

# 混沌粒子群优化算法<sup>\*</sup>

高 鹰<sup>1,2</sup> 谢胜利<sup>1</sup>

(华南理工大学电子与信息学院 广州 510641)<sup>1</sup>

(广州大学信息机电学院计算机科学与技术系 广州 510405)<sup>2</sup>

**摘 要** 粒子群优化算法是一种新的随机全局优化进化算法。本文把混沌寻优思想引入到粒子群优化算法中,这种方法利用混沌运动的随机性、遍历性和规律性等特性首先对当前粒子群体中的最优粒子进行混沌寻优,然后把混沌寻优的结果随机替换粒子群体中的一个粒子。通过这种处理使得粒子群体的进化速度加快,从而改善了粒子群优化算法摆脱局部极值点的能力,提高了算法的收敛速度和精度。仿真结果表明混沌粒子群优化算法的收敛性能明显优于粒子群优化算法。

**关键词** 粒子群优化算法,混沌寻优,优化

## Chaos Particle Swarm Optimization Algorithm

GAO Ying<sup>1,2</sup> XIE Sheng-Li<sup>1</sup>

(College of Electronic & Information Engineering, South China University of Technology, Guangzhou 510641)<sup>1</sup>

(Dept. of Computer Science and Technology, Guangzhou University, Guangzhou 510405)<sup>2</sup>

**Abstract** Particle swarm optimization is a new stochastic global optimization evolutionary algorithm. In this paper, the chaotic search is embedded into original particle swarm optimizers. Based on the ergodicity, stochastic property and regularity of chaos, a new superior individual is reproduced by chaotic searching on the current global best individual, and a stochastic selected individual from the current "population" is replaced by the new superior individual. The particle swarm optimization embedded chaotic search quickens the evolution process, and improves the abilities of seeking the global excellent result and convergence speed and accuracy. The experiment results demonstrate that the proposed algorithms are superior to original particle swarm optimization algorithms.

**Keywords** Particle swarm optimization, Chaotic search, Optimization

## 1 引言

Kennedy 和 Eberhart<sup>[1,2]</sup>于 1995 年提出的粒子群优化算法是一种基于群智能的随机优化进化算法。同遗传算法类似,是一种基于群体的具有全局寻优能力的优化工具。但它没有遗传算法中用的交叉以及变异等复杂的遗传操作,其优势在于简单、易于实现同时又有深刻的智能背景,既适合科学研究,又特别适合工程应用。自从粒子群优化算法提出以来,一直受到计算智能等领域的研究人员的广泛关注,在短短的几年时间里取得了丰硕的研究成果<sup>[2~14]</sup>。然而, Kennedy 等人提出的粒子群优化算法亦有其不足:易陷入局部极值点,进化后期收敛速度慢,精度较差等。为了克服粒子群优化算法的这些不足,研究人员提出了许多改进的粒子群优化算法,如:1998 年 Shi Y 提出的带惯性因子的粒子群优化算法<sup>[3]</sup>,随后于 2001 年给出的模糊自适应粒子群优化算法<sup>[4]</sup>;为控制粒子的飞行速度, Clerc M 于 1999 年提出的带约束因子的粒子群优化算法<sup>[5]</sup>;借鉴遗传算法的思想, Angeline P. (1998) 提出了杂交粒子群优化算法<sup>[6]</sup>,之后, Lovbjerg M (2001) 给出了具有繁殖和子群的粒子群优化算法<sup>[7]</sup>, 2003 年 Natsuki 又给出的具有高斯变异的粒子群优化算法<sup>[8]</sup>;为使粒子群优化算法更易

跳出局部极值点, Van (2001) 给出了协同粒子群优化算法<sup>[9,10]</sup>;文<sup>[11,12]</sup>(1997,2000)对粒子群优化算法进行了扩展而提出了离散粒子群优化算法等。这些算法从不同方面对粒子群优化算法进行了改进,不同程度地提高了算法的收敛速度和精度,但效果并不是非常理想。混沌(Chaos)是自然界中一种常见的非线性现象。混沌变量看似杂乱的变化过程其实含有内在的规律性,利用混沌变量的随机性、遍历性及规律性可以进行优化搜索<sup>[15,16]</sup>。本文将混沌优化思想引入到粒子群优化算法中,给出了混沌粒子群优化算法。其基本思想是首先对粒子群体中的最优粒子进行混沌寻优,然后把混沌寻优的结果随机替换粒子群体中的一个粒子。这种处理改善了粒子群优化算法摆脱局部极值点的能力,提高了算法的收敛速度和精度。仿真结果表明混沌粒子群优化算法的收敛性能明显优于粒子群优化算法。

