

## 一种改进的微粒群优化算法

冯春丽<sup>1</sup> 唐毅谦<sup>2</sup> 赵悦<sup>1</sup>

(1. 辽宁工学院信息科学与工程学院, 锦州, 121001; 2. 辽宁工学院工程技术学院, 锦州, 121001)

**摘要:**标准微粒群优化(PSO)算法是一种群体智能算法,它容易陷入局部极值点,进化后期收敛速度慢且精度较差,而且参数的选择对算法的优劣影响很大。针对这些缺点,首先提出了一种在位置进化方程中引进动态参数的方法,改进了标准微粒群算法收敛速度;然后通过速度、位置进化方程中同时引进动态参数来提高算法收敛速度和收敛率。经J. D. Schaffer函数和LevyNo. 5函数对改进算法的测试表明,相比于标准微粒群算法,该方法的收敛速度和平均收敛率均有大幅度提高。

**关键词:**微粒群算法; 优化; 动态参数; 收敛速度

**中图分类号:** TP18 **文献标识码:** A

## Improved Particle Swarm Optimization Algorithm

Feng Chunli<sup>1</sup>, Tang Yiqian<sup>2</sup>, Zhao Yue<sup>1</sup>

(1. School of Information Science & Engineering, Liaoning Institute of Technology, Jinzhou, 121001, China;

2. School of Engineering, Liaoning Institute of Technology, Jinzhou, 121001, China)

**Abstract:** The normal particle swarm optimization (PSO) algorithm is a kind of swarm intelligence methods. It is easy to be trapped into local extremum, and its convergence speed is lower and the precision is worse in the late evolution. Furthermore, the parameter selection can affect the algorithm. Aimed at these disadvantages of normal PSO, the new algorithm by introducing dynamical parameters in the evolution of the position equation is proposed first. The convergence speed is improved in the new algorithm. And then, by introducing dynamical parameters in the evolution of the speed equation and the position equation at the same time, the new algorithm improves its convergence speed and convergence rate. The new method tested by functions J. D. Schaffer and Levy No. 5 shows that the convergence speed and the average convergence rate increase.

**Key words:** particle swarm algorithm; optimization; dynamic parameter; convergence speed

## 引言

微粒群优化(PSO)算法最早是由美国 James Kennedy 和 Russell Eberhart 在 1995 年提出的一种智能计算方法<sup>[1]</sup>。其基本思想是受他们早期对鸟类群体行为研究结果的启发,利用和改进了生物学家 Frank Heppner 的生物群体模型。短短十几年, PSO 算法的研究已经获得了很大的发展<sup>[2]</sup>, PSO

作为一种高效的并行搜索算法,非常适于对复杂环境中的优化问题的求解,因此,对其进行理论和应用研究具有重要的学术意义和工程价值。目前, PSO 已经应用到如电力、化工、机器人机械设计、通讯、经济学、生物信息和医学运筹学等领域<sup>[3]</sup>。

大量实验发现,微粒群算法比遗传算法(GA)具有更好的性能<sup>[4]</sup>,它能更快地达到全局最优值,并且基本不受问题峰值和维数增加的影响,在处理

基金项目:国家自然科学基金(60274024)资助项目。

收稿日期:2006-03-20;修订日期:2006-05-01

作者简介:冯春丽,女,硕士研究生,1981年10月生;唐毅谦(联系人),男,博士、教授,1964年4月生,E-mail:yqt111@163.com。

复杂的尤其是多峰值问题上有很强的优越性。但标准 PSO 算法往往不能兼顾收敛速度、全局探索能力和局部搜索能力。本文通过引入动态系数调整位置进化方程的方法以提高 PSO 算法收敛速度。在动态调整惯性权重和加速度权重的基础上<sup>[5]</sup>,同时动态调整位置进化方程,通过计算实例表明效果显著。

