

一种基于 PSO 和 GA 的混合算法

姚 坤¹, 李菲菲¹, 刘希玉²

YAO Kun¹, LI Fei-fei¹, LIU Xi-yu²

1. 山东师范大学 信息科学与工程学院, 济南 250014

2. 山东师范大学 管理学院, 济南 250014

1. School of Information Science and Engineering, Shandong Normal University, Jinan 250014, China

2. School of Management, Shandong Normal University, Jinan 250014, China

E-mail: yaokun@gmail.com

YAO Kun, LI Fei-fei, LIU Xi-yu. Hybrid algorithm based on PSO and GA. Computer Engineering and Applications, 2007, 43(6): 62-64.

Abstract: A novel PSO-GA Hybrid Algorithm (PGHA) is proposed, combining the strengths of particle swarm optimization with genetic algorithm. PGHA combines the update rules of velocity and situation of PSO's with the ideas of selection, crossover and mutation from GA. Through comparing PGHA to standard PSO in evolving solution to four standard functions, results show that PGHA outperforms PSO.

Key words: particle swarm optimization; Genetic Algorithm; PSO-GA hybrid algorithm

摘 要: 结合 PSO 算法和 GA 算法的优势, 提出了一种新颖的 PSO-GA 混合算法 (PGHA)。混合算法利用了 PSO 算法的速率和位置的更新规则, 并引入了 GA 算法里的选择、交叉和变异思想。通过混合算法对 4 个标准函数进行实验并与标准 PSO 算法比较, 结果表明混合算法表现出更好的性能。

关键词: 微粒群算法; 遗传算法; PSO-GA 混合算法

文章编号: 1002-8331(2007)06-0062-03 **文献标识码:** A **中图分类号:** TP18

1 引言

遗传算法 (GA) 是由 Holland 提出的一种进化计算模型^[1]。该算法将问题的解编码成为染色体, 通过对染色体的选择、交叉和变异等遗传操作, 产生并保留最优解。遗传算法被广泛用于函数优化、机器学习、智能控制、模式识别等领域。

微粒群优化算法 (PSO) 是由 Kennedy 和 Eberhart 提出的一种基于群体智能的计算模型^[2]。与遗传算法类似, 微粒群优化算法也是基于种群的算法。微粒群优化算法中每个微粒受到自身和种群中最优个体的影响不断调整自身的速率和位置并飞向全局最优解。与遗传算法相比, 微粒群算法具有更智能的背景, 实现起来更简单, 因此, 微粒群优化算法不但可用于科学研究也可用于工程应用。目前, 微粒群优化算法已经引起了进化计算、优化及其它领域的广泛关注^[3-6]。

对于微粒群优化算法的改进, 已经有人做了很多的研究, 也产生了一些微粒群优化算法和遗传算法的混合算法^[7-9]。这些算法初始时, 给每个微粒分配一个混合概率, 然后在每一次迭代过程中根据混合概率选择一定数目的微粒到一个交配池中, 池中的每个微粒随机搭配成对, 通过交叉操作每一对父个体产生两个子个体, 子个体用来替换父个体以保持种群数目不变。X.H.Shi, Y.H.Lu^[10]和 Matthew Settles, Terence Soule^[10]又对以上

算法进行了改进。PSO-GA 混合算法 (PGHA) 是利用了遗传算法的随机性以增加搜索范围, 然后利用微粒群算法在找到的最优微粒附近进行更细致的搜索。PGHA 对 4 个数值函数的优化结果表明, PGHA 的性能优于微粒群算法。

2 PSO 与 GA

2.1 PSO

标准 PSO 算法设想有一个由 m 个 D 维微粒所组成的群体^[2], 第 i 个微粒由一个 D 维的向量 $X_i (i=1, 2, \dots, m)$ 来表示, 它表明这个微粒在搜索空间中的位置为 $X_i (x_{i1}, x_{i2}, \dots, x_{id}) (i=1, 2, \dots, m)$, 每一个微粒的位置是一个潜在的解。通过把这个位置放在具体的目标函数中来计算微粒的适应值。如果适应值较高, 则相应的微粒就较好。第 i 个微粒的“飞行”速率也是一个 D 维向量, 表示为 $V_i (v_{i1}, v_{i2}, \dots, v_{id}) (i=1, 2, \dots, m)$ 。微粒 i 的历史最好位置记为 $P_i (p_{i1}, p_{i2}, \dots, p_{id})$, 群体中最好微粒的位置记为 $P_g (p_{g1}, p_{g2}, \dots, p_{gd})$, 标准 PSO 的位置和速率更新公式由下式表示:

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

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

混合算法中用到的是由 Clerc 提出的带约束因子的 PSO

基金项目: 国家自然科学基金 (the National Natural Science Foundation of China under Grant No.6037405); “泰山学者”建设工程专项经费资助。

作者简介: 姚坤 (1981-), 硕士研究生, 研究方向为群体智能、进化计算; 李菲菲 (1981-), 硕士研究生, 研究方向为群体智能; 刘希玉 (1964-), 博士, 教授, 博士生导师, “泰山学者”, 研究方向为人工神经网络与进化计算及其应用等。

算法^[11]。其速度和位置更新公式由式(3)和式(4)来定义。

$$v_{id}(t) = \chi(\omega v_{id}(t-1) - c_1 r_1 (p_{id} - x_{id}(t-1)) + c_2 r_2 (p_{gd} - x_{id}(t-1))) \quad (3)$$

$$x_{id}(t) = x_{id}(t-1) + v_{id}(t) \quad (4)$$

$$\chi = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}} \quad (5)$$

where, $\varphi = c_1 + c_2$, $\varphi > 4$

x_{id} —微粒 i 在第 d 维上的位置;

v_{id} —微粒 i 在第 d 维上的速率;

ω —惯性权重;

r_1, r_2 —内的随机数;

c_1, c_2 —个体和社会参数;

p_{id} —微粒 i 在第 d 维上的个体历史最好位置;

p_{gd} —全体微粒在第 d 维上的历史最好位置。

混合算法中的 φ 设为 4.1, χ 就为 0.73。根据 Yuhui Shi^[12] 的推荐, ω 设为 0.8 并线性递减到 0, 最大速率 V_{\max} 被设置为 X_{\max} 。

2.2 GA

对于具体问题, 标准 GA 将问题的解编码成为染色体^[1], 然后定义一个初始化种群, 种群中的个体代表问题的部分可行解, 搜索空间就是由这些不同的染色体所组成的解空间。搜索开始之前, 在搜索空间中随机地选择一系列染色体形成初始种群, 然后通过计算个体对具体目标函数的适应值, 以竞争方式来选择个体。通过依次使用诸如选择、变异和交叉等遗传操作来获得比前一代染色体适应值高的新一代的染色体。这个过程迭代进行, 直到满足结束条件, 最后一代的最好的染色体就是最优的解。

混合算法用到的 GA 采用文献[13]中的顺序小生境技术:

初始化规模为 N 的种群并计算个体适应值;

while(不满足停止规则)

将全部个体按适应值大小排序;

将排序后的种群按顺序组成 $\lfloor N/2 \rfloor$ 对父个体;

对每对父个体按概率进行交叉操作, 若子个体优于父个体则保留子个体, 否则保留父个体;

按概率做随机变异操作, 若优于原个体则保留新个体, 否则保留原个体;

重新计算适应值;

endwhile

由于每次交叉和变异后都和父个体、原个体进行比较, 只淘汰结构相似的父个体和原个体从而保持了种群的多样性。交叉操作采用微粒位置维度交叉, 交叉概率为 $Crossover_{ratio}$ ^[10], 如图 1 所示。

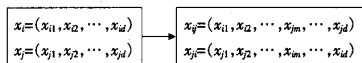


图1 交叉操作示意图

变异概率为 $Mutation_{ratio}$, 变异操作可能产生 3 种结果:

(1) 85% 的概率为:

$$gene_{mutated} = gene_{current} + X_{\max} * \varphi / 3$$

(2) 10% 的概率为:

$gene_{mutated} = gene_{current} * (M_A)^\varphi$, 其中 M_A 为变异改变参数, φ 为统一的随机数其范围是 $[0.0, 0.1]$ 。

(3) 5% 的概率为:

$gene_{mutated} = gene_{current} * \rho$, 其中 ρ 是一个统一的随机变量, 其变化范围是 $[-1, 1]$ 。

3 PSO-GA 混合算法(PGHA)

混合算法使用前面介绍的 PSO 算法和 GA: (1) 初始化数目为 $popsiz$ 的微粒群; (2) 计算微粒的适应值并保留最优微粒的位置和适应值; (3) 用混合概率 H_{ratio} 将微粒群分为两个子群; (4) 对 $popsiz * H_{ratio}$ 子群使用 PSO 算法; (5) 对剩余的子群使用 GA 算法。步骤(2)–(5)重复执行, 直到满足结束条件。算法流程图如图 2 所示。

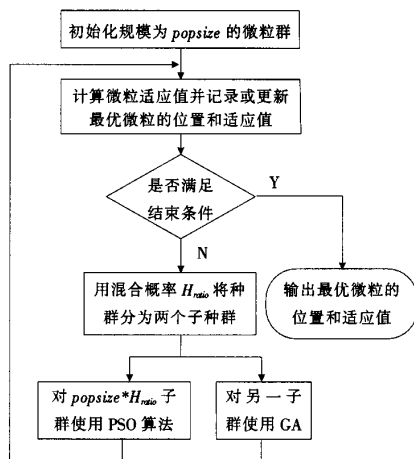


图2 PGHA 流程图

4 仿真实验

选取了两个数值最优化函数用来对 PGHA 与 PSO 的相关性能进行比较。这两个标准测试函数在其它的进化研究中经常使用。

$$f_1(x_2 - \cos 20\pi x + 1) + (y^2 - \cos 20\pi y + 1) \quad -5.12 \leq x, y \leq 5.12 \quad (6)$$

该函数是二维多峰函数, 在 $-5.12 \leq x, y \leq 5.12$ 范围内有 5 个山峰, 有 1 个极小值在 $(0, 0)$ 处, 该极小值为 0。

$$f_2 = \frac{\sin(\sqrt{(x-25)^2 + (y-25)^2} + 2.718)}{\sqrt{(x-25)^2 + (y-25)^2} + 2.718} \quad 0 \leq x, y \leq 50 \quad (7)$$

$$f_3 = e^{-(|x|+|y|)^{20}} \cos(|x|+|y|) \quad (8)$$

该函数随着坐标点偏离原点, 其值呈现多个局部极大值。其各峰值等值线从原点起呈菱形向外散布。取得全局最优值的坐标点只有 $(0, 0)$ 一点, 最优值为 1。

$$f_4 = \frac{\sin(\sqrt{(x-50)^2 + (y-50)^2} + e)}{\sqrt{(x-50)^2 + (y-50)^2} + e} + 1 \quad (9)$$

此函数在 $(50, 50)$ 处取得全局最大值 1.151 2, 其次优值为 1.128 37, 它是一个比前面几个函数更难的多峰优化函数。

PGHA 和 PSO 对于每个函数进行 20 次优化, 然后根据找到全局最优值的次数和找到的 20 个最优值的平均值来比较它们的性能。为了实验的简单起见, 把变异概率设为 0 即 $Mutation_{ratio} = 0$ 。详细的参数设置请参见表 1 和表 2。

表 3 所示是 PGHA 和 PSO 对 4 个函数进行 20 次优化找到全局最优值的次数和找到的 20 个最优值的平均值。表 4 是 PGHA 和 PSO 对函数 f_3 进行 20 次优化的具体数据。

表 3 的实验结果表明, 标准 PSO 对于多峰函数尤其对 f_2 和 f_4 这类特殊多峰函数是低效的。PGHA 对于函数 f_1 , 运行 20 次每一次都能找到全局最小值, 对于函数 f_2 和 f_4 , PGHA 的性能也要明显的优于标准 PSO。从表 4 可以看出, PSO 对于函

表 1 函数设置

函数	定义域	种群规模	迭代次数
f_1	$-5.12 \leq x, y \leq 5.12$	20	20 000
f_2	$0 \leq x, y \leq 50$	20	10 000
f_3	$-50 \leq x, y \leq 50$	40	10 000
f_4	$0 \leq x, y \leq 100$	60	30 000

表 2 参数设置

	PSO	PGHA
ω	0.8	0.8
c_1	2.1	2.1
c_2	2.0	2.0
χ	0.73	0.73
H_{ratio}	-	0.5
$Crossover_{ratio}$	-	0.8
$Mutation_{ratio}$	-	0

