

一种改进惯性权重的变异微粒群优化算法

蒋晓鸣, 雷霖, 王厚军

(电子科技大学 自动化工程学院, 四川 成都 610054)

摘要:针对微粒群优化算法的早熟收敛和进化后期收敛速度慢等问题,提出了一种改进惯性权重的变异微粒群优化算法。在算法运行过程中,对适应度值不同的微粒赋予不同的惯性权重,使算法既具有良好的空间探索能力又有良好的局部挖掘能力;在群体最优信息陷入停滞时引入变异算子,对聚集在局部最优微粒附近的微粒的位置和速度进行变异操作,使算法摆脱局部最优点的束缚。对4种典型函数的测试结果表明,新算法的全局搜索能力和收敛速度都得到了提高,并且能够有效避免早熟收敛问题。

关键词:微粒群;惯性权重;变异

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2008)06-0079-04

An Improved Inertia Weight Mutation Particle Swarm Optimization

JIANG Xiao-ming, LEI Lin, WANG Hou-jun

(School of Automation Engineering, University of Electronic Science and
Technology of China, Chengdu 610054, China)

Abstract: Proposes an improved inertia weight mutation particle swarm optimization to solve the premature convergence problem, and to avoid the slow convergence in the later convergence phase. When running the algorithm, different inertia weight values are given to particles according to their fitness. Thus the algorithm is engaged with both good exploration ability and good exploitation ability. When the optimum information of the swarm is stagnant, mutation operator is introduced to change the location and speed of the particles which are close to the local optimum position, and thus to reduce the possibility of trapping at the local optimum. According to the experimental results using four typical functions, the global searching ability and the speed of convergence of the new algorithm are both improved, and the premature convergence problem is effectively avoided.

Key words: particle swarm; inertia weight; mutation

0 引言

微粒群优化(particle swarm optimization, PSO)算法是1995年由Kennedy和Eberhart博士共同提出的一种新的群体智能优化算法,该算法具有概念简单、容易实现、参数较少、收敛速度较快、鲁棒性好等优点,目前已广泛应用于函数寻优、神经网络训练、模式分类、模糊系统控制以及其它一些工程领域^[1]。

与其他优化算法一样,微粒群算法亦有其不足,早熟收敛和进化后期收敛速度慢是PSO算法的两大难题。为此,研究人员提出了许多改进方法,比如:动态改变惯性权重^[2];线性改变加速因子^[3];协同微粒群优化算法^[4];引入多样性度量来保证微粒的多样性^[5];引

入变异操作提高全局搜索能力^[6]等。笔者在研究标准PSO算法的基础上,提出一种改进惯性权重的变异PSO优化算法IWMP SO。通过对4种标准函数测试,改进算法具有较快的全局收敛速度和良好的全局搜索能力,能比较有效避免早熟收敛问题。

1 改进惯性权重的变异PSO算法

在标准PSO算法运行过程中,某微粒若发现一个当前最优位置,其他微粒就会迅速向其靠拢,当此最优位置为一局部最优点时,微粒群就不能在解空间内重新搜索新的全局极值gBest,从而出现早熟收敛。理论上,PSO这种随机的基于群体搜索的优化算法必须测试空间内的每一个点,才能确保收敛于全局最优,然而这种做法带来的计算量极其巨大,是不现实的。因此,可以考虑在搜索初期使微粒尽可能飞跃整个搜索空间,以期在微粒位置上获得更好的多样性;当群体陷入早熟收敛时,采用变异机制使其跳出局部最优;在

收稿日期:2007-09-29

作者简介:蒋晓鸣(1981-),男,四川成都人,硕士研究生,从事检测技术及自动化控制研究;雷霖,教授,硕士生导师,从事检测技术、故障诊断、神经网络、混沌理论等研究。

