

文章编号: 1001-0920(2006)06-0636-05

## 粒子群优化算法的收敛性分析及其混沌改进算法

刘洪波, 王秀坤, 谭国真

(大连理工大学 计算机系, 辽宁 大连 116023)

**摘要:** 分析了粒子群优化算法的收敛性, 指出它在满足收敛性的前提下种群多样性趋于减小, 粒子将会因速度降低而失去继续搜索可行解的能力; 提出混沌粒子群优化算法, 该算法在满足收敛性的条件下利用混沌特性提高种群的多样性和粒子搜索的遍历性, 将混沌状态引入到优化变量使粒子获得持续搜索的能力. 实验结果表明混沌粒子群优化算法是有效的, 与粒子群优化算法、遗传算法、模拟退火相比, 特别是针对高维、多模态函数优化问题取得了明显改善.

**关键词:** 粒子群优化算法; 混沌; 多模态函数优化问题; 遗传算法; 模拟退火算法

**中图分类号:** TP301.6 **文献标识码:** A

## Convergence Analysis of Particle Swarm Optimization and Its Improved Algorithm Based on Chaos

L I U H ong-bo, W A N G X iu-kun, T A N G uo-zhen

(Department of Computer, Dalian University of Technology, Dalian 116023, China Correspondent: L I U Hong-bo, E-mail: lhb@dlut.edu.cn)

**Abstract:** The particle swarm optimization (PSO) algorithm is analyzed. Its premature convergence is due to the decrease of velocity of particles in search space that leads to a total implosion and ultimately fitness stagnation of the swarm. A chaotic particle swarm optimization (CPSO) algorithm is introduced to overcome the problem of premature convergence. CPSO uses the properties of ergodicity, stochastic property, and regularity of chaos to lead particles' exploration. This enable the swarm system to have the ability of "sustainable development." Simulation results show that CPSO prevents premature convergence effectively and is better than PSO, genetic algorithm and simulated annealing on some benchmark function optimization problems.

**Key words:** Particle swarm optimization; Chaos; Multimodal function optimization problem; Genetic algorithms; Simulated annealing

### 1 引言

粒子群优化算法(PSO)是一种新兴的群智优化算法<sup>[1]</sup>, 其思想来源于人工生命和演化计算理论, 是对鸟群觅食过程中的迁徙和聚集的模拟. 由简单个体组成的群落以及个体之间的互动行为模拟搜索最优解, 收敛速度快. 相对遗传算法、模拟退火等算法而言, PSO 更简单有效<sup>[2]</sup>, 并已得到众多学者的重视和研究<sup>[3-5]</sup>, 许多实际应用非常成功<sup>[6-8]</sup>. 但它存在早熟问题, 大量的研究只集中在讨论粒子的轨迹

对算法收敛性所产生的影响<sup>[9,10]</sup>, 以及惯性因子的更新和种群拓扑结构的改进<sup>[11,12]</sup>, 而深入分析粒子的速度对算法收敛性的影响并给出其具体的证明并不多见.

本文从理论上分析粒子的速度对算法收敛性的影响, 并给出了证明. 同时提出了混沌粒子群优化算法, 该算法在满足收敛性的条件下利用混沌的伪随机性, 对初始值敏感性和遍历性引导粒子群中的粒子搜索, 提高种群的多样性和粒子搜索的遍历性, 将

收稿日期: 2005-05-08; 修回日期: 2005-08-26

基金项目: 国家自然科学基金项目(60373095); 国家973计划项目(2100CCA 00700); 教育部科学基金项目(KP0302).

作者简介: 刘洪波(1971—), 男, 武汉人, 博士, 讲师, 从事进化计算、神经网络等研究; 王秀坤(1945—), 女, 辽宁辽阳人, 教授, 博士生导师, 从事进化计算、机器学习等研究.