表 3 实验结果

f.	Standard		PSO PGHA	
	Find Global optima		Find Global optima	
	Times	Avg.	Times	Avg.
f_1	8	0.178 2	20	0
f_2	7	-0.027 65	18	-0.156 29
f_3	7	0.809 149 1	13	0.935 989 85
f_4	4	1.049 990 5	14	1.140 815 5

表 4 函数 f_3 的实验结果

Times	PSO's Optima	PGHA's Optima
1	0.882 088	1
2	1	1
3	1	1
4	0.605 044	1
5	0.882 088	1
6	1	1
7	1	0.685
8	0.686 059	0.648
9	0.851 595	1
10	0.777 923	1
11	0.686 059	1
12	0.439 88	1
13	0.439 88	1
14	1	1
15	0.663 799	0.882 088
16	1	1
17	0.918 709	1
18	0.663 799	0.858
19	0.686 059	0.918 709
20	1	0.876

数 f_3 极易陷入局部最优值，而 PGHA 的表现要明显好于标准 PSO。

5 结论

PGHA 是基于 GA 和 PSO 的一种混合进化算法，它利用了 GA 的搜索的随机性的同时保持了 PSO 的搜索性能。文章的最后通过对 4 个标准数值函数的优化，说明对于这个几个函数，PGHA 的性能要明显优于 PSO。为了进一步证明 PGHA 的性能，下一阶段要做的工作有：

- (1)利用 PGHA 对更多的函数进行实验以验证它的有效性。
 - (2)用带有变异操作的 PGHA 进行实验，即 $Mutation_{ratio} \neq 0$ 。
- (收稿日期：2006 年 10 月)

参考文献：

[1] Holland J H.Adaptation in natural and artificial system[M].[S.l.]: The University of Michigan Press,1975.

[2] Kennedy J,Eberha R C.Particle swarm optimization[C]//Proceedings of the IEEE International Conference on Neural Networks,1995: 1942-1948.

[3] 王玫,朱云龙,何小贤.群体智能研究综述[J].计算机工程,2005,31 (22):194-196.

[4] 陈国初,俞金寿.微粒群优化算法[J].信息与控制,2005,34(3):318-327.

[5] 张燕,汪镭,康琦,等.微粒群优化算法及其改进形式综述[J].计算机工程与应用,2005,41(2):1-3.

[6] 夏桂梅,曾建潮.微粒群算法的研究现状及发展趋势[J].山西师范大学学报,2005,19(1):23-25.

[7] Angeline P J.Evolutionary optimization versus particle swarm optimization: philosophy and performance differences [J].Evolutionary Programming,1998,7:601-610.

[8] Fan H Y.A modification to particle swarm optimization algorithm[J]. Engineering Computations,2002,19:970-989.

[9] Shi X H,Lu Y H.Hybrid evolutionary algorithm based on PSO and GA[C]//The 2003 Congress on Evolutionary Computation,2003, 4:2393-2399.

[10] Settles M,Soule T.Breeding swarms:a GA/PSO hybrid[C]//Proceedings of the 2005 Conference on Genetic and Evolutionary Computation,2005:161-168.

[11] Clerc M.The swarm and the queen:towards a determininistic and adaptive particle swarm optimization[C]//Congress on Evolutionary Computation,1999:1951-1957.

[12] Shi Yu-hui,Eberhart R C.Parameter selection in particle swarm optimization[J].Computer Science,1998,1447:591-600.

[13] 周北岳,郭观七.小生境技术对遗传算法的改进作用研究[J].岳阳师范学院学报,2001,14(4):18-21.

(上接 39 页)

querying[J].IEEE Trans on Pattern Analysis and Machine Intelligence,2002,24(8):1026-1038.

[3] Forstner W.A framework for low-level feature extraction[C]//Computer Vision-ECCV '94,1995:383-394.

[4] Kim D W, Lee K H, Lee D.A novel initialization scheme for the fuzzy c-means algorithm for color clustering[J].Pattern Recognition

Letters, 2004, 25(2):227-237.

[5] 何清法,李国杰.综合分块主色和相关回馈技术的图像检索方法[J].计算机辅助设计与图形学学报,2001,13(10).

[6] Pratt W K.数字图像处理[M].北京:机械工业出版社,2005:46-48.

[7] Mitchell T M.机器学习[M].北京:机械工业出版社,2003:136-140.

[8] 余鹏,封举富.基于高斯混合模型的纹理图像分割[J].中国图像图形学报,2005,10(3):282-283.