## 1 标准微粒群算法

在 PSO 中,每个优化问题的解都看作是搜索空间中一只鸟,称之为“微粒”,把这些微粒看作是  $D$  维搜索空间的一些没有质量和体积但具有速度和位置的点,每个微粒都具有一个适应值,速度决定它们飞翔的方向和距离,微粒们追随当前的最优微粒在解空间中搜索<sup>[6]</sup>。PSO 初始化为一群随机微粒,然后通过迭代找到最优解。在每一次迭代中,微粒通过跟踪两个“极值”(个体极值和全局极值)来更新自己。迭代中止条件根据具体问题,一般选为最大迭代次数或微粒群搜索到的最优位置满足预定最小适应阈值。

微粒群算法的描述如下<sup>[7]</sup>:设在  $D$  维空间中有  $n$  个微粒,  $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$  为微粒  $i$  的当前位置,  $V_i = (v_{i1}, v_{i2}, \dots, v_{id})$  为微粒  $i$  的当前飞行速度,  $P_i = (p_{i1}, p_{i2}, \dots, p_{id})$  为微粒  $i$  所经历过的具有最好适应值的位置,称为个体最好位置(pBest)。  $P_g = (p_{g1}, p_{g2}, \dots, p_{gd})$  为整个种群中微粒所经历过的具有最好适应值的位置,称为全局最好位置(gBest)。

对第  $t$  代的第  $i$  个微粒,微粒群算法根据下列进化方程计算第  $t+1$  代的第  $j$  维的速度和位置

$$v_{ij}(t+1) = \omega(t)v_{ij}(t) + c_1r_1(p_{ij}(t) - x_{ij}(t)) + c_2r_2(p_{gi}(t) - x_{ij}(t)) \quad (1)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad (2)$$

式中:  $\omega(t)$  为惯性权重,目前称惯性权重线性变化<sup>[8]</sup>的为标准微粒群算法,它使微粒保持运动的惯性,使其有能力探索新的区域;  $c_1$  和  $c_2$  为加速常数,它们使每个微粒向 pBest 和 gBest 位置加速运动;  $r_1$  和  $r_2$  为  $[0, 1]$  范围里变化的随机数。此外,微粒的速度  $V_i$  被一个最大速度  $V_{\max}$  所限制。如果当前对微粒的加速将导致它在某维的速度  $v_{id}$  超过该维的最大速度  $V_{\max, d}$ , 则该维的速度被限制为该维最大速度  $V_{\max, d}$ 。

从微粒速度更新公式可看成由 3 部分组成: (1)微粒先前的速度,说明了微粒目前的状态,起到了平衡全局和局部搜索的能力; (2)认知部分,表示

微粒本身的思考,使微粒有了足够强的全局搜索能力,避免局部极小; (3)社会部分,体现了微粒间的信息共享。在这 3 部分的共同作用下,微粒根据历史经验并利用信息共享机制,不断调整自己的位置,以期找到问题的最优解。PSO 算法的搜索性能取决于对全局搜索和局部搜索能力的平衡,这很大程度上依赖于算法的控制参数<sup>[9]</sup>,包括种群规模、最大速度、最大代数、惯性权因子和加速常数等。

## 2 参数的动态调整

PSO 算法具有很多的优点: (1)算法通用,不依赖于问题信息; (2)群体搜索,并具有记忆能力,保留局部个体和全局种群的最优信息; (3)原理简单,容易实现; (4)协同搜索,同时利用个体局部信息和群体全局信息指导算法下一步的搜索。但大量的实验事实证明,标准微粒群算法容易陷入局部最优,搜索精度不高,因此有必要对其进行改进。

### 2.1 位置的动态调整

为使微粒在开始的时候,以较大增量增加来加速靠近最优位置,而在后期以较小的增量增加,以防止因位置跨越过大而脱离最优位置陷入局部最优。本文对 PSO 进化方程(2)进行修正,引进变化