混沌状态引入到优化变量使粒子获得继续搜索的能力

### 2 粒子群优化算法

粒子群优化算法寻优的过程被看成是鸟群协同觅食, 其目标就是全局最优解 其算法描述如下:

Step 1: 设优化问题的定义域为  $[-r, r]$ , 维度为  $d$ , 令粒子最大速度  $v_{\max} = r$   $t = 0$  时, 对  $n$  个  $d$  维粒子的位置和速度进行初始化, 令  $v_{i,j}(0) = U(-1, 1)v_{\max}$ ,  $x_{i,j}(0) = U(-1, 1)^* r$ , 其中  $U(-1, 1)$  是  $[-1, 1]$  区间正态分布的随机数

Step 2: 若满足终止条件, 则输出结果  $x^*$  和  $f(x^*)$  并结束算法; 否则, 转 Step 3

Step 3:  $t = t + 1$ , 实施最优保存策略, 即

$$x_i^{\#}(t) = \arg \min_{1 \leq i \leq n} (f(x_i^{\#}(t-1)), f(x_i(t)));$$

$$x^*(t) = \arg \min_{1 \leq i \leq n} (f(x^*(t-1)),$$

$$f(x_1(t)), \dots, f(x_n(t))).$$

Step 4: 针对每个粒子的每一维度执行个性算子、自意识算子、群意识算子联合操作, 即

$$v_{i,j}(t+1) = w v_{i,j}(t) + \mathcal{Q}_{i,j}^1(t) (x_{i,j}^{\#}(t) - x_{i,j}(t)) + \mathcal{Q}_{i,j}^2(t) (x_j^*(t) - x_{i,j}(t)), \quad (1)$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1). \quad (2)$$

其中:  $1 \leq i \leq n, 1 \leq j \leq d, w$  通常是小于或等于 1 的正常数,  $\mathcal{Q}_{i,j}^k = c_k U_k(0, 1), U_k(0, 1)$  是  $[0, 1]$  区间正态分布的随机数,  $k = 1, 2$ , 转 Step 2

粒子群在  $d$  维解空间搜索, 粒子处在一定的位置上, 由一个适应函数  $f(x)$  决定它的适应值, 并有一个飞行速度直接影响它每次飞翔的距离 初始时一群粒子随机地处在解空间中的不同位置, 然后根据式(1)和(2)来更新自己的速度和位置, 其中  $v_{i,j}(t)$  表示  $t$  时粒子  $i$  的第  $j$  维的速度,  $x_{i,j}(t)$  表示  $t$  时第  $i$  个粒子的第  $j$  维的位置,  $x_i^{\#}(t)$  表示  $t$  时粒子  $i$  的当前极值,  $x^*(t)$  表示  $t$  时粒子群的当前极值 在式(1)的右边共有三项: 第 1 项是粒子上一次的速度  $v_{i,j}(t)$  与惯性因子  $w$  的乘积; 第 2 项是粒子自身行为差异比较; 第 3 项是粒子群行为差异比较; 后两项合称为粒子的“意识”

### 3 粒子群优化算法的收敛性分析

文献[10]对粒子群优化算法中粒子轨迹收敛进行了深入分析, 本文在此基础上进一步分析粒子的速度对算法收敛性的影响

假定  $\mathcal{Q}_{i,j}^1(t), x_{i,j}^{\#}(t), x_j^*(t)$  是常数, 即  $\mathcal{Q}_{i,j}^1 = \mathcal{Q}_{i,j}^1(t), x_{i,j}^{\#} = x_{i,j}^{\#}(t), x_j^* = x_j^*(t)$ . 由式(1)和(2)消去速度相关的参数, 则

$$x_{i,j}(t+1) =$$

$$(1 + w - \mathcal{Q}_{i,j}^1 - \mathcal{Q}_{i,j}^2) x_{i,j}(t) - w x_{i,j}(t-1) + \mathcal{Q}_{i,j}^1 x_{i,j}^{\#} + \mathcal{Q}_{i,j}^2 x_j^*. \quad (3)$$

令

$$A = \begin{bmatrix} 1 + w - \mathcal{Q}_{i,j}^1 - \mathcal{Q}_{i,j}^2 & -w & \mathcal{Q}_{i,j}^1 x_{i,j}^{\#} + \mathcal{Q}_{i,j}^2 x_j^* \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

则式(3)的齐次矩阵形式为

$$\begin{bmatrix} x_{i,j}(t+1) \\ x_{i,j}(t) \\ 1 \end{bmatrix} = A \cdot \begin{bmatrix} x_{i,j}(t) \\ x_{i,j}(t-1) \\ 1 \end{bmatrix}. \quad (4)$$

