

一种可逃逸的杂交粒子群算法

李 明, 曹德欣

(中国矿业大学 理学院, 江苏 徐州 221116)

摘 要: 针对标准粒子群优化算法在优化复杂函数时容易早熟, 收敛精度低等缺点, 根据遗传学中优良个体之间杂交产生优良后代概率大的特性, 提出一种改进方案, 由于个体最优位置包含的有用信息多于粒子当前位置, 在每一次迭代中, 对所得到的个体最优位置进行交叉操作, 以产生优良后代, 并当粒子群陷入早熟收敛时, 运用新的细菌觅食趋化操作使所有粒子在不破坏现有种群结构的情况下, 逐步摆脱局部最优的束缚. 将其应用于函数优化中, 得到了较好的优化效果.

关键词: 粒子群优化; 个体最优; 杂交; 模拟退火; 趋化; 非线性方程组

中图分类号: TP18

文献标识码: A

文章编号: 1000-7180(2012)12-0094-05

Escapable Hybrid Particle Swarm Optimization

LI Ming, CAO De-xin

(School of Science, China University of Mining and Technology, Xuzhou 221116, China)

Abstract: The particle swarm optimization algorithm has a few disadvantages in solving complex functions, including low solving precisions and high possibilities of being trapped in local optimum. According to genetics good parent often produce hybrid good offspring, an improved program is proposed, as the personal best position of the particle contains more useful information than the current position, in each iteration, crossover is operated to the best personal particle, a better position may be got, and when the particle swarm trap premature convergence, without destroying the existing population structure by using new bacterial feeding chemotaxis and all particles gradually get rid of the shackles of local optimum. Numerical examples show the effectiveness of the proposed algorithm.

Key words: particle swarm optimization; personal best; hybrid; simulated annealing; chemotaxis; nonlinear equations

1 引言

粒子群优化算法 (Particle Swarm Optimization, PSO) 是一种基于群体的优化算法. 与其它进化算法相比, 粒子群优化算法的优势在于收敛速度快, 没有复杂的交叉、变异等操作, 并且易于实现等.

文献[1]对算法的稳定性和收敛性作了初步分析, 指出基本 PSO 算法无法保证收敛到全局最优. 国内外的研究者针对该缺点已提出了多种改进的 PSO 算法^[2-3]. 本文对这些问题提出了一种可逃逸的杂交粒子群算法.

2 标准粒子群算法

在 PSO 算法的模型中, 优化问题的每个解对应搜索空间中的一个粒子, 每个粒子都有 2 个参数——位置和速度. PSO 算法首先随机初始化一群粒子, 然后通过迭代找到最优解或近似最优解. 在每次迭代中, 粒子通过追踪 2 个“极值”来更新自己的速度和位置: 个体极值和全局极值. 另外, 每个粒子还有一个由优化函数所决定的适应度, 以评价该粒子当前位置的优劣, 即解的优劣. PSO 算法的进化方程如下:

收稿日期: 2012-04-08; 修回日期: 2012-05-22

基金项目: 国家自然科学基金(70901073); 中央高校基本科研专项基金项目(JGK101676)

$$v_{id}^{t+1} = wv_{id}^t + c_1 r_1 (p_{id} - x_{id}^t) + c_2 r_2 (p_{gd} - x_{id}^t) \quad (1)$$

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

其中, t 为当前的进化代数; c_1, c_2 为学习因子, 通常取值为 2; r_1, r_2 为 $[0, 1]$ 之间的随机数; p_{id} 为粒子自己找到的最优解, 称为个体极值; p_{gd} 为整个粒子群目前找到的最优解, 称为全局极值. 文献[4]通过大量实验证明, 如果 w 随算法迭代的进行而线性减小, 将明显改善算法的收敛性能, 设 $w_{\max}, w_{\min}$ 分别为 w 的最大和最小值, $t, t_{\max}$ 分别为当前迭代次数及最大迭代次数, 则有

$$w = w_{\max} - (w_{\max} - w_{\min}) \times \frac{t}{t_{\max}} \quad (3)$$

上述算法为标准粒子群算法 (SPSO).