进化后期,提高群体的收敛速度和精度,使其尽快收敛到全局最优解,这也是文中改进 PSO 算法的主要思路。

1.1 改进的惯性权重 ω

在微粒群算法的可调整参数中,惯性权重 ω 是最重要的参数,较大的 ω 有利于提高算法的全局搜索能力,而较小的 ω 会增强算法的局部搜索能力。文献[7]提出 ω 随算法迭代次数线性减小,将显著改善算法的性能^[2]。即令 ω_{\max} 为最大加权系数, ω_{\min} 为最小加权系数, $iter$ 为当前迭代次数, $iter_{\max}$ 为算法总迭代次数,则有:

$$\omega = \omega_{\max} - iter * \frac{(\omega_{\max} - \omega_{\min})}{iter_{\max}} \quad (1)$$

但在实际问题中,对于不同问题,其每一代所需的比例关系并不相同,所以, ω 线性递减只对某些问题很有效。此外,如果在进化初期搜索不到最优点,随着 ω 的逐渐减小,算法局部收敛能力加强,容易陷入局部最优;如果在进化初期探测到次好点,这时, ω 的相对取小就可使算法很快搜索到最优点,而 ω 的线性递减降低了算法的收敛速度。

因此,文中在 ω 线性递减的思路,对 ω 提出一种全新的改进方法,即让少部分微粒的 ω 在某些代并不跟随大部分微粒线性递减,而是产生突变,变为某固定值,并保持一定的代数。具体做法是:在算法运行中(以下考虑优化问题为“最小化”情况),当某微粒的 $f_i < f_{ave}$ 时,就有机会以概率 β 设置其 ω 为一新的值 ω_s ,并保持 ω_s 不变持续 M 代;当某微粒的 $f_i > f_{ave}$ 时,就有机会以概率 η 设置其 ω 为 ω_b ,并保持 ω_b 不变持续 N 代;未发生改变的大部分微粒 ω 则仍然每代线性递减(已发生突变的微粒在其保持代数内不再发生突变)。其中 f_i 为第 i 个微粒的适应度, f_{ave} 为微粒群目前的平均适应度,则有:

$$f_{ave} = \frac{1}{n} \sum_{i=1}^n f_i \quad (2)$$

其中 n 为微粒群的微粒数。 ω_s 在这里取一个相对较小的值,比如 0.2,其目的是让适应度优于平均适应度的这部分微粒有机会在已搜索到的最优解区域做局部挖掘(exploitation)工作,以便寻找到最优解。而 ω_b 取一个相对较大的值,比如 1.5,其目的是让适应度值相对较差的微粒做一个开发(exploration)工作,搜索更大的空间,以便发现新的解域。概率 β 和 η 可以取(0,1)之间的任意数值,一般取值为 0.2。对于 ω_s 保持的代数 M ,采用线性递增策略,即令 M_{\max} 为保持的最大代数, M_{\min} 为保持的最小代数, $iter$ 为当前迭代次数, $iter_{\max}$ 为算法总迭代次数,则有:

$$M = M_{\min} + iter * \frac{(M_{\max} - M_{\min})}{iter_{\max}} \quad (3)$$

而对 ω_b 保持的代数 N ,则采用线性递减策略,这样设置的原因是在算法运行初期,尽量让较多微粒保持较大的 ω 值以便探索更大的空间,在位置上获得更好的多样性;而在后期,让更多的微粒保持较小 ω 的值有利于局部最优解空间的挖掘,加快收敛速度。通过后面的实验结果显示, M 和 N 做这样的改进,显著提高了算法的收敛速度。