式(4)中系数矩阵的特征多项式为

$$(1 - \lambda)(w - \lambda(1 + w - \mathcal{Q}_{i,j}^1 - \mathcal{Q}_{i,j}^2) + \lambda^2). \quad (5)$$

该特征多项式存在三个根值, 即

$$\lambda = 1,$$

$$\alpha_{i,j} = \frac{1 + w - \mathcal{Q}_{i,j}^1 - \mathcal{Q}_{i,j}^2 + \mathcal{Y}_{i,j}}{2},$$

$$\beta_{i,j} = \frac{1 + w - \mathcal{Q}_{i,j}^1 - \mathcal{Q}_{i,j}^2 - \mathcal{Y}_{i,j}}{2}.$$

其中:  $\mathcal{Y}_{i,j} = \sqrt{(1 + w - \mathcal{Q}_{i,j}^1 - \mathcal{Q}_{i,j}^2)^2 - 4w}$ . 因为  $\alpha, \beta$  均为式(4)系数矩阵的特征多项式的根, 所以式(3)可写为

$$x_{i,j}(t) = k_1 + k_2 \alpha_{i,j} + k_3 \beta_{i,j}. \quad (6)$$

同理可得

$$v_{i,j}(t) = h_1 \alpha_{i,j} + h_2 \beta_{i,j}. \quad (7)$$

如果  $(1 + w - \mathcal{Q}_{i,j}^1 - \mathcal{Q}_{i,j}^2)^2 \geq 4w$ , 则  $\mathcal{Y}, \alpha, \beta$  为实数; 如果  $(1 + w - \mathcal{Q}_{i,j}^1 - \mathcal{Q}_{i,j}^2)^2 < 4w$ , 则  $\mathcal{Y}, \alpha, \beta$  为复数 为了讨论方便, 如果  $\mathcal{Y}, \alpha, \beta$  为实数,  $\alpha, \beta$  分别表示  $\alpha, \beta$  的绝对值; 如果  $\mathcal{Y}, \alpha, \beta$  为复数,  $\alpha, \beta$  分别表示  $\alpha, \beta$  的模值

**定理 1** (粒子群优化算法收敛定理) 粒子群优化算法是收敛的, 当且仅当  $\max(\alpha, \beta) < 1$ .

**证明** 不失一般性, 忽略具体维度标识, 对式(6)求极限有

$$\lim_{t \rightarrow +\infty} x(t) = \lim_{t \rightarrow +\infty} (k_1 + k_2 \alpha + k_3 \beta) = k_1 + k_2 \lim_{t \rightarrow +\infty} \alpha + k_3 \lim_{t \rightarrow +\infty} \beta.$$

当  $\alpha > 1$  或  $\beta > 1$  时,  $\lim_{t \rightarrow +\infty} x(t)$  不存在, 即粒子的轨迹是发散的;

当  $\alpha < 1$  且  $\beta < 1$  时, 有  $\lim_{t \rightarrow +\infty} x(t) = k_1$ , 粒子的轨迹是收敛的;

当  $\max(\alpha, \beta) = 1$ , 有  $\lim_{t \rightarrow +\infty} x(t) = k_1 + k_2$ , 或  $\lim_{t \rightarrow +\infty} x(t) = k_1 + k_3$ ,

粒子的轨迹也是收敛的

由于粒子的位置表示解空间搜索的解,所以命题成立

该定理同时还确切地给出粒子群优化算法的参数选择范围 对于式(6),代入初始值和随机量 $Q^j$ 的期望值 $c_k/2$ ,可得

$$w = (c_1 + c_2)/2 - 1 \quad (8)$$

当 $w = 1, c_1 = 2, c_2 = 2$ 时,式(8)取等号

**推论1(粒子速度收敛性)** 如果粒子群优化算法是收敛的,粒子群中的粒子速度或者不断减小到0,或者一直以初始化速度迭代到算法结束

**证明** 不失一般性,忽略具体维度标识,对式(7)求极限有

$$\lim_{t \rightarrow +\infty} v(t) = \lim_{t \rightarrow +\infty} (h_1\alpha + h_2\beta).$$

