DOI:10.11721/cqnuj20130624

基于随机参数的粒子群优化算法*

黄少荣

(广东司法警官职业学院 信息管理系，广州 510520)

摘要:粒子群优化算法本质上是一种全局随机优化技术,优化性能高但容易陷于局部最优,并且算法性能很大程度上依赖于参数设置。本文对该算法的 3 个控制参数进行数据实验和调查,分析参数设置对算法性能的影响规律,提出一种改进的粒子群优化算法,该算法在迭代的每一代中,惯性权重和加速系数都是在一定范围内随机产生:$\omega=\mathrm{rand}(0.4,0.7)$,$C_1=\mathrm{rand}(0.5,3.0)$,$C_2=\mathrm{rand}(1,3.5)$。由于该算法的控制参数不再固定取值;而且在一定范围内随机产生,从而增强了算法的多样性和遍历性,能够有效避免算法早熟收敛。通过标准函数的测试,验证了该算法性能优于固定参数粒子群算法和随机加速系数粒子群算法,具有更好的收敛性和稳定性。

关键词:粒子群优化算法;惯性权重;加速系数;随机参数

中图分类号:TP301.6　　　**文献标志码**:A　　　**文章编号**:1672-6693(2013)06-0123-05

粒子群优化算法(Particle swarm optimization, PSO)是一种新型的全局随机搜索算法,由 Kennedy 和 Eberhart 于 1995 年提出,算法概念起源于对鸟群捕食行为的研究[1]。PSO 通过个体间的协作与竞争,实现复杂空间最优解的搜索。由于 PSO 具有良好的生物社会背景而容易理解、简单通用、参数少易实现、收敛速度快等优点,在科学研究和工程应用领域得到广泛关注。PSO 的可调参数只有 3 个:惯性权重 ω、加速系数 c_1 和 c_2,但研究发现 PSO 性能很大程度上依赖于参数设置。为了提高算法性能,很多研究者对 PSO 的参数进行研究并提出多种参数调整策略,对惯性权重的调整有:线性递减策略[2]、自适应控制策略[3-4]、随机产生策略[5]等;对加速系数的调整有:同步线性递减策略[6]、异步时变策略[7]、动态调整策略[8]和随机产生策略[9]等。这些改进措施不同程度地提高了算法的收敛速度和精度。

PSO 参数的各种调整策略中,惯性权重和加速系数经常被单独分析和设置,在调整一种参数时由于另一种参数的取值不一致,很难对各种参数调整做出比较。本文分别对 PSO 的惯性权重和加速系数进行详细的实验调查,得出参数设置对算法性能的影响规律,提出一种基于随机系数的粒子群优化算法,通过惯性权重和加速系数在更大范围内取值以增加粒子多样性,提高算法优化性能。

1 基本粒子群优化算法

在 PSO 中,一群粒子通过个体之间的交互作用,协同来求解优化问题。假设在一个 D 维搜索空间中,有 m 个粒子,每个粒子的位置表示为搜索空间中的一个 n 维向量。第 i 个粒子当前位置表示为 $X_i=(x_{i1},x_{i2},\cdots,x_{iD})$,速度为 $V_i=(v_{i1},v_{i2},\cdots,v_{iD})$,粒子自身经历的历史最优位置为 $P_i=(p_{i1},p_{i2},\cdots,p_{iD})$,种群的历史最优位置为 $P_g=(p_{g1},p_{g2},\cdots,p_{gD})$。每一代中,粒子的速度和位置根据如下方程进行迭代。

$$v_{id+1}=\omega v_{id}+c_1\times\mathrm{rand}()\times(p_{id}-x_{id})+c_2\times\mathrm{rand}()\times(p_{gd}-x_{id}) \tag{1}$$

$$x_{id+1}=x_{id}+v_{id} \tag{2}$$

式中,$i=1,2,\cdots,m$,$d=1,2,\cdots,D$,ω 为惯性权重,c_1 和 c_2 为加速系数,rand( )为[0,1]上的均匀分布的伪随机数。(1)式由 3 部分组成:第 1 部分为“惯性”,代表粒子维持自己先前速度的趋势;第 2 部分为“认知”,代表粒子向自身历史最佳位置逼近的趋势;第 3 部分为“社会”,代表粒子向群体最佳位置逼近的趋势。算法中止条件一般为最大迭代次数或粒子群迄今为止搜索到的最优位置满足适应阈值。

