i - x_i) + \\ c_2 \cdot \text{rand}() \cdot (\text{gbest}_i - x_i), \\ x_i = x_i + v_i. \end{cases} \quad (1)$$

其中: $v_i$ 是粒子的速度; $\text{rand}()$ 是介于(0,1)之间的随机数; $x_i$ 是粒子的当前位置; $c_1$ 和 $c_2$ 是学习因子,一般取2.0。 $v_i$ 的取值最大为 $v_{\max}$ ,决定当前位置与最好位置之间区域的分辨率(或精度)。如果 $v_{\max}$ 太高,微粒可能会飞过最优解;如果 $v_{\max}$ 太小,则微粒不能在局部最优区间之外进行足够的探索,容易造成早熟,导致陷入局部最优值。Shi等人<sup>[8]</sup>对式(1)进行了修正,引入惯性权重因子 $w$ ,即

$$\begin{cases} v_i = w \cdot v_i + c_1 \cdot \text{rand}() \cdot (\text{pbest}_i - x_i) + \\ c_2 \cdot \text{rand}() \cdot (\text{gbest}_i - x_i), \\ x_i = x_i + v_i. \end{cases} \quad (2)$$

Shi与Eberhart经实验证明:修改过的粒子群优化算法对优化大多数的Benchmark方程较原始的算法有了明显改进。惯性权重的选择有固定权重和时变权重两种。固定权重的选择就是选择某一常数作为权重值;在优化过程中不变;而时变权重为

$$w = w_0 - w_1 \cdot \text{Step} / \text{Gen}, \quad (3)$$

即惯性权值线性递减法,选定某一个变化范围,在迭代过程中按照某一个递减率线性减小。其中: $w_0$ ,  $w_1$ 为初始惯性权重;Step为粒子群当前迭代的次数;Gen为预设的总的迭代次数。 $w$ 值越大,其全局搜索能力越强;反之, $w$ 值越小,其局部搜索能力越强<sup>[8,9]</sup>。

共轭梯度法是一个经典的非线性无约束条件优化算法,具有快速收敛的特点。其基本思想是利用已知点处的梯度构造一组共轭方向,并据此搜索目标函数极值<sup>[7]</sup>。设初始点 $X^{(1)}$ ,取 $d_1 = -\nabla f(X^{(1)})$ ,其中 $X^{(1)}$ 表示向量矢量, $d_1$ 表示梯度负方向。设 $k=1$ ,已得到 $k$ 个相互共轭的方向 $d^{(1)}, d^{(2)}, \dots, d^{(k)}$ ,以及由 $X^{(1)}$ 开始依次沿上述方向精确一维搜索得到点 $X^{(2)}, \dots, X^{(k)}, X^{(k+1)}$ ,即

$$X^{(i+1)} = X^{(i)} + \alpha_i d^{(i)}, \quad (4)$$

其中 $\alpha_i$ 表示通过某种线性搜索计算出的步长。通过

精确一维搜索保证方向导数为0,即

$$\nabla f^T(X^{(i+1)}) d^{(i)} = 0, \quad i = 1, 2, \dots, k. \quad (5)$$

在 $X^{(i+1)}$ 点构造新方向 $d^{(k+1)}$ 为 $-\nabla f(X^{(k+1)})$ 与 $d^{(1)}, d^{(2)}, \dots, d^{(k)}$ 的组合。由于 $d^{(1)}, d^{(2)}, \dots, d^{(k)}$ 相互共轭,有

$$\begin{cases} d^{(k+1)} = -\nabla f^T(X^{(k+1)}) + \sum_k d^{(k)}, \\ \alpha_k = \frac{\nabla f^T(X^{(k+1)}) \nabla f(X^{(k+1)})}{\nabla f^T(X^{(k)}) \nabla f(X^{(k)})}. \end{cases} \quad (6)$$

由以上分析可知,粒子群算法的全局收敛能力很强,但很容易因为早熟而陷入局部最优解;而共轭梯度法因为采用的是梯度极值为0的判断标准,其局部收敛能力非常强。因此,将粒子群全局收敛能力和共轭梯度局部收敛能力有机融合起来会有效地克服粒子群早熟现象。

### 3 混合粒子群优化算法

不同于文献[10]提出的梯度加速算法,共轭梯度法不只是在负梯度方向进行一次线性搜索获得加速步长,它获得的结果也是局部极小值。因此,本文提出一种基于共轭梯度法的混合粒子群算法,它在群体最优信息陷入停滞时,通过引入共轭梯度法计算的信息来影响粒子速度的更新,以保持群体的活性,减小群体陷入局优的可能性。