如果粒子群优化算法是收敛的,那么有

$$\max(\alpha, \beta) = 1$$

1) 当 $\max(\alpha, \beta) = 1$ ,有 $\lim_{t \rightarrow +\infty} v(t) = 0$ ;

2) 当 $\max(\alpha, \beta) = 1$ ,有 $\lim_{t \rightarrow +\infty} v(t) = h_1$ ,或 $\lim_{t \rightarrow +\infty} v(t) = h_2$ ,或 $\lim_{t \rightarrow +\infty} v(t) = h_1 + h_2$ ,由式(7)的初始化条件可知粒子一直以初始化速度迭代到算法结束

粒子群优化算法收敛性和粒子速度收敛性的几何解释是:如果一个粒子当前的位置、该粒子的当前最优值和粒子群的当前最优值三者一致,该粒子会因为它以前的速度和惯性因子不为零而远离最佳位置导致算法不能收敛;如果以前的速度非常接近零,粒子一旦赶上了粒子群的当前最佳粒子,种群多样性就慢慢丧失,所有的粒子将会集聚到相同位置并停止移动,粒子群优化出现停滞状态,却仍没有搜索到满意解,这种情况大多导致早熟;如果粒子速度一直以初始化速度迭代到算法结束,相当于自意识算子和群意识算子失效,不利于全局最小值的搜索,算法的适应性将显著降低 为了避免早熟,提高算法的适应性,本文在粒子群优化算法中引入混沌,提出了基于混沌的改进算法,该算法在满足粒子群优化算法收敛定理条件下,粒子速度既不会减小到0又不会一直以初始化速度迭代到算法结束

## 4 混沌粒子群优化算法

### 4.1 混沌的特性

混沌是一种普遍的非线性现象,具有伪随机性、对初始值的敏感性和遍历性等精致的内在规律<sup>[13,14]</sup>。一种具有非线性反馈能力的混沌神经元映射<sup>[15]</sup>如下:

$$z_{n+1} = \eta_n - 2 \tanh(\gamma z_n) \exp(-3z_n^2) \quad (9)$$

其中: $z_n$ 是神经元的 $n$ 时内部状态, $n = 0, 1, 2, \dots; \eta$

是衰减因子( $0 < \eta < 1$ ); $\gamma$ 是比例因子,通常 $\gamma = 5$  该映射的点迹分布如图1所示,其迭代次数为2000,分布开阔而且比较均衡,有利于优化搜索的遍历性

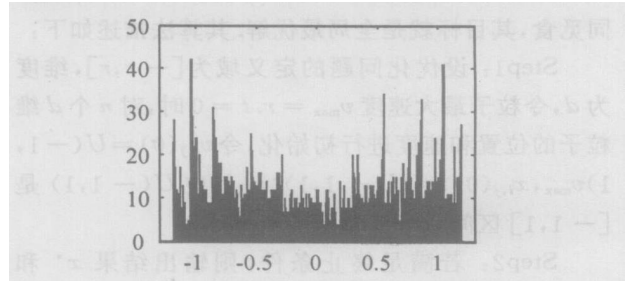


图1 混沌神经元映射的点迹分布

### 4.2 混沌粒子群优化算法基本思路与算法设计

混沌粒子群优化算法的基本思想体现在两个方面:1)采用混沌系列初始化粒子的位置和速度,既不改变粒子群优化算法初始化时所具有的随机性本质,又利用混沌提高了种群的多样性和粒子搜索的遍历性;2)将混沌状态引入到优化变量使粒子获得持续搜索的能力

**混沌初始化:**对式(9)中的 $z_n$ 按取值范围分别赋予 $n \times d$ 个初始值 $z_{0,i}(i = 0, 1, \dots, d)$ ,混沌后对粒子初始位置和速度进行初始化处理

**粒子速度控制:**依据混沌PSO的基本思想,粒子迭代过程中的速度更新公式由式(1)变为

$$v_{i,j}(t+1) = w v_{i,j}(t) + Q_{i,j}^j(t) (x_{i,j}^*(t) - x_{i,j}(t)) + Q_{i,j}^j(t) (x_j^*(t) - x_{i,j}(t)) \quad (10)$$