参数  $\mu(t) = \mu_{\max} - \frac{(\mu_{\max} - \mu_0) \cdot t}{\text{iter}_{\max}}$ , 随迭代次数  $t$  线性减少,  $\text{iter}_{\max}$  为最大进化代数,  $\mu_0, \mu_{\max}$  的取值视具体情况而定,  $\mu_0$  建议取值范围为  $1.8 \sim 1.0$ ,  $\mu_{\max}$  建议取值范围为  $0.8 \sim 0.4$ 。则位置更新方程(2)变为

$$x_{ij}(t+1) = x_{ij}(t) + \mu(t)v_{ij}(t+1) \quad (3)$$

为验证该算法的有效性,这里采用 J. D. Schaffer

函数  $f(X) = \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2} - 0.5$ ,  $-100 \leq x_1, x_2 \leq 100$ , 进行测试。

在标准 PSO 算法中,  $\omega$  是随进化代数逐渐减少,这里取为  $\omega(t) = 1.0 - \frac{0.6t}{\text{iter}_{\max}}$ ,  $c_1 = c_2 = 1.8$ 。在位置动态 PSO 算法中,取  $\mu_0 = 0.4$ ,  $\mu_{\max} = 1.0$ , 故  $\mu(t) = 1.0 - \frac{(1.0 - 0.4) \cdot t}{\text{iter}_{\max}}$ , 群体规模取为 20,  $\text{iter}_{\max} = 500$ , 分别进行了 5 次试验,每次共进行 50 次计算,误差取为 0.001。计算结果与标准 PSO 的对比如表 1 所示。

从计算结果可以看出,位置动态参数 PSO 对平均收敛代数(即收敛速度)有明显提高,但对平均收敛率来说,改善效果却不尽人意。

### 2.2 速度、位置的动态调整

对全局的搜索,通常方法是在前期有较高探索

表1 位置动态调整 PSO 与标准 PSO 对比

算法	试验次数	平均收敛率/%	平均收敛代数
标准 PSO	1	48	321.42
	2	46	327.13
	3	48	334.25
	4	48	334.79
	5	50	328.40
位置动态调整 PSO	1	42	269.81
	2	38	280.58
	3	54	265.30
	4	54	256.70
	5	52	288.08

能力以得到合适的种子,而后期有较高的开发力以加快收敛速度。同时动态调整惯性权重 $\omega$ 、加速度权重 $c_2$ 两个参数的速度动态调整 PSO<sup>[5]</sup>,能够使得 PSO 算法在初期具有较强的全局收敛能力,而晚期具有较强的局部收敛能力,为进一步改进算法性能,结合本文提出的在位置进化方程(2)中加动态调整参数,可以得到如下进化方程

$$v_{ij}(t+1) = \omega(t)v_{ij}(t) + c_1r_1(p_{ij}(t) - x_{ij}(t)) + c_2(t)r_2(p_{gj}(t) - x_{ij}(t)) \quad (4)$$

$$x_{ij}(t+1) = x_{ij}(t) + \mu(t)v_{ij}(t+1) \quad (5)$$

式中: $c_2(t) = c_{20} + \frac{(c_{2iter\_max} - c_{20})t}{iter\_max}$ ,  $c_{20}, c_{2iter\_max}$  的取值与  $\mu_0, \mu_{max}$  一样,需要视情况而定。建议  $c_{20}$  取值范围为 0.6~1.0,  $c_{2iter\_max}$  取值范围为 1.6~2.0。

为验证本文提出的动态调整的 PSO 算法的性能,进行了以下实例的计算,所用的测试函数为:

$$f(X) = \sum_{i=1}^5 [i \cos((i-1)x_1 + i)] \times \sum_{j=1}^5 [j \cos((j+1)x_2 + j)] + (x_1 + 1.42513)^2 + (x_2 + 0.80032)^2 \quad (6)$$