3 改进的粒子群优化算法 (EPSO)

(1) 个体最优杂交. 根据生物遗传学, 优良个体之间杂交产生优良后代的概率大于劣势个体之间的杂交. 现有的粒子群杂交算法, 都是针对粒子当前位置进行杂交, 而个体最优位置包含的有用信息比粒子当前位置多, 也就是包含的优秀维数比当前位置的多. 通过交换两个个体最优的某几维, 有可能得到更好的位置. 例如对 Sphere 函数来说, P_1, P_2 为两个个体最优.

P_1	1	3	0	2
		↑		↑
P_2	2	0	2	1

通过交叉, 得到两个新的位置:

P'_1	1	0	0	1
P'_2	2	3	2	2

P'_1 的值更接近全局最优 (0, 0, 0, 0), 得到了更好地个体最优位置, 计算 P'_1, P'_2 的适应值, 若优于原个体最优, 则更新个体最优并更新全局最优.

(2) 混合模拟退火算法和粒子群算法的细菌觅食趋化操作. 细菌觅食算法综述见文献[5], 在此不作详细介绍. 细菌觅食算法中的趋化操作包括翻转和前进, 翻转是细菌向任意方向移动单位步长, 在改进的粒子群算法中调用的趋化操作翻转更新公式为:

$$x_{id}^{t'} = x_{id}^t + c(i)\varphi(i) \quad (4)$$

$c(i)$ 表示第 i 个粒子的步长, $\varphi(i)$ 表示第 i 个粒

子的翻转方向. 每当粒子完成一次翻转后, 计算函数适应值 $f(x_{id}^{t'})$, 更新粒子个体最优和全局最优, 若 $\min\{1, \exp(-(f(x_{id}^{t'}) - f(x_{id}^t))/T)\} > \text{rand}$, 执行前进操作, 粒子将沿同一方向继续移动若干步, 如此循环直至适应度不再满足改善的条件, 或达到设定的移动步数临界值. 通过以上定义的操作, 所有粒子将逐步摆脱局部最优点的束缚, 避免了重新初始化破坏现有种群结构的缺点.

由于 PSO 是按照追随种群最优粒子的策略进行迭代更新的, 因此在运行过程中, 如果某粒子发现一个当前最优位置, 其它粒子将迅速向其靠拢. 此时种群多样性非常低, 如果该最优位置为一局部最优点, 粒子群就无法在解空间内重新搜索, 算法陷入局部最优, 出现早熟现象. 本文使用文献[6]定义的 D^p 来描述种群的多样性, 如下所示:

$$\bar{x} = \frac{1}{m} \sum_{i=1}^m x_{id} \quad (5)$$

$$D^p = \frac{1}{m} \sum_{i=1}^m |x_{id} - \bar{x}_d| \quad (6)$$

$$D^p = \frac{1}{n} \sum_{d=1}^n D_d^p \quad (7)$$

当种群多样性低于预定的阈值时, 所有粒子执行上述改进的趋化操作.

通过以上分析, 算法具体描述如下:

Step1: 随机初始化粒子群中粒子的位置和速度, 设置相关参数, 评价每个粒子的适应度.

Step2: 将粒子的 p_i 设置为当前位置, p_g 设置为初始群体中最佳粒子位置.

Step3: 判断算法是否满足收敛准则, 如果满足, 跳转到 Step9 执行, 否则执行 Step4.

Step4: 对粒子群中的所有粒子按公式 (1)、(2) 相继执行更新其速度和位置, 计算粒子的适应度, 更新粒子的个体最优和全局最优.

Step5: 判断算法是否满足收敛准则, 如果满足, 跳转到 Step9 执行, 否则执行 Step6.

Step6: 对所有粒子随机两两配对, 对配对粒子的个体历史最优位置, 随机选取半数维度交叉, 完成后, 计算其适应度, 更新个体最优和全局最优, 并检验是否满足结束条件, 若满足跳转到 Step9 执行.

Step7: 根据式 (5)~(7) 计算种群多样性 D^p , 若低于预先给定的阈值, 则执行混合模拟退火算法和粒子群算法的细菌觅食趋化操作.

Step8: 将迭代次数 t 增加 1, 并返回到 Step3 执行.