其中

$$v_{i,j}(t) = \begin{cases} v_{i,j}(t), & \text{flag}(t) = 0; \\ U(-1, 1) * \rho * v, & \text{flag}(t) = 1; \end{cases}$$

$$\text{flag}(t) = \begin{cases} 1, & f(x^*(t)) = f(x^*(t-1)) = \dots = f(x^*(t-9)); \\ 0, & f(x^*(t-1)) > f(x^*(t)); \end{cases}$$

$$\hat{v} = r^* z_n / 1.2; \rho = f(x^*) - f_{\text{objective}}$$

其中: $z_n$ 是由式(9)产生的新的混沌序列, $f_{\text{objective}}$ 为优化目标值或满意解

## 5 实验及分析

优化算法的性能比较通常是基于一些称为基准函数<sup>[16]</sup>的典型问题展开的,本文就4种有代表性基准函数对所讨论的算法进行测试和分析,这4种基准函数为:

$f_1$ : Rosenbrock's function

$$f(x) = \sum_{i=1}^n (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2),$$

$$x \in [-50, 50], x^* = (1, 1, \dots, 1), f(x^*) = 0$$

$f_2$ : Rastrigin's function

$$f(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10),$$

$$x \in [-5.12, 5.12],$$

$$x^* = (0, 0, \dots, 0), f(x^*) = 0$$

$f_3$ : Griewank's function

$$f(x) = \frac{1}{4000} \sum_{i=1}^n (x_i)^2 - \prod_{i=1}^n \cos(x_i / \sqrt{i}) + 1,$$

$$x \in [-300, 300],$$

$$x^* = (0, 0, \dots, 0), f(x^*) = 0$$

$f_4$ : Ackley's function

$$f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e,$$

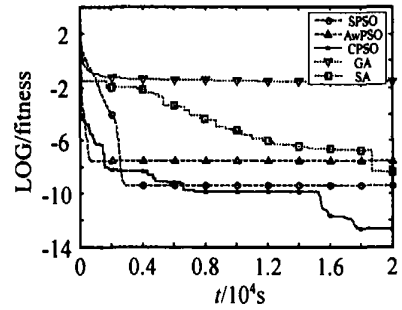
$$x \in [-32, 32], x^* = (0, 0, \dots, 0), f(x^*) = 0$$

其中: Rosenbrock 函数是连续单模态函数; Rastrigin 函数、Griewank 函数和 Ackley 函数是连续多模态函数, 有多个(局部)极小值, 一个全局极小值, 而且在到达全局极小值的过程中存在大规模的曲率变化

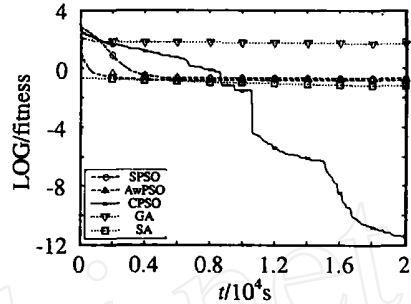
本文将所提出的CPSO 与标准 PSO (SPSO)<sup>[1]</sup>、Aw PSO<sup>[11]</sup>、遗传算法(GA)、模拟退火(SA) 进行性能比较 实验中 SPSO 和 CPSO 的  $c_1 = c_2 = 1.49, w = 0.7$ , 满足式(8), 粒子群的种群规模为 20; 遗传算法的种群规模为 20, 交叉率为 0.8, 变异率为 0.02; 模拟退火算法的初始温度为 50, 温度调整操作次数为 20, 温度约简因子为 0.85. 所有算法的优化目标是函数值最小化, 针对每个函数不同维度运行 10 次, 计算其平均结果和标准方差

限于篇幅, 只给出了优化 Rastrigin 函数的结果对比曲线(如图 2 所示), 其他结果见表 1. CPSO 能

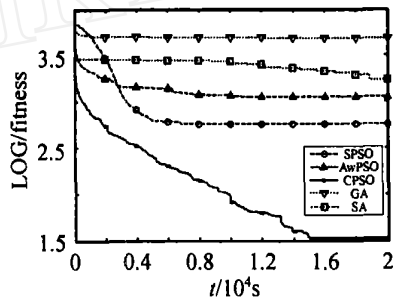
到达的最好值明显优于其他算法, 模拟退火在维度较低时也具有很好的性能 对于 200 维以内的函数,