## 2 混沌粒子群优化算法

一般将由确定性方程得到的具有随机性的运动状态称为混沌,呈现混沌状态的变量称为混沌变量。如下的 Logistic 方程<sup>[15]</sup>是一个典型的混沌系统:

$$z_{n+1} = \mu z_n (1 - z_n) \quad n=0,1,2,\dots \quad (1)$$

<sup>\*</sup>国家自然科学基金(60274006)、广东省优秀人才基金(2000-6-15)、华南理工大学自然科学基金资助项目。高 鹰 博士后,副教授,主要研究领域:盲信号处理、人工神经网络、小波分析和智能信息处理等。谢胜利 教授,博士生导师,主要研究领域:智能信息处理、盲信号处理、非线性系统学习控制等。

式中  $\mu$  为控制参量, 方程(1)可以看作是一个动力学系统。 $\mu$  值确定后, 由任意初值  $z_0 \in [0, 1]$ , 可迭代出一个确定的时间序列  $z_1, z_2, z_3, \dots$ 。一个混沌变量在一定范围内有如下特点: 随机性, 即它的表现同随机变量一样杂乱; 遍历性, 即它可以不重复地历经空间内的所有状态; 规律性, 该变量是由确定的迭代方程导出的。混沌优化方法是一种新颖的优化方法, 它利用混沌系统特有的遍历性来实现全局最优, 而且它不要求目标函数具有连续性和可微性的性质。

粒子群优化算法最初是 Kennedy 和 Eberhart<sup>[1,2]</sup> 从模拟社会行为而发展起来的具有全局寻优能力的优化工具。它通过迭代搜寻最优值, 系统初始化为一组随机解, 而粒子(潜在的解)在解空间追随最优的粒子进行搜索。假设在一个  $n$  维的目标搜索空间中, 有  $N$  个粒子组成一个群体, 其中第  $i$  个粒子表示一个  $n$  维的向量  $x_i = (x_{i,1}, x_{i,2}, \dots, x_{i,n})$ ,  $i=1, 2, \dots, N$ , 分量  $x_{i,j}$  在  $[a_j, b_j]$  范围内取制值, 即  $a_j \leq x_{i,j} \leq b_j$ ,  $i=1, 2, \dots, N, j=1, 2, \dots, n$ , 每个粒子的位置就是一个潜在的解。将  $x_i$  带入一个目标函数就可以计算出其适应值, 根据适应值的大小衡量  $x_i$  的优劣。第  $i$  个粒子的“飞行”速度也是一个  $n$  维的向量, 记为  $u = (u_{i,1}, u_{i,2}, \dots, u_{i,n})$ ,  $i=1, 2, \dots, N$ 。记第  $i$  个粒子迄今为止搜索到的最优位置为  $p_i = (p_{i,1}, p_{i,2}, \dots, p_{i,n})$ ,  $i=1, 2, \dots, N$ , 整个粒子群迄今为止搜索到的最优位置为  $p_g = (p_{g,1}, p_{g,2}, \dots, p_{g,n})$ , 粒子群优化算法采用下列公式对粒子操作:

$$u \leftarrow u + c_1 r_1 (p_i - x_i) + c_2 r_2 (p_g - x_i) \quad (2)$$

$$x_i \leftarrow x_i + u \quad (3)$$

其中,  $i=1, 2, \dots, N$ ; 学习因子  $c_1$  和  $c_2$  是非负常数;  $r_1$  和  $r_2$  是介于  $[0, 1]$  之间的随机数。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最小适应阈值。

粒子群优化算法虽然简单, 但其有易陷入局部极值点, 进化后期收敛速度慢, 精度较差等的缺点。如能采取某种优化手段使每一代群体的质量进一步提高, 则无疑会有助于后面的搜索过程。为此, 我们把混沌优化思想引入到粒子群优化算法中, 提出了混沌粒子群优化算法。主要措施是利用混沌运动的遍历性以当前整个粒子群迄今为止搜索到的最优位置为基础产生混沌序列, 把产生的混沌序列中的最优位置粒子随机替代当前粒子群中的一个粒子的位置。提出的混沌粒子群优化算法的具体步骤如下:

① 确定参数: 学习因子  $c_1, c_2$ , 和群体规模  $N$ , 进化次数, 混沌寻优次数。

② 随机产生  $N$  个粒子的种群。

③ 按(2)和(3)式对粒子进行操作。

④ 对最优位置  $p_g = (p_{g,1}, p_{g,2}, \dots, p_{g,n})$  进行混沌优化。将  $p_{g,i}$  ( $i=1, 2, \dots, n$ ) 映射到 Logistic 方程(1)的定义域  $[0, 1]$ :

$z_i = \frac{p_{g,i} - a_i}{b_i - a_i}$ , ( $i=1, 2, \dots, n$ ), 然后, 用 Logistic 方程(1)进行迭代产生混沌变量序列  $z_i^{(m)}$  ( $m=1, 2, \dots$ ), 再把产生的混沌变量序列  $z_i^{(m)}$  ( $m=1, 2, \dots$ ) 通过逆映射  $p_{g,i}^{(m)} = a_i + (b_i - a_i) z_i^{(m)}$  ( $m=1, 2, \dots$ ) 返回到原解空间, 得

$$p_g^{(m)} = (p_{g,1}^{(m)}, p_{g,2}^{(m)}, \dots, p_{g,n}^{(m)}), (m=1, 2, \dots)$$

在原解空间对混沌变量经历的每一个可行解  $p_g^{(m)}$  ( $m=1, 2, \dots$ ) 计算其适应值, 保留性能最好的可行解  $p^*$ 。

⑤ 随机从当前群体中选出的一个粒子用  $p^*$  取代。

⑥ 若达到最大代数或得到满意解, 则优化过程结束, 否

则返回步骤③。

### 3 算法仿真比较

下面以求一个基准测试函数的最小值为例, 通过计算机仿真来评价比较混沌粒子群优化算法和粒子群优化算法的性能, 并和带惯性因子的粒子群优化算法(IWPSO)、杂交粒子群优化算法(Crossover PSO, CRPSO)和带高斯变异的粒子群优化算法(MPSO)进行比较, 基准测试函数如下:

$$f(x, y) = x^2 - 0.4 \cos(3\pi x) + 2y^2 - 0.6 \cos(4\pi y) - 1$$

其中  $-10 \leq x, y \leq 10$ , 在  $[-10, 10]$  区间内有 1 个全局最小值点  $(0, 0)$ , 全局最小值为 0。

算法的初始化参数如下: 粒子群规模 20, 学习因子  $c_1 = 1, c_2 = 1$ 。带惯性因子的粒子群优化算法中的惯性因子  $w = 0.9$ , 杂交粒子群优化算法中的交叉概率  $P_c = 0.5$ , 带高斯变异的粒子群优化算法中的变异概率  $P_m = 0.05$ , 为评价算法的收敛性能, 进化次数设为 1000, 混沌寻优次数为 500, 连续运行 50 次所得函数全局最小值点的平均值和全局最小值的平均值作为算法的衡量指标。为便于图示说明, 下面的仿真图中横轴表示进化次数, 纵轴表示最优适应度值的对数(即每次进化所得全局最小值的对数)。

图 1 是函数  $f$  最优适应度值的对数(即每次进化所得全局最小值的对数)随进化次数变化的曲线图(50 次独立运行的平均), 图中, 上面一条曲线对应于粒子群优化算法, 而下面一条曲线对应混沌粒子群优化算法。从图中可以看出, 混沌粒子群优化算法的收敛性能明显优于粒子群优化算法的收敛性能。表 1 是粒子群优化算法和混沌粒子群优化算法数值仿真结果, 由表 1 可以看出, 混沌粒子群优化算法对函数的求解结果优于粒子群优化算法的求解结果(50 次独立运行的平均)。对其它函数所做的大量的计算机仿真结果亦说明了这一点, 限于篇幅, 这里就不再给出结果。

表 1 PSO 和 CPSO 算法数值仿真结果

	全局最小值点平均值	全局最小值平均值
PSO	(0.01234485, 0.00233262)	0.01575119
CPSO	(1.464e-007, 0.623e-007)	4.3978e-011