1.2 改进的变异机制

当 PSO 算法陷入早熟收敛时,微粒会出现“聚集”现象,发生“聚集”现象的微粒的适应度都相同,且速度 v 都很小,表现出强烈的“趋同性”。

文献[6]中提出 Multistart PSO 解决此问题,即每迭代若干次后,保留微粒群的历史最优位置,微粒全部重新初始化,以提高微粒的多样性,扩大搜索空间,跳出局部最优点的吸引。但微粒群全部初始化会破坏当前微粒的结构,使收敛速度减缓,而搜索精度也可能降低,体现不出 PSO 算法本身的优势。因此,借鉴该算法对微粒重新初始化的思想,对算法进行改进和优化,提出利用变异算子 P_m ,在全局最佳适应值 f_{best} 连续 G 代没有改进时,对聚集在 $gBest$ 周围的适应度值 f_i 等于 f_{best} 的微粒的速度和位置进行变异操作,即对这些微粒的速度和位置重新随机初始化,让这些微粒进入其它区域进行搜索,变异操作反复迭代执行,直到群体发现新的 f_{best} 为止。

文中对微粒的变异操作由随机算子 P_m 控制,即对满足变异条件的微粒按一定的概率 P_m 变异。 P_m 的计算公式如下:

$$P_m = \begin{cases} \lambda, & \text{if } f_i = f_{best} (i = 1, 2, \dots, m) \\ 0, & \text{others} \end{cases} \quad (4)$$

其中 λ 值越大,变异的微粒数目就越多,算法跳出局部最优点的能力也越强,但 λ 值过大对算法的稳定性有一定影响,一般 λ 的取值在 0.4 ~ 0.6 之间。

为了避免变异操作影响算法的稳定性,必须合理地选择代数 G 。若 G 太小,就有可能影响有潜力微粒的飞行,使得算法变为类似于随机搜索的算法;若 G 太大,将使微粒群处于停滞状态过久,削弱变异操作的作用。

综上所述,改进惯性权重的变异微粒群算法实际上是在微粒群优化算法的基本框架中,对微粒配置不同的惯性权重 ω ,使群体既具有空间拓展能力又有局部挖掘能力,并在算法陷入局部最优时,利用变异算子对聚集在 $gBest$ 周围的微粒的速度和位置进行变异,来提高算法跳出局部最优的能力。

2 仿真实验及性能分析

下面将通过4个典型函数优化问题来测试文中改进惯性权重的变异 PSO 算法(IWMP SO)的寻优性能,并与标准 PSO 算法(SPSO)和惯性权值递减的 PSO 算法(LDW)进行比较。

4个测试函数为:

F1: Sphere 函数。

$$f_1(x) = \sum_{i=1}^n x_i^2 \quad -100 \leq x_i \leq 100$$

F2: Griewank 函数。

$$f_2(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

$$-600 \leq x_i \leq 600$$

F3: Schaffer 函数。

$$f_3(x) = 0.5 - \frac{(\sin \sqrt{x_1^2 + x_2^2})^2 - 0.5}{(1.0 + 0.001(x_1^2 + x_2^2))^2}$$

$$-30 \leq x_1, x_2 \leq 30$$

F4: Ackley 函数。

$$f_4(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$$

$$-10 \leq x_i \leq 10$$

其中 $f_1(x)$ 是单峰二次函数, 函数 $f_2(x)$ 、 $f_3(x)$ 、 $f_4(x)$ 是具有大量局部极值点的多峰函数, 极难找到全局最优值。

实验设置的参数如下: 选取种群微粒数为 30; 测试函数维数为 10 维, $f_3(x)$ 函数为 2 维; 速度上限 v 取各函数初始范围的上限; $c_1 = c_2 = 2$; 惯性权重 $\omega_{\max} = 0.9$, $\omega_{\min} = 0.2$, $\omega_s = 0.2$, $\omega_b = 1.5$; 微粒惯性权重突变概率 $\eta = \beta = 0.2$; 突变后的惯性权重保持代数 $M_{\max} = N_{\max} = 50$, $M_{\min} = N_{\min} = 10$; 变异算子 P_m 中 $\lambda = 0.6$; 根据对 Sphere、Griewank 等测试函数的实验