(a) 30-D Rastrigin's function



(b) 100-D Rastrigin's function



(c) 200-D Rastrigin's function

图 2 优化 Rastrigin 函数结果对比

表 1 优化基准函数的结果比较

算法	$f_1$		$f_2$		$f_3$		$f_4$	
	30 维	100 维	30 维	100 维	30 维	100 维	30 维	100 维
SPSO	19.773 2 ± 11.462 0	189.656 7 ± 51.345 5	5.365 5e-10 ± 7.744 3e-10	0.942 1 ± 0.452 6	0.315 7 ± 0.017 7	1.337 4 ± 1.508 5	2.799 2e-14 ± 3.883 0e-14	4.126 2 ± 1.034 0
AwPSO	21.113 9 ± 5.062 1	216.258 0 ± 54.334 3	3.750 2e-08 ± 2.451 3e-08	0.837 5 ± 1.821 2	0.072 0 ± 3.235 0e-12	3.092 3 ± 2.519 5	0.717 9 ± 0.001 4	0.898 2 ± 0.017 2
CPSO	0.032 0 ± 0.051 2	97.628 6 ± 26.386 1	3.511 8e-13 ± 9.701 0e-13	6.079 0e-12 ± 1.935 2e-12	0.007 8 ± 0.011 0	0.010 9 ± 0.023 5	1.749 2e-14 ± 4.810 3e-15	1.645 7e-04 ± 1.565 9e-04
GA	222.951 0 ± 26.487 4	7.273 0e+ 003 ± 459.104 4	0.487 4 ± 0.957 5	81.536 5 ± 17.771 9	6.846 0 ± 0.606 0	179.596 6 ± 7.390 8	1.743 7 ± 0.042 7	2.357 0 ± 0.007 9
SA	29.055 2 ± 4.829 1	138.323 3 ± 38.102 9	8.972 0e-09 ± 9.432e-09	0.745 3 ± 14.463 3	0.319 3 ± 1.788 0	31.427 0 ± 11.465 6	0.660 6 ± 0.065 7	1.016 7 ± 0.053 2

CPSO 均能以极高的精度逼近全局最小点,但对于维度超过200的目标函数而言,粒子搜索到的可行解在全局最小点附近增加缓慢,虽然相对其他算法有较大的改善,但CPSO的结果仍不理想,成功率也只有95%。这也是优化问题所共同面临的“维度灾”问题

粒子群中的粒子因在解空间飞行所到达的位置不同而搜索到不同的可行解,即因粒子的位置不同而表现出的多样性称为种群多样性,它是评价粒子群搜索可行解的能力的重要尺度。它由下式给出:

$$\text{Diversity} = \frac{1}{n} \sqrt{\frac{1}{d} \sum_{j=1}^d (x_{i,j} - \bar{x}_j)^2} \quad (11)$$

其中:  $\bar{x}$  为粒子群的中心位置, 它的第  $j$  维为  $\bar{x}_j$ 。

图3是PSO与CPSO的种群多样性比较曲线,图中PSO的种群多样性在迭代的过程中一直呈下降趋势,1000代时保持在一个很低的水平;而CPSO的种群多样性在迭代的过程中存在3个不同的阶段:1)初始阶段呈波动式下降;2)在500代左右出现反弹呈波动式增加;3)1200代后一直保持在一个较高的水平。

