第28卷第6期
2008年12月

杭州电子科技大学学报
Journal of Hangzhou Dianzi University

Vol.28, No.6
Dec.2008

一种改进的粒子群算法

徐青鹤, 刘士荣, 吕　强
(杭州电子科技大学自动化研究所, 浙江 杭州 310018)

摘要:针对粒子群算法搜索精度不高的问题,提出了一种改进的粒子群算法。该算法一方面通过跟踪个体极值、全局极值和周围极值来搜索解空间的最优值;另一方面通过引入3种非线性递减函数对惯性权重进行调整,仿真结果表明改进的粒子群算法具有更强的寻优能力及更高的搜索精度。

关键词:粒子群算法;极值;惯性权重

中图分类号:TP18　　**文献标识码**:A　　**文章编号**:1001-9146(2008)06-0103-04

0 引　言

粒子群算法[1](Particle Swarm Optimization, PSO)源于对鸟类捕食行为的研究,是一种智能优化算法,受到国内外众多学者的关注。提出了一种用模糊规则动态调整的方法,提高了算法的全局搜索能力和搜索精度[2];在PSO算法中利用梯度信息,不仅加快了寻优速度,而且可以避免局部极小[3]。这些方法从不同角度对PSO算法进行了改进,取得不错的效果。但是这些方法也存在一些不足(实现困难、不符合实际应用、改进效果不明显)。基于上述原因,本文提出一种改进简单,实现容易,但效果明显的粒子群算法,称为改进的粒子群算法(Modified Particle Swarm Optimization, MPSO)。该算法基于两方面对粒子群算法进行改进,一方面除跟踪两个极值(个体极值和全局极值)外,还跟踪每个粒子周围搜索到的最值(称为周围极值);另一方面通过引入3个非常有效的非线性函数对惯性权重ω进行调整。实验结果表明该方法的有效性。

1 粒子群算法

PSO算法通过跟踪两个极值来搜索解空间的最优值:一个是每个粒子搜索到的最值,称为个体极值;另一个是整个群体搜索到的最值,称为全局极值。其更新公式表示为:

$$v_i^{k+1} = \omega v_i^k + c_1\zeta(p_i^k - \chi_i^k) + c_2\eta(p_g^k - \chi_i^k) \tag{1}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{2}$$

式中,$v_i = [v_{i1}, v_{i2}, \cdots, v_{in}]$是粒子的速度;$x_i = [x_{i1}, x_{i2}, \cdots, x_{in}]$是粒子的目前位置;$p_i$是迄今搜索到的个体最优解;$p_g$是整个群体迄今搜索到的最优解;ω称为惯性权重;$c_1$、$c_2$为常数;ζ、η是[0,1]区间内均匀分布的随机数。

收稿日期:2008-08-31
基金项目:国家自然科学基金资助项目(60675043),浙江省科技计划资助项目(C21051)
作者简介:徐青鹤(1984-),男,浙江温州人,在读研究生,移动机器人的路径规划与控制.

2 改进的粒子群算法

为了平衡算法收敛速度和寻优能力,MPSO将从两方面同时对基本PSO进行改进:

2.1 嵌入局部最优模式的粒子群更新算法

粒子群研究依赖于几种简单的社会结构,尤其是个体与它们直接相邻邻居之间的影响和所有个体与种群中性能最好的个体之间的影响[4]。基本粒子群算法没有很好的体现个体与他们直接相邻邻居之间的影响,故本文在粒子群算法中嵌入局部最优模式即引入周围极值,则粒子的速度更新公式为:

$$v_i^{k+1} = \omega v_i^k + c_1\zeta(p_i^k - x_i^k) + c_2\eta(p_g^k - x_i^k) + c_3\eta(p_j^k - x_i^k) \quad (3)$$

式中,p_j表示粒子跟踪其周围粒子的最优值,c_3是粒子跟踪周围最优值的权重系数,其他一些系数同上。本文在搜索周围极值时采用轮形"Lbest"模型。这种改进的粒子群算法用MPSO - W(Modified Particle Swarm Optimization - Wheel)表示。

2.2 非线性递减策略调整ω的粒子群更新算法

在粒子群算法的可调参数中,惯性权重ω对算法的性能影响很大,较大的权值有利于提高算法的全局搜索能力,而较小的权值会增强算法的局部搜索能力。本文受线性调整ω策略思想的启发,构造3种非线性函数分别对ω进行调整,试图能够更加合理的反映粒子群搜索的非线性过程。其具体表示如下:

$$\omega = -(\omega_{max} - \omega_{min})(iter/Gen)^n + \omega_{max} \quad (4)$$

这是一条开口向下的非线性曲线,其中:ω_{max}、ω_{min}分别表示最大和最小的惯性权重值,表示最大迭代次数,表示当前的迭代次数。采用这种非线性调整ω的粒子群算法,用MPSO - DF(Modified Particle Swarm Optimization - Down Function)来表示。大量的实验结果表明当$n = 1/2$时,寻优能力最强,故本文采用$n = 1/2$。

$$\omega = (\omega_{max} - \omega_{min})(iter/Gen)^3 + (\omega_{min} - \omega_{max})(iter/Gen)^2 + (\omega_{min} - \omega_{max})(iter/Gen) + \omega_{max} \quad (5)$$