---

* **收稿日期**:2012-09-23　**修回日期**:2012-12-25　**网络出版时间**:2013-11-20 14:46

资助项目:广东省自然科学基金(No. 101754539192000000)

作者简介:黄少荣,女,副教授,硕士,研究方向为计算智能,E-mail:huangshaorong@163.com

网络出版地址:http://www.cnki.net/kcms/detail/50.1165.N.20131120.1446.201306.123_051.html

2 参数实验调查

2.1 惯性权重 ω 对算法性能的影响规律

惯性权重 ω 的作用是控制粒子原有速度对新速度的影响，使粒子保持运动的惯性，有能力探索新的空间，进而使得 PSO 在全局搜索和局部开发之间取得平衡。ω 是 PSO 的一个关键参数，它的取值决定了算法性能：在 PSO 算法前期，较大的 ω 有利于粒子搜索更大的范围，而在算法后期较小的 ω 能加快收敛速度。

表 1　标准测试函数

测试函数	S	$f_{\min}$
$f_1(x)=\sum_{i=1}^{n}\|x_i\|+\prod_{i=1}^{n}\|x_i\|$	$[-10,10]^n$	0
$f_2(x)=\max\{\|x_i\|,1\leqslant i\leqslant n\}$	$[-100,100]^n$	0
$f_3(x)=\sum_{i=1}^{n}-x_i\sin(\sqrt{\|x_i\|})$	$[-500,500]^n$	−12 569.5
$f_4(x)=\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)+10]$	$[-5.12,5.12]^n$	0

为了调查 ω 对算法性能的影响，本文通过 Benchmarks 测试函数集的 4 个标准函数(其中 $f_1\sim f_2$ 为单峰函数，$f_3\sim f_4$ 为多峰函数)对 ω 做数据统计实验，测试函数如表 1 所示。

在其它参数不变的情况下(种群规模 $NP=30$，加速系数 $c_1=c_2=2.0$，进化代数 1 000，函数维数 30)通过 ω 的变化找出该参数对算法的影响：将 ω 的取值范围设定为[0.1,1.0]，从 0.1 开始取值，每次递增 0.1，直到 1.0，总共取 10 个值，利用这 10 个不同 ω 的 PSO 对函数进行优化，最终结果采用算法独立运行 30 次后的平均值，得到不同 ω 取值下 PSO 的优化性能，结果如图 1～4 所示。