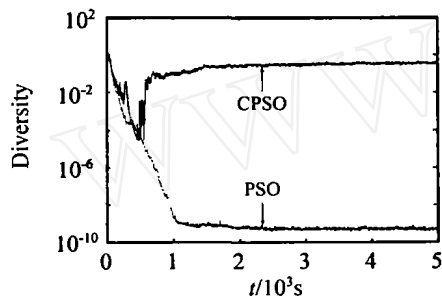


图3 种群多样性比较

## 6 结论与展望

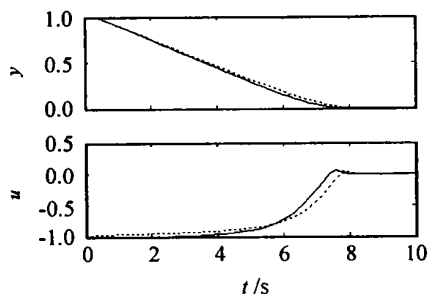
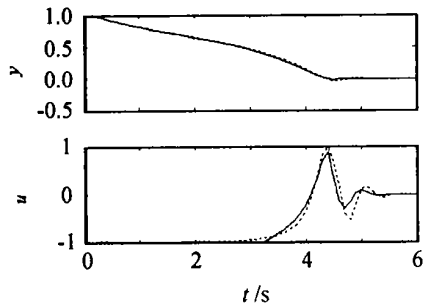
本文从理论上分析了粒子群优化算法的收敛性,指出粒子群优化算法满足收敛性的前提下种群多样性就可能会慢慢减小,粒子将会因速度降低而失去继续搜索可行解的能力,粒子群优化出现停滞状态;这种情况大多导致早熟。为此,提出混沌粒子群优化算法,该算法满足收敛性的条件下利用混沌的伪随机性、对初始值的敏感性和遍历性来引导粒子搜索,提高种群的多样性和粒子搜索的遍历性,将混沌状态引入到优化变量使因速度降低而失去搜索可行解能力的粒子获得继续搜索的能力。对所提出的混沌粒子群优化算法进行了实验,结果表明混沌粒子群优化算法是有效的,与其他优化算法相比,特别是针对高维、多模态函数优化问题取得了明显改善。

值得注意的是,本文的混沌载入系数是根据粒子当前最佳适应值与目标值进行自适应调整的,更灵活的调整策略需要进一步研究。另外,通向混沌还有更丰富的混沌发展模式,在本文工作的基础上引入其他混沌发展模式,进一步改进粒子群优化算法处理优化过程中的“维度灾”问题值得研究。

## 参考文献(References)

- [1] Kennedy J, Eberhart R. *Swarm Intelligence* [M]. San Francisco: Morgan Kaufmann Publishers, 2001.
- [2] Boeringer D W, Werner D H. Particle Swarm Optimization Versus Genetic Algorithms for Phased Array Synthesis [J]. *IEEE Trans on Antennas and Propagation*, 2004, 52(3): 771-779.
- [3] Parsopoulos K E, Vrahatis M N. Recent Approaches to Global Optimization Problems Through Particle Swarm Optimization [J]. *Natural Computing*, 2002, 1(2-3): 235-306.
- [4] 彭宇, 彭喜元, 刘兆庆. 微粒群算法参数效能的统计分析 [J]. *电子学报*, 2004, 32(2): 209-213.  
(Peng Y, Peng X Y, Liu Z Q. Statistic Analysis on Parameter Efficiency of Particle Swarm Optimization [J]. *Acta Electronica Sinica*, 2004, 32(2): 209-213.)
- [5] Liu H B, Li B, Wang X K, et al. Survival Density Particle Swarm Optimisation for Neural Network Training [A]. *Lecture Notes in Computer Science* [C]. Springer-Verlag, 2004, 3173: 332-337.
- [6] Sousa T, Silva A, Neves A. Particle Swarm Based Data Mining Algorithms for Classification Tasks [J]. *Parallel Computing*, 2004, 30(5-6): 767-783.
- [7] 谭国真, 柳亚玲, 高文. 随机时间依赖网络的K期望最短路径 [J]. *计算机学报*, 2003, 26(3): 323-331.  
(Tan G Z, Liu Y L, Gao W. K Expected Shortest Path in Stochastic and Time-Dependent Network [J]. *Chinese J of Computers*, 2003, 26(3): 323-331.)
- [8] 李宁, 刘飞, 孙德宝. 基于带变异算子粒子群优化算法的约束布局优化研究 [J]. *计算机学报*, 2004, 27(7): 897-903.  
(Li N, Liu F, Sun D B. A Study on the Particle Swarm Optimization with Mutation Operator Constrained Layout Optimization [J]. *Chinese J of Computers*, 2004, 27(7): 897-903.)
- [9] Clerc M, Kennedy J. The Particle Swarm -Explosion, Stability, and Convergence in a Multidimensional Complex Space [J]. *IEEE Trans on Evolutionary Computation*, 2002, 6(1): 58-73.
- [10] Frans van den Bergh. *An Analysis of Particle Swarm Optimizers* [D]. Pretoria: University of Pretoria, 2001: 78-85.