Step9: 输出全局最优, 算法结束.

4 优化性能测试

4.1 数值试验

为测试算法的性能,将 SPSO、文献[7]提出的 CIWPSO、本文提出的 EPSO 算法应用于求解七个被广泛使用的经典测试函数,如表 1 所示. 其中, f_1 、 f_2 为单峰函数, f_3 是很难极小化的病态二次函数, f_4 、 f_5 、 f_6 、 f_7 为多峰函数, f_7 是极为困难的多峰值函数,其包含无穷多个局部最优解和全局最优解,一般算法都很难找到全局最优点.

表 1 测试函数的范围、维数、Vmax、理论最优

函数	范围	维数	Vmax	理论最优值
Sphere	(-100,100)	30	50	0
Schwefel	(-30,30)	30	15	0
Rosenbrock	(-30,30)	30	15	0
Rastrigrin	(-5.12,5.12)	30	2.56	0
Griewank	(-600,600)	10	300	0
Ackley	(-30,30)	30	15	0
Shaffer	(-30,30)	2	15	0

实验在 MATLAB7. 8. 0(R2009a)软件环境下完成,计算机 CPU 为酷睿双核 2. 40 GHz 主频,内存为 2 GB. 实验的参数设置如下:算法的群体大小都设置成 30,最大进化代数设置为 2000,其中 f_7 进化代数为 1000,每种算法独立运行 30 次. 惯性权重 $w_{\min}=0.1$, $w_{\max}=0.9$, $c_1=2$, $c_2=2$, CIWPSO 算法采用文献[8]的取值 $c_1=c_2=1.5$, $T_{\max}=10000$, $T_{\min}=0.01$,温度冷却系数 $k=0.9$,步长 $c=1$,前进步数最大值 $ms=5$,每个粒子最大翻转次数 $mt=4$.

4.2 结果分析

从图 1~图 3、表 2 中可以看出,无论在低维还是高维函数优化中,本文提出的算法 EPSO 在 7 个函数优化问题中,无论是平均值、标准差、最大值、最小值,均取得了优于 SPSO 和 CIWPSO 的效果,特别是在 f_7 函数, EPSO 在 30 次实验中全部找到全局最优解.

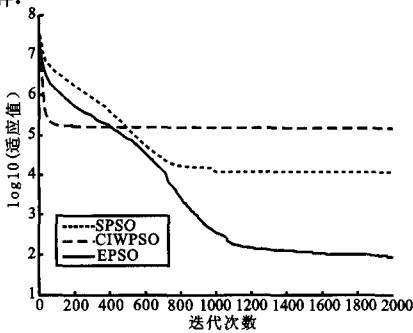


图 1 Rosenbrock 函数适应度曲线图

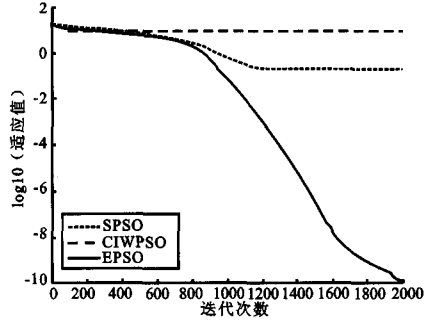


图 2 Ackley 函数适应度曲线图

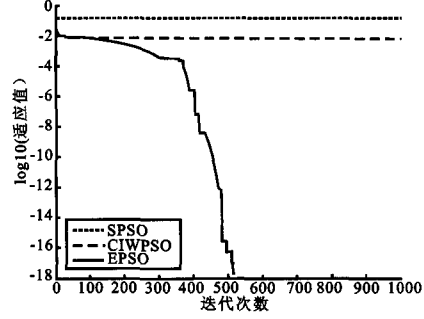


图 3 Shaffer 函数适应度曲线图

5 改进 PSO 算法求解非线性方程组问题

长期以来,人们在理论和数值计算方面对非线性方程组作了大量的研究,但是非线性方程组的求解还是困扰人们的难题,传统的牛顿迭代法仍然是用于解决该方程组用的最广泛的方法,但是这种类型的方法对初始值的依赖特别强,且是一种局部收敛方法,限制了其在实际工程领域里的广泛应用. 本节应用 EPSO 算法求解非线性方程组问题,我们采用文献[9]给出的具有高度的非线性的两个例子进行数值模拟,待求解的非线性方程组分别为:

(1)

$$\begin{cases} f_1(x)=x_1+x_2-2x_3=0 \\ f_2(x)=x_1x_2=1 \\ f_3(x)=x_1^2+x_2^2=2 \\ D_1=\{(x_1,x_2,x_3)|0\leq x_1,x_2,x_3\leq 2\} \end{cases}$$

(2)

$$\begin{cases} f_1(x)=x_1^2+x_2^1-5x_1x_2x_3-85=0 \\ f_2(x)=x_1^3-x_2^3-x_3^2-60=0 \\ f_3(x)=x_1^3+x_3^1-x_2-2=0 \\ D_2=\{(x_1,x_2,x_3)|0\leq x_1\leq 5,0\leq x_2\leq 4,0\leq x_3\leq 2\} \end{cases}$$

参数设置同 4 节,方程组(1)及(2)数值结果分别见表 3 (粗体标记为最好结果).

通过数值模拟可以看出,与 SPSO, CIWPSO 相比,本文的 EPSO 虽在精确解方面不及 SPSO,但在最后的平均适应值上均优于两种算法.

表 2 3 种算法在测试函数的实验结果的对比(粗体标记为最好结果)

函数名	算法	mean	std	max	min
f_1	SPSO	1.0901045e-10	5.5414293e-10	3.0401671e-9	1.0005002e-18
	CIWPSO	8.5082818e+2	4.1660592e+2	1.7976473e+3	8.2111649e+1
	EPSO	5.4867751e-17	2.0354932e-16	1.1028094e-15	8.4232992e-21
f_2	SPSO	2.1524623e+1	4.5106985e+1	1.4083835e+2	1.6816982e-7
	CIWPSO	1.00485403e+2	6.0390506e+1	2.1023241e+2	8.8862363
	EPSO	3.8788246e-10	9.3879148e-10	4.3207939e-9	1.9481879e-13
f_3	SPSO	1.22177558e+4	3.10493895e+4	9.00749781e+4	2.246403604e+1
	CIWPSO	1.53455369e+5	1.68806368e+5	5.81321402e+5	9.046563633e+3
	EPSO	8.6867177e+1	8.66019732e+1	4.41458456e+2	1.970649191e+1
f_4	SPSO	5.5374041e+1	1.9987974e+1	9.7505522e+1	1.8904212e+1
	CIWPSO	7.9773399e+1	2.1559961e+1	1.2288786e+2	4.7914208e+1
	EPSO	1.7528985e+1	6.6728153	4.2854198e+1	5.9702758
f_5	SPSO	6.7312963e-2	2.9293862e-2	1.3770491e-1	1.2316072e-2
	CIWPSO	7.2186547	4.3166175	2.2048122e+1	1.49433009
	EPSO	3.4868974e-2	2.0865421e-2	8.1142809e-2	0
f_6	SPSO	1.98063395e-1	4.69423951e-1	1.64622363	3.28580274e-9
	CIWPSO	8.61018168	1.68099132	1.1909201e+1	5.01058134
	EPSO	1.2429826e-10	8.45095996e-11	3.836087e-10	1.4368062e-11
f_7	SPSO	1.560881e-1	7.0787927e-2	2.9094995e-1	1.35421064e-2
	CIWPSO	6.1534095e-3	4.76208337e-3	9.71590987e-3	0
	EPSO	0	0	0	0

表 3 3 种算法在非线性方程组中的实验结果

		迭代结果平均值			平均适应值
		X_1	X_2	X_3	$\sum_{i=1}^3 f_i(x) $
例 1	SPSO	9.995057153e-1	1.000484348	9.999953992e-1	1.99159236e-5
	CIWPSO	1.003952624	9.95482755e-1	9.99716542e-1	1.13107252e-3
	EPSO	9.993775851e-1	1.0005774511	9.99999215e-1	3.5382615e-6
	精确解	1	1	1	0
例 2	SPSO	3.991591833	2.955389978	8.692927579e-1	4.849205675e-1
	CIWPSO	3.9860108246	2.954269245	8.482323656e-1	8.709756366e-1
	EPSO	4.0334960994	2.8996232589	9.305747207e-1	3.435883699e-1
	精确解	4	3	1	0