整个算法是利用粒子群进行全局搜索,当粒子群算法稳定下来后,对粒子群中每个粒子产生一个随机值。如果这个随机值小于设定的值,则对这个粒子进行共轭梯度法计算,并保存所计算的粒子速度值,作为下一代迭代中该粒子的速度值。整个算法的步骤如下:

Step1: 初始化权重、学习参数和粒子的位置、速度。

Step2: 计算当代最佳粒子。

Step3: 计算全局历史最佳粒子。

Step4: 计算个体历史最佳粒子。

Step5: 判断收敛是否停滞,如果没有停滞,则进行标准粒子群算法,计算完后转Step6;否则,对每个粒子产生一个随机数,如果随机数小于所设定的值 $P$ ,则对该粒子开始进行共轭梯度法,即Step7。

Step6: 判断算法是否满足终止条件,满足则停止计算;否则转Step2。

Step7: 将粒子群优化算法计算得到的该粒子位置值赋给 $X^{(1)}$ ,且 $d^{(1)} = -\nabla f(X^{(1)})$ , $k=1$ 。

Step8: 判断 $\nabla f(X^{(k)}) < ?$ 小于则转Step5,对下个粒子进行判断;否则进行下一步。

Step9: 求 $\min f(X^{(k)} + \alpha_k d^{(k)})$ , $X^{(k+1)} = X^{(k)} + \alpha_k d^{(k)}$ 。

Step10: 判断 $K = n?$ 如果相等则令 $X^{(1)} =$

$X^{(n+1)}$ , 且  $d^{(1)} = -\nabla f(X^{(1)})$ ,  $K = 1$ , 转 Step8; 如果不相等则执行下一步.

Step11:  $d^{(k+1)} = -\nabla f(x^{(k)}) + kd^{(k)}$ ,  $k = k + 1$ , 转 Step8.

### 4 仿真实验

从以上粒子群算法分析可知,  $w$  值影响局部搜索能力, 目前普遍采用的是线性递减权值法<sup>[8,9]</sup>, 学习因子  $c_1$  和  $c_2$  一般固定为  $2.0^{[1]}$ . 选取 5 个函数进行计算<sup>[11-14]</sup>, 如表 1 所示. 其中: 函数 1 为 schaffer's  $f_6$ , 有无数个极小值, 全局最小点是  $(0, 0)$ , 最小值为 0; 函数 2 为 Sphere, 是单峰函数, 全局最小点是  $(0, 0)$ , 最小值是 0; 函数 3 为 Rosenbrock, 是单峰函数, 在极小值附近沿曲线  $x_2 = x_1^2$  有陡峭的峡谷, 很容易陷入局部值, 它的全局最小点是  $(1, 1)$ , 最小值是 0; 函数 4 为 Rastrigrin, 有很多局部极小点, 全局最小点是  $(0, 0, \dots, 0)$ , 最小值是 0; 函数 5 为 Griewank, 有多个局部极小点, 全局最小点是  $(0, 0, \dots, 0)$ , 最小值是 0.

按照文献[10]提供的 3 种不同惯性权重因子  $w$  的取值范围, 种群数 30, 迭代次数 4 000 次, 分别对以上 5 个函数进行基于共轭梯度法的混合粒子群算法. 为便于比较, 不同权重范围下的标准粒子群算法

和梯度加速粒子群算法的结果都用文献[10]所提供的平均最优值来表示. 本文所提出的混合算法在标准粒子群算法迭代到 3 980 次时, 引入共轭梯度法, 对每个粒子产生一个随机数, 如果小于 0.1, 则进行共轭梯度算法, 结果如表 2 所示. 其中: SPSO 为标准粒子群算法, TPSO 为带有梯度加速的混合粒子群算法, GPSO 为基于共轭梯度法的混合粒子群算法.

从表 2 中可知, 对于不同的惯性权重, 基于共轭梯度法的混合粒子群算法不但比标准粒子群算法在很大程度上提高了寻优效果, 而且整体上比带有梯度加速的混合粒子群算法具有更好的优化效果, 尤其对一些典型的有多局部极小值或易陷入局部值的非线性函数, 具有更好的寻优效果.