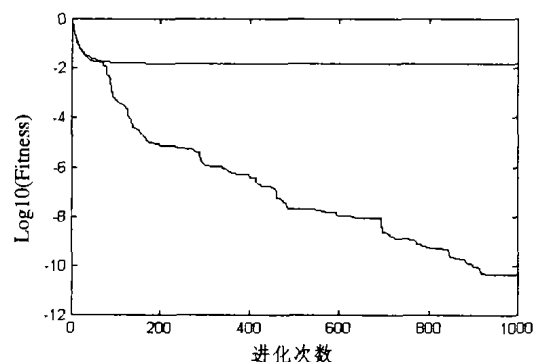


图 1 PSO 和 CPSO 算法进化曲线比较

文[3]对(2)式作了如下的改动:

$$u \leftarrow wu + c_1 r_1 (p_i - x_i) + c_2 r_2 (p_g - x_i) \quad (4)$$

其中  $w$  是非负数, 称为惯性因子(inertia weight)。由此得到的算法称为带惯性因子的粒子群优化算法。

借鉴遗传算法的思想, 文[6]最早提出了杂交粒子群优化算法。粒子群中的粒子被赋予一个杂交概率, 这个杂交概率由用户确定, 与粒子的适应值无关。在每次迭代中, 依据杂交概率选取指数量的粒子放入一个池中。池中的粒子随机地两两

杂交,产同样数目的孩子粒子,并用孩子粒子代替父母粒子,以保持群的粒子数日不变。孩子粒子的位置由父母粒子的位置的加权和计算,即:

$$child_1(x) = p \times parent_1(x) + (1-p) \times parent_2(x) \quad (5)$$

$$child_2(x) = p \times parent_2(x) + (1-p) \times parent_1(x) \quad (6)$$

其中  $x$  是  $D$  维的位置向量,而  $child_k(x)$  和  $parent_k(x)$ ,  $k=1,2$  分别指明是孩子粒子还是父母粒子的位置; $p$  是  $D$  维均匀分布的随机数向量, $p$  的每个分量都在  $[0,1]$  取值,' $\times$ ' 表示向量分量对应相乘。孩子粒子的速度分别由下面的公式得到:

$$child_1(v) = \frac{parent_1(v) + parent_2(v)}{|parent_1(v) + parent_2(v)|} |parent_1(v)| \quad (7)$$

$$child_2(v) = \frac{parent_1(v) + parent_2(v)}{|parent_1(v) + parent_2(v)|} |parent_2(v)| \quad (8)$$

其中  $v$  是  $D$  维的速度向量,而  $child_k(v)$  和  $parent_k(v)$ ,  $k=1,2$  分别指明是孩子粒子还是父母粒子的速度。杂交粒子群优化算法引入了较多的待调整参数,对使用者的经验有一定要求。

表2 CPSO、IWPSO、CRPSO 和 MPSO 算法数值仿真结果

	全局最小值点平均值	全局最小值平均值
CPSO	(0.723e-006, 0.418e-006)	5.018e-010
IWPSO	(0.239e-005, 0.393e-006)	8.175e-009
CRPSO	(0.00130518, 0.00086676)	0.00346642
MPSO	(0.00022500, 0.00036320)	0.00622002

文[8]进一步把变异运算引入粒子群优化算法中,提出了带高斯变异的粒子群优化算法,变异运算依据变异概率选取指定数量的粒子按高斯变异算子进行变异,用变异后的粒子代替原粒子,即:

$$mutation(x) = x * (1 + Gaussian(\sigma)) \quad (9)$$

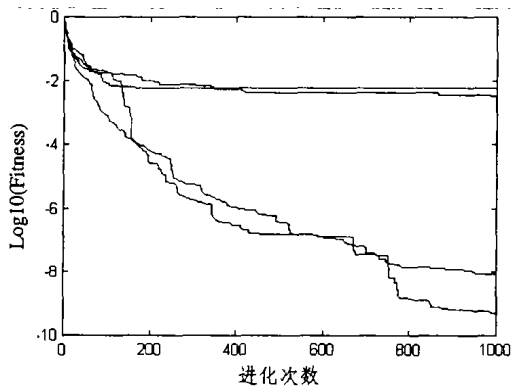


图2 CPSO、IWPSO、CRPSO 和 MPSO 算法进化曲线比较

图2是CPSO、IWPSO、CRPSO和MPSO算法收敛曲线比较,从上往下的四条曲线依次对应的是MPSO算法、CRPSO算法、IWPSO算法和CPSO算法。表2是这四种算法数值仿真结果。从图2和表2可知混沌粒子群优化算法优于杂交粒子群优化算法和带高斯变异的粒子群优化算法,而略好于