结果(见表 1), G 取值在 10 ~ 30 比较合适, 本实验均采用 $G = 20$ 。为了全面比较优化效果, 以两种方式进行: (1) 迭代 1000 次比较最优适应值、最差适应值、平均最优适应值(mean best fitness, MBF)和平均计算时间(mean computing time, MCT), 当适应度值小于 $1.0e-20$ 时, 表示为 0; (2) 给定寻优应达到的精度和最大迭代次数(3000 次), 比较平均进化迭代次数、寻优率和平均计算时间。在测试时每种算法对每一个测试函数分别进行 100 次独立实验, 统计实验结果汇总如表 2、表 3 所示。

表 1 连续不变化代数 G 对算法性能的影响
(迭代次数 1000, 收敛精度 10^{-15})

G	搜索成功率	平均搜索用时(s)
2	0	33.5845
5	70%	28.7836
10	100%	22.0966
15	100%	21.7524
20	100%	21.2062
25	100%	21.8932
30	100%	21.5808
35	100%	24.5106

表 2 在给定精度下寻优率、进化代数和计算时间对比

算法	测试函数	收敛精度	寻优率	平均步数	MCT(s)
SPSO	Sphere	10^{-10}	0	3000	89.4065
	Griewank	10^{-10}	0	3000	90.0078
	Schaffer	10^{-10}	100%	787.3	7.6239
	Ackley	10^{-10}	0	3000	86.0532
LDW	Sphere	10^{-10}	100%	1470.5	42.9975
	Griewank	10^{-10}	100%	1269.8	42.8592
	Schaffer	10^{-10}	100%	397.2	3.0023
	Ackley	10^{-10}	100%	1918.4	56.6020
IWMP SO	Sphere	10^{-10}	100%	516.4	16.3656
	Griewank	10^{-10}	100%	367.6	12.1938
	Schaffer	10^{-10}	100%	28.6	0.8161
	Ackley	10^{-10}	100%	966	29.8590

表 3 在给定进化代数下适应值和计算时间对比

算法	测试函数	理论最优	BEST	WORST	MBF	MCT(s)
SPSO	Sphere	0	6.3164	14.1902	9.6632	30.5218
	Griewank	0	0.0014	0.0047	0.0027	32.5912
	Schaffer	0	0	0.00537	0.00142	8.4115
	Ackley	0	4.6102	5.1004	4.8613	34.4996
LDW	Sphere	0	6.8020e-017	1.0646e-015	4.2219e-016	29.9375
	Griewank	0	0	1.7932e-017	1.1102e-018	34.2610
	Schaffer	0	0	0.00126	0.00097	8.4239
	Ackley	0	1.1187e-008	3.6075e-008	2.1747e-008	36.3906
IWMP SO	Sphere	0	0	0	0	33.2855
	Griewank	0	0	0	0	33.5498
	Schaffer	0	0	0	0	9.3500
	Ackley	0	6.3248e-011	2.3120e-010	1.4926e-010	35.2030

由表 3 可以看出,在迭代 1000 次后,LDW 和 I-WMPSO 算法搜索到的最优解都远远优于 SPSO 算法,在收敛精度上 IWMP SO 算法更是优于 LDW 算法。对于 Griewank、Schaffer 等多峰函数,在全局极值周围,存在无限多个局部最优,因此极易陷入局部最优,当 SPSO 和 LDW 算法陷入局部最优解无法跳出时,IWMP SO 算法的变异机制使其能够迅速摆脱局部最优解的束缚,收敛到全局最优解。在计算时间上 IWMP SO 算法比 SPSO 和 LDW 算法只略慢一点,这是由于 I-WMPSO 算法结构简单,在变异机制中只判断最佳适应度是否连续代不变,在迭代运算中只涉及到初等运算,因此对运算量增加很少,计算效率高。由表 2 可以看出,SPSO 算法的性能最差,除了 Schaffer 函数,几乎不能在给定迭代次数下收敛到指定精度,LDW 和 I-WMPSO 都能 100% 收敛到指定精度;在收敛速度上 I-WMPSO 相较 LDW 和 SPSO 算法有了大幅提高,这源于 IWMP SO 算法中对不同微粒赋予不同的惯性权重 ω ,使得微粒在位置上具有良好的多样性,能避免群体陷入局部最优,特别是到了进化中后期,群体中拥有较小 ω 值的微粒数量逐渐增多,有效提高了算法的收敛速度。