基于共轭梯度法的混合粒子群算法寻优的仿真结果如图 1 ~ 图 5 所示, 为突出引入共轭梯度法所产生的克服粒子群早熟的效果, 从混合粒子群算法迭代到 3 901 次时开始连续选取 100 个寻优值. 从仿真结果(菱形线对应的  $w$  为 0.9 ~ 0.2, 粗实线对应的  $w$  为 0.5 ~ 0.2, 虚线对应的  $w$  为 0.9 ~ 0.5)可以看出, 在粒子群算法收敛变慢或停滞时, 引入共轭梯度算法可在较大程度上改善粒子群寻优效果, 从

表 1 函数表达式及其取值范围

序号	函数名	表 达 式	维数	取值范围
1	Schaffer's $f_6$	$f(x_1, x_2) = 0.5 + \frac{(\sin \sqrt{x_1^2 + x_2^2} - 0.5)}{(1.0 + 0.001 \cdot (x_1^2 + x_2^2))^2}$	2	$[-100, 100]$
2	Sphere	$f(x_1, x_2, \dots, x_{30}) = \sum_{i=1}^{30} x_i^2$	30	$[-100, 100]$
3	Rosenbrock	$f(x_1, x_2, \dots, x_{30}) = \sum_{i=1}^{29} (100 \cdot (x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	30	$[-30, 30]$
4	Rastrigrin	$f(x_1, x_2, \dots, x_{30}) = \sum_{i=1}^{30} (x_i^2 - 10 \cdot \cos(2 \cdot x_i) + 10)$	30	$[-5.12, 5.12]$
5	Griewank	$f(x_1, x_2, \dots, x_{30}) = \frac{1}{4000} \cdot \sum_{i=1}^{30} x_i^2 - \prod_{i=1}^{30} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	$[-600, 600]$

表 2 不同权重下的 3 种优化方法的结果比较

惯性权重	函数 1			函数 2			函数 3			函数 4			函数 5		
	SPSO 平均量最优值	TPSO 平均最优值	GPSO 最优值	SPSO 平均量最优值	TPSO 平均最优值	GPSO 最优值	SPSO 平均量最优值	TPSO 平均最优值	GPSO 最优值	SPSO 平均量最优值	TPSO 平均最优值	GPSO 最优值	SPSO 平均量最优值	TPSO 平均最优值	GPSO 最优值
0.9 ~ 0.2	1.94e <sup>-4</sup>	0	0	7.04e <sup>-19</sup>	3.62e <sup>-35</sup>	4.4e <sup>-42</sup>	53	21.2	2.0e <sup>-11</sup>	36.08	10.0	0	0.013 9	1.15e <sup>-16</sup>	0
0.5 ~ 0.2	1.94e <sup>-4</sup>	0	0	7.04e <sup>-19</sup>	3.62e <sup>-35</sup>	3.8e <sup>-42</sup>	53	21.2	3.9e <sup>-11</sup>	36.08	10.0	0	0.013 9	1.15e <sup>-16</sup>	0
0.9 ~ 0.5	1.94e <sup>-4</sup>	0	0	7.04e <sup>-19</sup>	3.62e <sup>-35</sup>	3.9e <sup>-41</sup>	53	21.2	2.3e <sup>-11</sup>	36.08	10.0	0	0.013 9	1.15e <sup>-16</sup>	0