式中:  $-10 \leq x_1, x_2 \leq 10$ 。

该函数为 LevyNo. 5 函数,它具有 760 个局部极值点,全局最优解为  $x = (-1.3068, -1.4248)$ ,最优值为 -176.1375。本文测试过程中,取微粒数为 20,终止条件为最优适应值与最优解之差小于 0.0001,限定最大迭代次数  $iter\_max$  为 130,每次优化计算 100 次,共做 5 次试验,考察达到最优解的次数,并且求出平均每次所用的进化代数。各算法中,  $\omega(t) = 1.0 - \frac{0.5t}{iter\_max}$ , 标准 PSO 算法中,  $c_1 = c_2 = 1.8$ , 速度动态调整 PSO 中  $c_1 = 1.8, c_2(t) = 0.8 + \frac{(1.8-0.8)t}{iter\_max}$ ; 在本文改进的 PSO 算法中  $\mu(t) = 1.0$

$-\frac{(1.0-0.5) \cdot t}{iter\_max}$ ,  $c_1 = 1.8, c_2(t) = 0.8 + \frac{(1.8-0.8)t}{iter\_max}$ , 三种算法在 LevyNo. 5 函数的求解全局最优解过程中,其最优次数和平均进化代(数)的比较结果如表 2 所示。

表2 三种算法的比较

试验次数	项目	标准 PSO	速度动态调整 PSO <sup>[5]</sup>	本文改进 PSO
1	达最优次数	72	87	97
	平均进化代	117.31	109.08	99.57
2	达最优次数	68	88	96
	平均进化代	117.06	109.20	96.85
3	达最优次数	65	89	98
	平均进化代	115.40	108.33	98.06
4	达最优次数	68	89	97
	平均进化代	116.21	108.26	97.15
5	达最优次数	73	86	96
	平均进化代	117.23	107.22	97.97

可以看出,对于标准 PSO 算法及速度动态调整 PSO,达到最优解的次数明显少于本文提出的改进 PSO 算法,而且平均进化代数也均高于本文提出的 PSO 算法。需要说明的是:各个参数的动态调整除了采用线性关系外,还可采用其他比例关系,目前来说,没有一种对所有问题都十分有效的动态调整参数的方法。本文所用的仿真环境均为 VC++6.0 企业版。

### 3 结束语

PSO 算法只需很少的代码和参数,在应用过程中,关键参数的选择决定了算法整体性能的差异<sup>[10]</sup>。本文首先通过在位置进化方程加动态参数,动态调整位置进化方程,计算实例表明,在收敛速度方面有很大提高;又结合速度动态调整 PSO,进一步对微粒群算法进行动态调整,计算实例表明效果显著。

#### 参考文献:

- [1] Kennedy J, Eberhart R C. Particle swarm optimization[C]//Proc IEEE Int Conf Neural Networks Piscataway. NJ, IEEE Press, 1995, 1942-1948.
- [2] Ming Dong, Ravi K. Feature subset selection using a new definition of classifiability[J]. Pattern Recognition Letters, 2003, 24: 1215-1225.
- [3] 刘波, 王凌, 金以慧. 微粒群优化算法研究进展[J]. 化工自动化及仪表, 2005, 32(3): 1-6.
- [4] 陈国初, 俞金寿. 微粒群优化算法[J]. 信息与控制,

- 2005,34(3):319-324.
- [5] 张更新,赵辉,王红君. 基于动态参数的微粒群算法(PSO)的研究[J]. 天津理工大学学报,2005,21(4):42-44.
- [6] Zhang Liping, Yu Huanjun, Hu Shangxu. Optimal choice of parameters for particle swarm optimization [J]. J Zhejiang Univ SCI, 2005, 6A(6): 528-534.
- [7] Liu Yijian, Zhang Jianming, Wang Shuqing. Parameter estimation of cutting tool temperature nonlinear model using PSO algorithm[J]. Journal of Zhejiang University SCIENCE, 2005, 6A(10): 1026-1029.
- [8] Shi Y, Eberhart R C. Empirical study of partical swarm optimization [C]//Proceedings of the 1999 Congress on Evolutionary Computation. Piscataway, NJ, IEEE Service Center, 1999:1945-1950.
- [9] Trelea I C. The PSO algorithm convergence analysis and parameter selection[J]. Information Processing Letters, 2003, 85: 317-325.
- [10] 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.