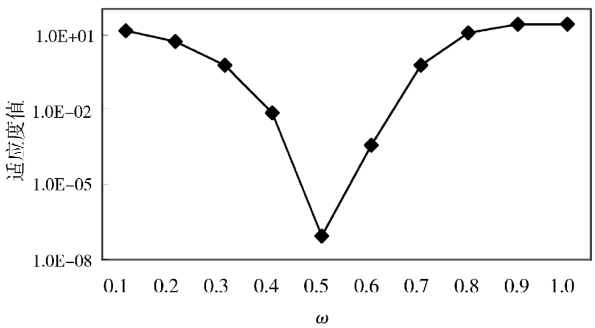

图 1　f_1 调查结果

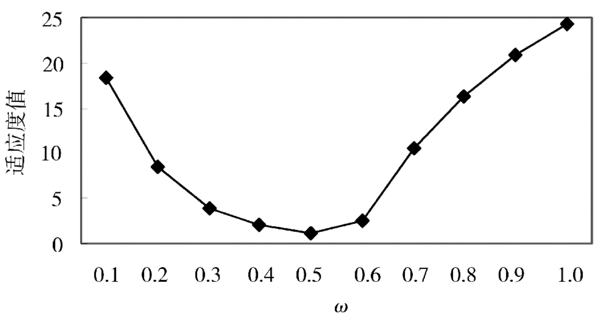

图 2　f_2 调查结果

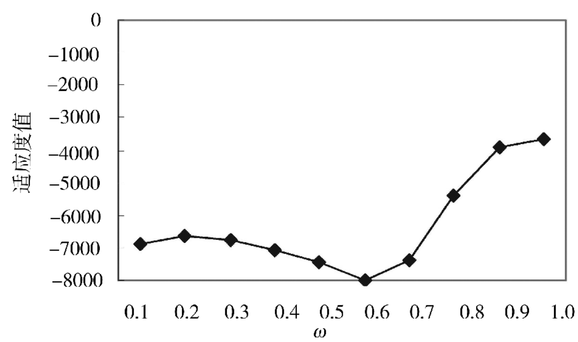

图 3　f_3 调查结果

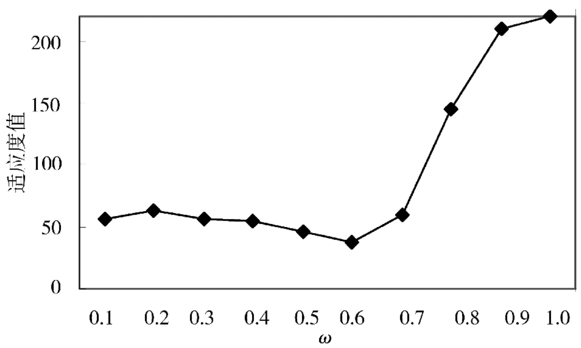

图 4　f_4 调查结果

由调查结果可得：

1)图 1～2 所示的单峰函数(f_1 和 f_2)中，当 ω 在[0.4,0.6]取值时，算法优化性能较好。特别地，当 $\omega=0.5$ 时，算法性能最好。

2)图 3～4 所示的多峰函数(f_3 和 f_4)中，当 ω 在[0.5,0.7]取值时，算法优化性能较好。特别地，当 $\omega=0.6$ 时，算法性能最好。

ω 取值不同导致算法优化性能不同的原因在于：较小的 ω 有使算法具有精细的局部搜索能力，但收敛速度慢，在有限迭代次数下难以收敛至全局最优；而较大的 ω 虽然增强了算法的全局搜索能力，但当粒子接近最优位置时，由于惯性权重过大，粒子有可能一下子飞过最优位置，导致算法难以收敛。

2.2 加速系数 c_1 和 c_2 对算法性能的影响规律

加速系数 c_1 和 c_2 决定粒子自身经验信息和群体经验信息对新速度的影响：c_1 代表粒子个体的“认知能力”，表示粒子在搜索过程中受自身经验影响，不断根据自身的经历调整飞行速度和方向；c_2 代表群体的“社会引导”功能，表示粒子在搜索过程中受种群经验影响，不断根据群体的经历调整自身飞行速度和方向。粒子充分利用自身经验和群体经验调整自身的状态是 PSO 具有优异特性的根本[10]。

为了调查 c_1 和 c_2 对算法性能的影响，选择单峰函数 f_1 和多峰函数 f_4 做详细数据分析：把 c_1 和 c_2 取值范围设定为[0.5,3.5]，c_1 和 c_2 从 0.5 开始取值，每次递增 0.1，直到 3.5，各取 31 个值，当 c_1 取某个值时，跟 c_2 的所有取值一一配对，组成加速系数组合，如 $c_1=0.5$ 时与 c_2 组成加速系数组合[(0.5,0.5),(0.5,0.6),…,(0.5,3.5)]。在每组加速系数组合下，其它参数设置为：种群规模 $NP=30$，惯性权重 $\omega=1.94$，进化代数 1 000，函数维数 30；最终结果采用算法独立运行 30 次后的平均值，得到不同 c_1 和 c_2 取值组合下 PSO 取得的所有满意解，结果如图 5～6 所示。