(下转第645页)

图 4  $\Delta = 0$  时的输出和输入响应图 5  $\Delta = \sin t$  时的输出和输入响应

算法的控制结果 从图中不难看出, 本文提出的算法具有更好的控制性能

## 6 结 论

本文针对饱和约束系统提出了一种鲁棒模型预测控制算法, 分别考虑了多面体不确定性和结构反馈不确定性 算法采用了无穷时域的最坏性能指标以及带有饱和特性的状态反馈控制结构, 其控制律可以通过求解一个 LMI 优化问题得到, 闭环系统的鲁棒稳定性取决于 LMI 优化问题在初始时刻的可行性 仿真结果表明与原有采用线性反馈控制结构

的算法相比, 提出的算法能够更加充分地利用输入的约束范围, 提高控制性能

## 参考文献 (References)

- [1] Lee J H, Yu Z H. Worst-case Formulation of Model Predictive Control for Systems with Bounded Parameters[J]. *Automatica*, 1997, 33(5): 763-781.
- [2] Kothare M V, Balakrishnan V, Morari M. Robust Constrained Model Predictive Control Using Linear Matrix Inequalities [J]. *Automatica*, 1996, 32 (10): 1361-1379.
- [3] Casavola A, Giannelli M, Mosca E. Minmax Predictive Control Strategies for Input Saturated Polytopic Uncertain Systems[J]. *Automatica*, 2000, 36 (1): 125-133.
- [4] Schuurmans J, Rossiter J A. Robust Predictive Control Using Tight Sets of Predicted States[J]. *IEEE Proc of Control Theory and Application*, 2000, 147(1): 13-18.
- [5] Wu F. LM Based Robust Model Predictive Control and Its Application to an Industrial CSTR Problem [J]. *J of Process Control*, 2001, 11(6): 649-659.
- [6] Lu Y H, Arkun Y. Quasi-Min-Max MPC Algorithm for LPV Systems [J]. *Automatica*, 2000, 36 (4): 527-540.
- [7] Cuzzola F A, Geromel J C, Morari M. An Improved Approach for Constrained Robust Model Predictive Control[J]. *Automatica*, 2002, 38(7): 1183-1189.
- [8] Hu T, Lin Z, Chen B M. Analysis and Design for Discrete-time Linear Systems Subject to Actuator Saturation [A]. *Proc of the 40th IEEE Conf on Decision and Control* [C]. Orlando, 2001: 4675-4680.

(上接第 640 页)

- [11] Mahfouf M, Chen M Y, Linkens D A. Adaptive Weighted Swarm Optimization for Multiobjective Optimal Design of Alloy Steels [A]. *Lecture Notes in Computer Science* [C]. Springer-Verlag, 2004, 3242: 762-771.
- [12] Frans van den Bergh, Engelbrecht A P. A Cooperative Approach to Particle Swarm Optimization [J]. *IEEE Trans on Evolutionary Computation*, 2004, 8(3): 225-239.
- [13] Ott E, Grebogi C, Yorke J A. Controlling chaos [J]. *Physical Review Letters*, 1990, 64(11): 1196-1199.
- [14] 单梁, 强浩, 李军, 等. 基于 Tent 映射的混沌优化算法

[J]. *控制与决策*, 2005, 20(2): 179-182

(Shan L, Qiang H, Li J, et al. Chaotic Optimization Algorithm Based on Tent Map [J]. *Control and Decision*, 2005, 20(2): 179-182.)

- [15] Ji M J, Tang H W. Application of Chaos in Simulated Annealing [J]. *Chaos, Solitons and Fractals*, 2004, 21 (4): 933-941.

- [16] 王凌. *智能优化算法及其应用* [M]. 北京: 清华大学出版社, 2001.

(Wang L. *Intelligent Optimization Algorithms with Applications* [M]. Beijing: Tsinghua University Press, 2001.)