6 结束语

本文提出了一种改进的粒子群算法. 这种算法在每次迭代时对粒子的个体最优位置进行交叉操作, 并当粒子陷入早熟收敛时, 采用混合模拟退火算

法和粒子群算法的趋化操作, 在不破坏当前种群结构的情况下, 逐步摆脱局部最优的束缚, 较好的解决了提高全局寻优能力与加快收敛速度之间的矛盾, 同时将改进的算法应用于求解非线性方程组问题, 实验结果表明了算法的有效性.

参考文献:

- [1] Van den Bergh. An analysis of particle swarm optimizers[D]. South Africa: University of Pretoria, 2002.
- [2] 任小康,郝瑞芝,孙正兴,等. 基于单纯性法的量子粒子群优化算法[J]. 微电子学与计算机, 2010(1): 154-157.
- [3] Lovbjerg M, Rasmussen T K, Krink T. Hybrid particle swarm optimizer with breeding and subpopulations [C]//Proceedings of the Genetic and Evolutionary Computation Conference. San Francisco, USA, 2001, 469-476.
- [4] 陈应显,牛文庆. 改进粒子群算法的多峰值优化研究[J]. 微电子学与计算, 2011, 28(12): 59-62.
- [5] Shi Y, Eberhart R C. A modified particle swarm optimizer[C]//Proceedings of the IEEE International Conference on Evolutionary Computation. Anchorage, Alaska, 1998: 69-73.
- [6] Passino K M. Biomimicry of bacterial foraging for distributed optimization and control[J]. IEEE Control Systems Magazine, 2002, 22(3): 52-67.
- [7] Shi Cheng, Yuhui Shi. Diversity control in particle swarm optimization[C]//Proceedings of IEEE Swarm Intelligence Symposium 2011. Paris, France, 2011: 110-118.
- [8] 曾毅. 浮点遗传算法在非线性方程组求解的应用[J]. 华东交通大学学报, 2005, 22(1): 152-155.
- [9] 吴秋波,王允诚,赵秋亮,等. 混沌惯性权重调整策略的粒子群优化算法[J]. 计算机工程与应用, 2009, 45(7): 49-51.

作者简介:

李明 男, (1988—), 硕士研究生. 研究方向为智能算法、优化方法及应用.

曹德欣 男, (1962—), 教授, 博士生导师. 研究方向为计算数学、数值优化、智能算法等.

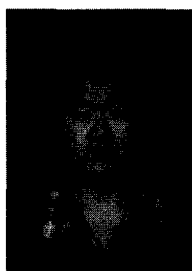
(上接第 93 页)

- [3] 肖欣招,胡春燕. 基于形态变换的图像增强方法研究[J]. 计算机与信息技术, 2007(7): 25-30.
- [4] 陈春宁,王延杰. 在频域中利用同态滤波增强图像对比度[J]. 微计算机信息, 2007, 23(2): 264-266.
- [5] 温学兵,赵江魏,梁学章. 基于小波去噪和直方图模板均衡化的手指静脉图像增强[J]. 吉林大学学报, 2008, 46(2): 291-292.
- [6] Kovacs-VajnaZs M, Rovatti R, Frazzoni M. Fingerprint ridge distance computation methodologies [J]. Pattern Recognition, 2000(33): 69-80.
- [7] 冷璐,黎明,张家树. 结合边缘细节局部自适应增强的直方图均衡化[J]. 微电子学与计算机, 2010, 27(1): 38-41.
- [8] 王建,庞宏伟. 基于 CLAHE 的 X 射线行李图像增强

[J]. 天津大学学报, 2010, 43(3): 194-198.

作者简介:

刘建科 男, (1966—), 博士研究生, 教授, 研究生导师. 研究方向为红外技术与工程、数字信号与图像处理.



李洋 男, (1988—), 硕士研究生. 研究方向为模式识别与智能系统、机器视觉.