由调查结果可得：

1)无论是对单峰函数还是多峰函数，算法取得较好性能的时候并不是 c_1 和 c_2 取某个固定的点，而是呈一条带状分布，这表明：只要 c_1 和 c_2 的取值符合条件“$3.0<=c_1+c_2<=4.0$”，就能使算法具有良好优化性能。

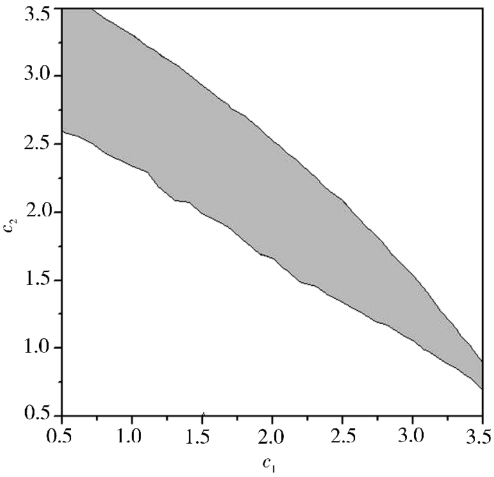

图 5　f_1 调查结果

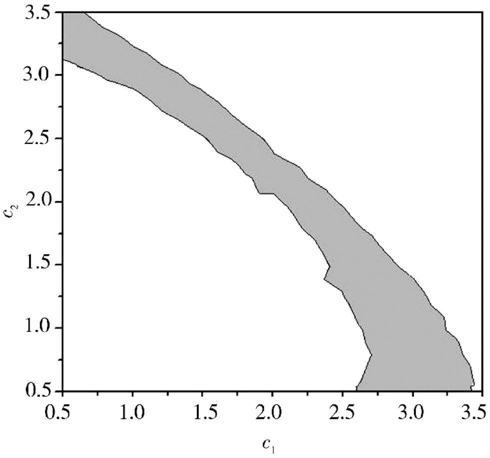

图 6　f_4 调查结果

2)当 c_1+c_2 的值超出[3.0,4.0]这个范围后，算法的性能随之下降。

3)如果把 c_1 和 c_2 仅设置为固定值或按一定规律变化，而不考虑具体优化模型和实现过程，将使算法容易陷入局部最优。

2.3 参数调查结论

根据对惯性权重 ω 和加速系数 c_1 和 c_2 的调查结果可以得出：PSO 本质上是一种随机算法，对于不同函数来说，最佳惯性权重和加速系数并非某一确定点，而是在一定区域内波动。对 ω、c_1 和 c_2 的取值可以是非固定的，传统的固定取值方法只是符合上述条件中的一个样本，除了固定取值，ω、c_1 和 c_2 还可以有多种取值组合，并且很多取值组合能够取得更好的解。

3 随机参数的粒子群优化算法

本文从 PSO 的随机性出发，在对参数进行充分调查基础上，以使算法简单容易使用为目的，基于文献[5]和文献[9]的研究，提出一种改进的随机参数粒子群优化算法：PSO 在执行每一代中，对惯性权重和加速系数不再固定取值，而是在每一代中，在一定范围内，随机产生一组 ω、c_1、c_2 的组合[5,9]

$$\omega = \mathrm{rand}(0.4,0.7) \tag{3}$$

$$c_1=\mathrm{rand}(0.5,3.0) \tag{4}$$

$$c_2=\mathrm{rand}(1.0,3.5) \tag{5}$$

并且 $3.0<=c_1+c_2<=4.0$，否则 c_1 和 c_2 都按比例缩小或者放大：

1) 当 $c_1+c_2<3.0$ 时，$c_1=\frac{c_1}{c_1+c_2}\times 3.0$，$c_2=\frac{c_2}{c_1+c_2}\times 3.0$；

2)当 $c_1+c_2>4.0$ 时，$c_1=\frac{c_1}{c_1+c_2}\times 4.0$，$c_2=\frac{c_2}{c_1+c_2}\times 4.0$。