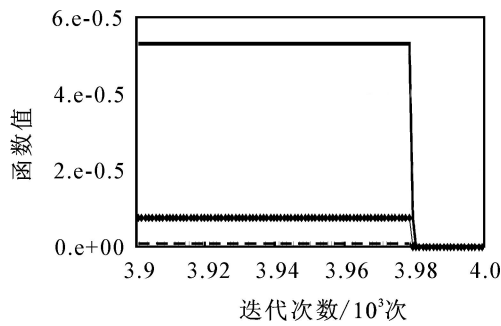


图1 函数1的基于共轭梯度法的混合粒子群算法寻优

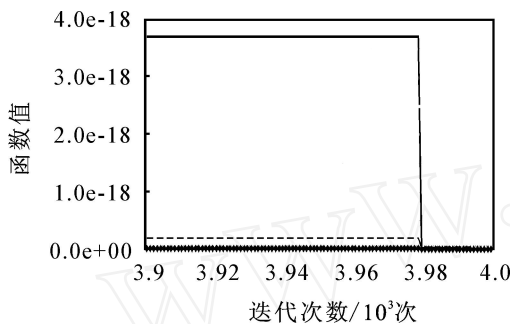


图2 函数2的基于共轭梯度法的混合粒子群算法寻优

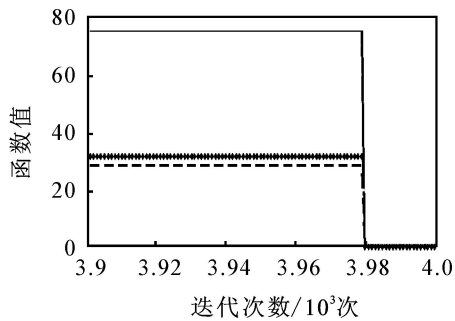


图3 函数3的基于共轭梯度法的混合粒子群算法寻优

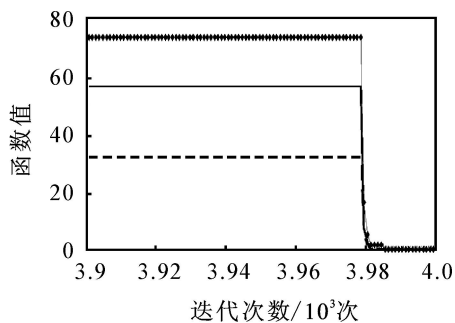


图4 函数4的基于共轭梯度法的混合粒子群算法寻优

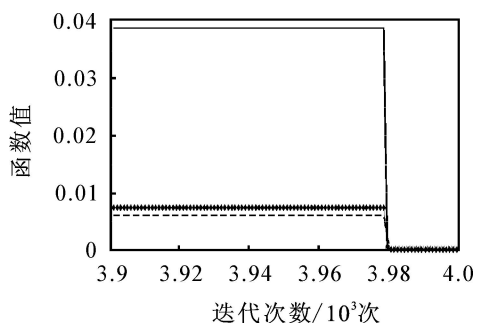


图5 函数5的基于共轭梯度法的混合粒子群算法寻优

而在一定程度上能起到克服因粒子群算法早熟带来的陷入局部最优的现象,获得一个较理想的寻优值。

## 5 结 论

本文将粒子群算法和共轭梯度算法有机结合起来共同寻优,充分利用了粒子群算法的全局搜索能力和共轭梯度算法的局部搜索能力。在不同的惯性权重取值范围内,对5个常用的测试函数进行了基于共轭梯度法的混合粒子群算法的仿真实验,并分别与文献[10]采用标准粒子群算法和带有梯度加速的粒子群算法所计算的结果进行了对比。结果表明,基于共轭梯度法的混合粒子群算法对于不同维数、有多个局部极值或易陷入局部值的非线性函数,能得到比标准粒子群算法和带有梯度加速的混合粒子群算法更好的优化效果。

## 参考文献(References)

- [1] Kennedy J, Eberhart R C. Particle swarm optimization [C]. Proc of the IEEE Int Conf on Neural Networks. Perth, 1995: 1942-1948.
- [2] Eberhart R C, Kennedy J. A new optimizer using particles swarm theory [C]. Proc of the 6th Int Symposium on Micro Machine and Human Science. Nagoya, 1995: 39-43.
- [3] Kwok N M, Liu D K, Dissanayake G. Evolutionary computing based mobile robot localization [J]. Engineering Applications of Artificial Intelligence, 2006, 19 (8): 857-868.
- [4] Yin Peng-yeng, Wang Jing-yu. A particle swarm optimization approach to the nonlinear resource allocation problem [J]. Applied Mathematics and Computation, 2006, 183 (1): 232-242.
- [5] Du Ji-xiang, Huang De-shuang, Zhang Guo-jun, et al. A novel full structure optimization algorithm for radial basis probabilistic neural networks [J]. Short Communication Neurocomputing, 2006, 70 (1): 592-596.
- [6] 潘美芹, 贺国强. 基于共轭梯度法的函数优化混合遗传算法[J]. 山东科技大学学报, 2000, 19(4): 10-13. (Pan Mei-qin, He Guo-qiang. A hybrid genetic algorithm of function optimization based on conjugate gradient algorithm [J]. J of Shandong University of Science and Technology, 2000, 19(4): 10-13.)
- [7] 吴祈宗. 运筹学与最优化方法[M]. 北京: 机械工业出版社, 2003. (Wu Qi-zhong. Operations research and optimization methods[M]. Beijing: China Machine Press, 2003.)
- [8] Shi Y, Eberhart R C. A modified particle swarm optimizer [C]. Proc of the IEEE Congress on Evolutionary Computation. Anchorage, 1998: 69-73.

(下转第519页)

预测函数控制算法, 在实现控制方案时, 不需事先知道系统模型的结构和参数, 在线计算量小, 能根据对象的变化自动辨识模型, 调整控制规律, 实现自适应控制. 本文给出了保证闭环系统稳定的条件, 依据该条件提出的遗传优化参数补偿方案, 扩大了该控制算法的适用范围, 提高了控制品质. 仿真实验表明, 本文所提方法适用范围广, 鲁棒性强.