这是一条开口向上的非线性曲线,采用这种非线性调整ω的粒子群算法,用MPSO - UF (Modified Particle Swarm Optimization - Up Function)来表示。

$$\omega = \omega_{min}(\omega_{max}/\omega_{min})^{1/(1+n_1 iter/Gen)} \quad (6)$$

这是一条指数曲线,采用这种非线性调整ω的粒子群算法,用MPSO - EF(Modified Particle Swarm Optimization - Exponential Function)来表示。实验结果表明当$n_1 = 5$时,寻优能力最好,故本文采用$n_1 = 5$。本文ω的取值范围为[0.4,0.95],$c_1 = c_2 = 2$。为方便起见,将上述两种改进方法结合时,即MPSO - W与MPSO - DF结合,用MPSO - I来表示;MPSO - W与MPSO - UF结合,用MPSO - II来表示;MPSO - W与MPSO - EF结合,用MPSO - III来表示。

2.3 收敛性分析

假设初始条件$x_i^0 = x_0, x_i^1 = x_1$,由式5、6得到以下非齐次递归关系:

$$x_i^{k+1} = (1 - \omega j_1 - j_2 - j_3)x_i^k - \omega x_i^{k-1} + j_1 p_i^k + j_2 p_g^k + j_3 p_j^k \quad (7)$$

式中,$j_1 = c_1\zeta, j_2 = c_2\eta, j_3 = c_3\eta$。则可得到式7的封闭形式为:$x_i^k = k_1 + k_2\alpha^k + k_3\beta^k$,可求的,$\alpha = \dfrac{1+\omega - j_1 - j_2 - j_3 + \gamma}{2}$,$\beta = \dfrac{1+\omega - j_1 - j_2 - j_3 - \gamma}{2}$。其中$\gamma = \sqrt{(1+\omega - j_1 - j_2 - j_3)^2 - 4\omega}$。现在考虑$x_i^k$的极值:

$$\lim_{k\to\infty} x_i^k = \lim_{k\to\infty}(k_1 + k_2\alpha^k + k_3\beta^k) \tag{8}$$

显然,只有当 $\max\|\alpha\|, \|\beta\|) < 1$ 时,$\{xki\}_{k=0}^{+\infty}$才收敛,即:

$$\lim_{k\to\infty} x_i^k = \lim_{k\to\infty}(k_1 + k_2\alpha^k + k_3\beta^k) = k_1 \tag{8}$$

当 $\max\|\alpha\|, \|\beta\|) < 1$ 时,得:$\omega > 0.5(j_1 + j_2 + j_3) - 1$。所以只有当条件满足 $\omega > 0.5(j_1 + j_2 + j_3) - 1$ 时,算法才保证收敛。但是 PSO 算法之所以搜索精度不高,是因为粒子速度下降至 0 时,所搜索到的解尚未达到全局最优。因此对 PSO 算法的改进不能只着眼于收敛性,而应调整算法的搜索范围,以及全局和局部搜索能力。鉴于惯性权重对粒子速度的影响及寻优模式对算法性能的影响,所以本文提出了基于两方面改进的粒子群算法,以提高 PSO 算法的全局寻优性能与效率。

3 仿真研究

为了验证 MPSO 算法性能,本文采用 6 个经典的测试函数进行测试,分别是 Tablet 函数、Quadric 函数、Rosebrock 函数及 Griewank 函数、Rastrigrin 函数、Ackley 函数。具体了解可参考文献 5,其中前 3 个为单峰函数,后 3 个为多峰函数。以上 6 个函数当它们取到合适的值时,他们的极值都为 0。在引入周围极值的同时,分别对惯性权重 ω 进行上述 3 种非线性调整,其中 $c_3 = 1$,取粒子数为 20 个,维数 30 维,最大迭代次数 2 000 次,其他条件均相同的情况下,对上述 6 个高维函数进行测试,重复做 30 次求得的均值与方差结果与基本 PSO 测试结果相比,其具体情况如表 1 所示。