由于采用随机策略，让 3 个控制参数都均匀随机地在一定范围内取值，让粒子群可以在更大范围内进行随

机搜索,提高了算法的全局优化能力,并且避免了设置参数时的反复测试以及因参数设置不当而产生的误差。由于本文提出新算法仅在参数选择上做出改进,算法流程与标准PSO一致,没有增加额外运算,因此计算时间复杂度与标准PSO同为$O(n^2)$,并且无需额外增加数据存储空间,大大降低了算法的理解复杂度和应用难度,保持了PSO算法实现简单、通用性强的优点,为拓宽PSO在优化领域的应用范围打下基础。

4 实验测试与结果分析

将本文算法计算结果与标准PSO(ω由0.9递减到0.4、$c_1=c_2=2.0$)、文献[9]提出的随机加速系数PSO(ω由0.9递减到0.4、$c_1=\mathrm{rand}(0.5,3.0)$、$c_2=\mathrm{rand}(1.0,3.5)$)做比较,3种算法的种群规模为30,最大进化代数为1 000,每次实验运行30次,取30次平均值为最终结果。3种算法计算结果如表2所示。

表2 函数测试结果

函数	维数	随机参数PSO		标准PSO		随机加速系数PSO	
		最优值	标准差	最优值	标准差	平均最优值	标准差
f_1	10	7.214 7e−022	1.312 6e−021	1.021 4e−014	4.156 9e−014	1.361 6e−015	2.578 4e−015
	20	3.919 1e−010	5.097 2e−010	1.351 3e−006	9.617 2e−006	1.017 6e−006	1.917 4e−006
	30	7.721 7e−006	1.243 1e−005	1.2811e−003	8.916 6e−004	1.601 4e−004	1.300 1e−004
f_2	10	1.176 9e−009	1.360 7e−009	1.669 3e−006	2.335 9e−006	2.179 3e−007	3.593 2e−007
	20	0.372 2	0.161 3	0.875 8	0.417 4	0.782 6	0.246 6
	30	5.500 1	1.164 1	5.601 8	1.382 1	5.523 5	1.273 2
f_3	10	−3 573.19	204.89	−3 404.18	212.305	−3 457.16	207.529
	20	−5 805.48	318.65	−5 777.93	368.774	−5 795.28	353.441
	30	−7 720.61	485.475	−7 670.53	483.917	−7 704.13	492.21
f_4	10	2.207 9	3.505 9	2.991 1	2.287 2	2.358 7	3.245 6
	20	16.661 1	5.923 6	18.645 2	6.735 1	17.562 3	7.229 2
	30	40.588 9	11.121 2	47.923 8	11.206 2	41.358 3	9.468 2

由表2可以看出:

1)最优值:在不同维数的函数测试中,随机参数PSO取得的平均最优值均低于另2种算法所取得的结果,特别是在单峰函数中,优化精度得到很大提高,表明本文提出的新算法具有较高的优化性能。

2)标准差:在不同维数的函数测试中,随机参数PSO的标准差几乎都小于另2种算法,表明新算法具有更高的稳定性。

从测试结果中可以得到,随机系数粒子群优化算法具有可行性。由于参数的设置是建立在具体的实验调查基础上的,因此保证了算法的有效性,而实验结果也证实了这一结论。

5 结束语

粒子群优化算法本质上是一种并行的全局性随机搜索算法,虽然只有3个可调参数,但很难确定出一套适合于所有优化问题的3个参数的取值组合。在应用PSO优化问题时经常从具体问题出发设置参数,并依靠经验通过大量测试、不断调整来确定最终参数取值,调整过程复杂并且效率低,对算法的进一步推广造成障碍。