带惯性因子的粒子群优化算法。

**结论** 本文把混沌优化思想引入到粒子群优化算法中,给出了混沌粒子群优化算法。其基本思想是在粒子群的每次进化过程中,首先对粒子群体中的最优粒子进行混沌寻优,然后随机地从粒子群体中选择一个粒子用混沌寻优的结果来替换。通过这种处理使得粒子群优化算法摆脱局部极值点的能力得到改善,提高了算法的收敛速度和精度。计算机仿真结果表明混沌粒子群优化算法的收敛性能明显优于粒子群优化算法,亦优于杂交粒子群优化算法和带高斯变异的粒子群优化算法,而略好于带惯性因子的粒子群优化算法。

## 参考文献

- Kennedy J, et al. Particle swarm optimization. In: IEEE Int'l Conf. on Neural Networks. Perth, Australia, 1995. 1942~1948
- Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In: Proc. of the sixth intl. symposium on Micro Machine and Human Science, Nagoya, Japan, 1995. 39~43
- Shi Y, et al. A modified particle swarm optimizer [C]. In: IEEE World Congress on Computational Intelligence, 1998. 69~73
- Shi Y, Eberhart R C. Fuzzy Adaptive particle swarm optimization [C]. In: Proc. of the Congress on Evolutionary Computation, Seoul Korea, 2001
- Clerc M. The swarm and the Queen: Towards a deterministic and adaptive particle swarm optimization [C]. In: Proc. of the Congress of Evolutionary Computation, 1999. 1951~1957
- Angeline P J. Evolutionary optimization versus particle swarm optimization: Philosophy and performance differences [C]. In: Evolutionary programming VII, 1998. 601~610
- Lovbjerg M, Rasmussen T K, Krink T. Hybrid particle swarm optimization with breeding and subpopulations [C]. In: Proc. of the third Genetic and Evolutionary computation conf. San Francisco, USA, 2001
- Higashi N, Iba H. Particle swarm optimization with Gaussian mutation [C]. In: Proc. of the Congress on Evolutionary Computation, 2003. 72~79
- Van den Bergh F, Engelbrecht A P. Training product unit networks using cooperative particle swarm optimizers [C]. In: Proc. of the third Genetic and Evolutionary computation conf. San Francisco, USA, 2001
- Van den Bergh F, Engelbrecht A P. Effects of swarm size cooperative particle swarm optimizers [C]. In: Proc. of the third Genetic and Evolutionary computation conf. San Francisco, USA, 2001
- Kennedy J, Eberhart R. Discrete binary version of the particle swarm algorithm [C]. IEEE Int'l Conf. on computational Cybernetics and Simulation, 1997. 4104~4108
- Clerc M. Discrete particle swarm optimization illustrated by the traveling salesman problem. <http://www.mauriceclerc.net>, 2000
- Ciuprina G, Ioan D, Munteanu I. Use of intelligent-particle swarm optimization in electromagnetics. IEEE Trans. on Magnetics, 2002, 38(2): 1037~1040
- Clerc M, Kennedy J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. on Evolutionary Computation, 2002, 6(1): 58~73
- 王东升, 曹磊. 混沌、分形及其应用 [M]. 合肥: 中国科学技术大学出版社, 1995
- 李兵, 蒋慰孙. 混沌优化方法及其应用 [J]. 控制理论与应用, 1997(4): 613~615

(上接第12页)

- e-government metadata Framework. [http://www.govtalk.gov.uk/documents/uk metadata framework vol202001-05.pdf](http://www.govtalk.gov.uk/documents/uk%20metadata%20framework%20vol202001-05.pdf)
- GCL (Government Category List) <http://www.govtalk.gov.uk/schemasstandards/gcl.asp>
- Mullen A. GILS metadata initiatives at the state lev-el. Government Information Quarterly, 2001, 18: 167~180
- Jordan, readiness for e-government. <http://www.surf-as.org/papers/e-gov-english.pdf>
- Linthcum D S. Mercator: Next generation Application integration From Infromation, or Process, to Service Mercator software, Inc,

2002

- Koeller A. Integration of Heterogeneous Database Discovery of Meta-Information and Maintenance of Schema-restructuring views: [PHD thesis]. Worcester polytechnic Institute, 2001
- IBM Corporation. Optimization of information to improve decision making in government:- the information value chain way. <http://www.ibm.com/services/files/ibv-infovaluech-ain.pdf>
- E-Government Strategy. <http://www.cio.gov/documents/egovreport.pdf>
- Reese W D. An Investigation of Techniques for Integrating Web-accessible Data: [PHD thesis]. University of Colorado, 2001