3 结束语

针对微粒群优化算法的早熟收敛和进化后期收敛速度慢等问题,提出了一种改进惯性权重的变异微粒群优化算法(IWMP SO)。对 4 种典型函数的测试结果

表明,该算法的全局搜索能力和收敛速度都得到了提高,并且能够有效避免早熟收敛问题。

参考文献:

- [1] Shi Y, Eberhart R C. Particle swarm optimization: development applications and resource[C]//In: Proc congress on evolutionary computation 2001. Piscataway, NJ: IEEE Press, 2001:81-86.
- [2] Shi Y, Eberhart R C. Fuzzy adaptive particle swarm optimization[C]//Proceedings of the IEEE Conference on Evolutionary Computation. Soul:IEEE, 2001:101-106.
- [3] Ratnaweera A, Halgamuge S K. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients[J]. IEEE transactions on evolutionary computation, 2004,8(3):240-255.
- [4] van den Bergh F, Engelbrecht A P. Training product unit networks using cooperative particle swarm optimization[C]//Proc of the third Genetic and Evolutionary Computation Conference (GECCO). San Francisco, USA:[s. n.],2001:126-131.
- [5] Riget J, Vesterstroem J S. A diversity guided particle swarm optimizer - the ARPSO[R]. Aarhus: University of Aarhus, EVALife, 2002.
- [6] van den Bergh F. An analysis of particle swarm optimizers [D]. Pretoria, South Africa: Department of Computer Science, University of Pretoria, 2002:81-83.
- [7] Shi Y H, Eberhart R C. Empirical study of particle swarm optimization[C]//In:Proceedings of the 1999 congress on evolutionary computation. Piscataway, NJ: IEEE Service Center, 1999:1945-1950.

(上接第 78 页)

率图像(b)由原图像隔行隔列抽取得到,可以看出,最近邻插值边缘锯齿效应最明显,双线性插值放大图像虽然减轻了锯齿效应,但整体比较模糊,缺少细节,传统的 WaDi 插值算法较好地改善了双线性插值的模糊现象,而文中算法则得到了更清晰的边缘细节,头发和帽子边沿处都比较清晰。由此可见,无论从客观评价标准上还是主观视觉效果上文中算法都优于传统插值算法和传统的 WaDi 插值算法。

3 结 语

文中利用局部不对称特征和局部梯度特征相结合来计算 Warped Distance,这样较之传统 WaDi 插值算法,更多考虑了插值像素点周围的局部信息,得到了更为准确的插值像素值。

实验结果表明,文中算法能够提高插值精度,而且也容易实现。

参考文献:

- [1] XIN Li, Orchard M T. New edge-directed interpolation[J]. IEEE Transactions on Image Processing, 2001, 10(10):1521-1527.
- [2] 卢 钰. 基于小波的图像插值研究[J]. 武汉理工大学学报, 2003, 25(1):81-83.
- [3] Honda H, Haseyama M, Kitajima H. Fractal interpolation for natural images[C]//1999 International Conference on Image Processing. [s. l.]:[s. n.], 1999:657-661.
- [4] Ramponi G. Warped distance for space-variant linear image interpolation[J]. IEEE Trans Image Processing, 1999, 8(9): 1293-1297.
- [5] Hadhoud M M, Dessouky M I, El-Samir F E A. Adaptive image interpolation based on local activity levels[C]//Twentyeth National Radio Science Conference. Cairo:NRSC, 2003.
- [6] Hwang Jung Woo, Lee Hwang Soo. Adaptive image interpolation based on local gradient features[J]. IEEE Signal Processing Letters, 2004, 11(3):359-362.