### 参考文献(References)

- [1] Richalet J, Arber C. Predictive functional control: Application to fast and accurate robots[C]. Proc of the 10th IFAC World Congress. Munich, 1987: 251-258.
- [2] Kuntze H B, Jacubasch A. On the application of a new method for fast and robust[C]. 1988 IEEE Int Conf on Robotics and Automation. Leqacv, 1988: 1574-1580.
- [3] 张泉灵, 王树青. 基于 ARMAX 模型自适应预测函数控制[J]. 信息与控制, 2000, 29(5): 431-436.  
(Zhang Quan-ling, Wang Shu-qing. Adaptive predictive function control based on ARMAX model [J]. Information and Control, 2000, 29(5): 431-436.)
- [4] 张泉灵, 王树青. 基于神经网络的非线性预测函数控制[J]. 浙江大学学报, 2001, 35(5): 497-500.  
(Zhang Quan-ling, Wang Shu-qing. Nonlinear predictive function control based on neural network [J]. J of Zhejiang University, 2001, 35(5): 497-500.)
- [5] 刘贺平, 张兰玲, 孙一康. 基于多层局部回归神经网络的多变量非线性系统预测控制[J]. 控制理论与应用, 2001, 18(2): 298-300.  
(Liu He-ping, Zhang Lan-ling, Sun Yi-kang. Predictive control of multivariable nonlinear system based on multilayer local recurrent neural networks [J]. Control Theory and Applications, 2001, 18(2): 298-300.)
- [6] Christos C, Zerovos, Dumout G A. Deterministic adaptive control based on Laguerre series representation [J]. Int J Control, 1988, 48(6): 2333-2359.
- [7] Vivas A, Poinet P, Pierrot F. Predictive functional control for a parallel robot [C]. Int Conf on Intelligent Robots and Systems. Las Vegas, 2003, 3: 2785-2790.
- [8] 李嗣福, 刘勇. 一种基于 Laguerre 函数模型的预测控制算法[J]. 中国科技大学学报, 1999, 29(3): 281-288.  
(Li Si-fu, Liu Yong. An adaptive predictive control algorithm based on the Laguerre function approximate model [J]. J of China University of Science and Technology, 1999, 29(3): 281-288.)
- [9] Zhang Quan-ling. Nonlinear predictive functional control using Hammerstein models [J]. J of Zhejiang University, 2002, 36(2): 119-22.
- [10] 韩璞, 王东风, 王国玉. 多模型预测函数控制及其应用研究[J]. 控制与决策, 2003, 18(3): 375-381.  
(Han Pu, Wang Dong-feng, Wang Guo-yu. Multi-model predictive functional control and its application [J]. Control and Decision, 2003, 18(3): 375-381.)
- [11] 胡品慧, 袁璞. 关于状态反馈预测控制系统的极点配置[J]. 控制与决策, 2001, 16(6): 910-913.  
(Hu Pin-hui, Yuan Pu. On pole placement of state feedback model predictive control systems [J]. Control and Decision, 2001, 16(6): 910-913.)
- [9] Parsopoulos K E, Plagianakos V P, Magoulas G D, et al. Improving particle swarm optimizer by function "stretching" [C]. Advances in Convex Analysis and Global Optimization. Netherlands: Kluwer Academic Publishers, 2001: 445-457.
- [10] 王俊伟, 汪定伟. 一种带有梯度加速的粒子群算法[J]. 控制与决策, 2004, 19(11): 1298-1300.  
(Wang Jun-wei, Wang Ding-wei. Particle swarm optimization algorithm with gradient acceleration [J]. Control and Decision, 2004, 19(11): 1298-1300.)
- [11] Clerc M, Kennedy J. The particle swarm: Explosion stability and convergence in a multi-dimensional complex space [J]. IEEE Trans on Evolutionary Computation, 2002, 6(1): 58-73.
- [12] Trelea I C. The particle swarm optimization algorithm: Convergence analysis and parameter selection [J]. Information Processing Letters, 2003, 85(6): 317-325.
- [13] Eberhart R C, Shi Y. Comparing inertia weights and constriction factors in particle swarm optimization [C]. Proc of the IEEE Congress on Evolutionary Computation. La Jolla, 2000: 84-88.
- [14] Van den Bergh F. An analysis of particle swarm optimizers [D]. South Africa: University of Pretoria, 2002.

(上接第 514 页)