表 1　30 维复杂函数优化测试结果

函数	PSO		MPSO - I		MPSO - II		MPSO - III	
	Average	standard error	Average	standard error	Average	standard error	Average	standard error
Tablet	1.63×10^{-9}	9.49×10^{-18}	2.93×10^{-15}	1.48×10^{-29}	8.86×10^{-16}	4.42×10^{-30}	2.76×10^{-17}	1.31×10^{-33}
Quadric	22.272	181.491	11.005	167.45	8.400	32.863	2.815	4.839
Rosebrock	60.571	1.00×10^{3}	29.957	677.024	54.590	1.05×10^{3}	28.942	483.705
Griewank	0.013	1.26×10^{-4}	0.013	1.60×10^{-4}	0.008	5.02×10^{-5}	0.012	1.05×10^{-4}
Rastrigrin	2.31×10^{-9}	1.80×10^{-17}	9.08×10^{-15}	3.89×10^{-28}	6.69×10^{-16}	1.25×10^{-30}	1.74×10^{-15}	9.08×10^{-29}
Ackley	0.258	0.365	0.122	0.139	0.081	0.101	0.599	0.684

由表 1 试验所得最优解的均值和方差可知,改进的粒子群算法 MPSO(MPSO - I、MPSO - II、MPSO - III)与基本粒子群算法(PSO)相比,其寻优能力和搜索精度及稳定性都有较大的提高。其中对 Tablet 函数、Quadric 函数、Rastrigrin 函数的试验效果最为明显,而对于其它 3 个函数来说,MPSO 算法其寻优能力和搜索精度也有所提高。为进一步说明 MPSO 算法良好的寻优能力,本文对上述改进结果不是特别明显的 Griewank 函数和 Ackley 函数进行另一搜索寻优试验,除迭代次数为 500 次外(每迭代 10 次输出这 10 次中的最小值),其它设置同上,重复做 20 次试验,其寻优平均最优适应度变化曲线如图 1 和图 2,同样表明该方法的有效性。而单独从某一方面做改进时,效果略有提高,但没有同时从两方面做改进时效果好。

4 结　论

本文针对基本粒子群算法对高维复杂函数搜索精度不高和容易陷入局部最优点的问题,提出了基于两方面改进的粒子群算法,从本文计算所得到的最优解的均值和方差及平均最优值变化曲线如图 1、2 所示,改进的 PSO 算法具有更强的寻优能力和更高的搜索精度,并且在稳定性方面都有较大的提高。

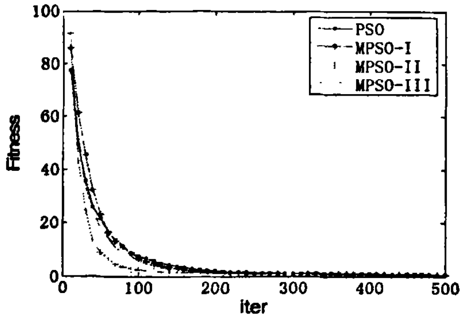

图1 4种方法对 Griewank 函数性能比较

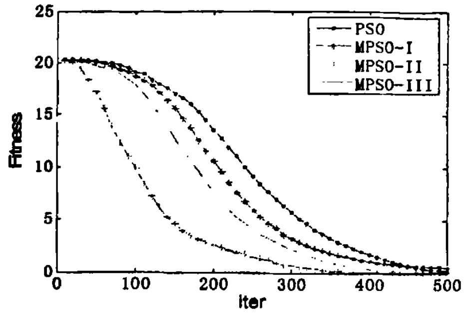

图2 4种方法对 Ackley 函数性能比较

参考文献

[1] Kennedy J, Eberhart R C. Particle Swarm Optimization[C]. Piscataway: Proceedings of the IEEE International Conference on Neural Networks. 1995:1942 - 1948.

[2] Shi Y, Eberhart R C. Fuzzy Adaptive Particle Swarm Optimization[C]. Soul: Proceedings of the IEEE Conference on Evolutionary Computation, 2001:101 - 106.

[3] Noel M M, Jannett T C. Simulation of a new hybrid particle swarm optimization algorithm[C]. Atlanta: Proc of the 36th Southeastern Symposium on System Theory, 2004:150 - 153.

[4] Clerc M, Kennedy J. The Particle Swarm: Explosion, stability and Convergence in a Multi - Dimensional Complex Space[J]. IEEE Transaction on Evolutionary Computation, 2002,6(1):58 - 73.

[5] Yao X, Liu Y, Lin G M. Evolutionary Programming Made Faster[J]. IEEE Transactions on Evolutionary Computation, 1999, 3 (2):82 - 102.

A Modified Particle Swarm Optimization Algorithm

XU Qing-he, LIU Shi-rong, LV Qiang

(*Institute of Automation, Hangzhou Dianzi University, Hangzhou Zhejiang* 310018, *China*)

Abstract: A modified particle swarm optimization is proposed to overcome the problems such as low precision that exist in the standard PSO algorithm. On one hand this algorithm searches for the extreme value by tracking three extreme values (individual extreme value, global extreme value, circumference extreme value). On the other hand three non - linear functions are introduced to adjust the inertia weight of the particle swarm optimization algorithm. Simulations show that modified particle swarm optimization algorithm has more powerful optimizing ability and higher optimizing precision.

Key words: particle swarm optimization; extreme value; inertia weight