本文在对PSO的3个重要参数进行数据实验和分析的基础上,提出了一种基于随机参数的PSO算法,让算法的3个控制参数均在一定范围内随机取值,以保持种群的多样性,平衡粒子局部和全局搜索能力,避免粒子由于某种惯性而导致收敛失败,提高全局优化能力。同时,更大范围内的参数取值,能够使群体在更大的搜索空间中进行寻优,避免了过早陷入局部最优,对提高全局收敛能力和优化最终解的质量起到关键作用。通过典型函数的测试结果表明,该随机参数PSO优化效率更高,并且稳定性好。由于新算法的可调参数是一组在各自取值范围内随机均匀取值的参数组合,因此无需考虑复杂的参数调整策略,这样既保持了PSO简单容易实现的优点又避免了因算法因参数设置不当而产生的误差,有利于算法的进一步推广和应用。

参考文献：

[1] Kennedy J,Eberhart R C. Particleswarm optimization[C]//Proc IEEE international conference on neural networks. Piscataway:IEEE Service Center,1995.

[2] Shi Y H,Eberhart R C. Amodified particle swarm optimizer [C]//Proc of the IEEE international conference on evolutionary computation. Piscataway:IEEE Service Center,1998.

[3] Zhan Z H,Zhang J. Adaptiveparticle swarm optimization[C]//The sixth international conference on ant colony optimization and swarm intelligence. Brussels:Springer,2008.

[4] 王克华,牛慧,张亚南.等.一种参数自适应调整和边界约束的粒子群算法[J].电子设计工程,2011,19(21):46-49.
Wang K H,Niu H,Zhang Y N,et al. Particle swarm optimization with adaptive parameters and boundary constraints[J]. Electronic Design Engineering,2011,19(21):46-49.

[5] 黄轩,张军,詹志辉.基于随机惯量权重的快速粒子群优化算法[J].计算机工程与设计,2009,30(3):647-650.
Huang X,Zhang J,Zhan Z H. The fast particle swarm optimization algorithm based on random inertia weight[J]. Computer Engineering and Design,2009,30(3):647-650.

[6] Suganthan P N. Particleswarm optimizer with neighborhood operator[C]//Proc of the congress on evolutionary computation. Washington:IEEE Service Center,1999.

[7] Ratnawecra A,Halgamuge S. Self-organizing Hierarchical-particle swarm optimizer with time-varying acceleration coefficients[J]. Evolutionary Computation,2004,8(3):240-255.

[8] 陈华,范宜仁,邓少贵.一种动态加速因子的自适应微粒群优化算法[J].中国石油大学学报:自然科学版,2010,34(6):173-175.
Chen H,Fan Y R,Deng S G. Adaptive particle swarm optimization algorithm with dynamic acceleration factor[J]. Journal of China University of Petroleum,2010,34(6):173-175.

[9] 黄少荣.基于随机加速系数的粒子群优化算法[J].微电子学与计算机,2010,27(6):114-117.
Huang S R. Particle swarm optimization algorithm based on the random acceleration coefficient[J]. Microelectronics & Computer,2010,27(6):114-117.

[10] 申元霞.自主式粒子群优化模型研究[J].重庆邮电大学学报:自然科学版,2009,21(4):507-511.
Shen Y X. Study on self-regulated model of particle swarm optimization[J]. Journal of Chongqing University of Posts and Telecommunications: Natural Science Edition,2009,21(4):507-511.

A New Particle Swarm Optimization with Random Parameters

HUANG Shao-rong

(Department of Information Management, Guangdong Justice Police Vocational College, Guangzhou 510520, China)

Abstract: Particle swarm optimization (PSO) is a powerful stochastic global technique, but easily trapped into local optimization, and its performance often depends heavily on the parameter settings. Based on analyzing the influence of the parameters setting in the experiment, this paper proposed a new particle swarm optimization algorithm which the inertia weight (ω) and acceleration coefficients (c_1 and c_2) are generated as random numbers within a certain range in each iteration process: ω=rand (0.4, 0.7), c_1=rand (0.5, 3.0), c_2=rand (1.0, 3.5). The proposed algorithms apply more particles' information, can easily jump out of local optimum and improve convergence performance. The experimental results demonstrate that the proposed algorithm is superior to the other two algorithms with a better astringency and stability.

Key words: particle swarm optimization (PSO); inertia weight; acceleration coefficients; random parameters

(责任编辑　游中胜)