

文章编号:1001-9081(2008)04-0939-03

一种多精英保存策略的遗传算法

朱 灿^{1,2}, 梁昔明¹

(1. 中南大学 信息科学与工程学院, 长沙 410083; 2. 长沙理工大学 计算机与通信工程学院, 长沙 410076)

(zhucanplus@163.com)

摘 要:根据种子到当前最优点的距离将种群分成两部分, 小于或等于某一自适应距离值的种子归入当前最优种群, 大于该距离值的次优种子形成次优种群集合。对此两个种群分别按照不同的进化策略协同进化并重组。通过界定最优种群边界来提高遗传算法局部搜索能力, 通过对次优种群自适应变异, 比较好地平衡种群的“选择压力”和“种群多样性”。数值结果表明了本方法的有效性和稳定性。

关键词:遗传算法; 多精英保存; 种群划分; 进化策略; 自适应变异

中图分类号: TP301.6; TP182 **文献标志码:** A

Novel genetic algorithm with multi-elitist preservation method

ZHU Can^{1,2}, LIANG Xi-ming¹

(1. College of Information Science and Engineering, Central South University, Changsha Hunan 410083, China;

2. School of Computer & Communication Engineering, Changsha University of Science & Technology, Changsha Hunan 410076, China)

Abstract: A novel genetic algorithm with several elitists preserved was proposed. The population was divided into two parts according to the distances between individuals and the current individual. One was the optimal population of current generation, and the other aggregated hypo-opt individuals. This method can enhance local searching performance by bounding constrained optimal population and can raise the population diversity by introducing self-adaptive mutation probability in hypo-opt population. Some numerical tests have been made and the results show that the algorithm is effective.

Key words: genetic algorithms; multi-elitist preservation; population division; evolution strategy; adaptive mutation

0 引言

尽管保留最优个体的遗传算法(Genetic Algorithm, GA)以概率为 1 收敛于最优解, 但实际应用时, 遗传算法经常出现早熟和后期收敛速度太慢等现象。

很多文献对遗传算法做了改进。文献[1,2]结合一些搜索技术来优化遗传算法。常用的搜索技术有模拟退火算法、梯度算子、混沌优化以及神经网络优化等。但是很多搜索技术本身计算量大, 并且有些带有局部搜索性质, 对于一些带有参数的搜索方法来说对不同极小点附近的种子使用相同的参数不一定很合适。文献[3~8]提出了将自适应思想应用到遗传算法的各个环节, 包括设计自适应适应度函数, 改进交叉和变异算子, 自适应地改变种群数, 自适应地改变交叉变异概率以及引入小生境技术等。但是, 通过提高变异概率来提高种群多样性的同时, 也使得自适应参数变动比较大, 甚至整个进化变成随机进化。

1 遗传算法早熟原因分析

Whitley 认为, GA 中最重要的两个因素就是“种群多样性”和“选择压力”, 而选择压力过大是导致早熟收敛的一个重要原因^[9]。本文认为, 所谓“选择压力过大”, 其实质是适应度函数不准确。以最小化问题为例, 一般文献都将目标函数改为适应度函数。但实际上, 目标函数越小, 从函数值的角度来看, 该种子确实越靠近最优解; 但从自变量的角度看, 这个种子并非就越靠近全局最优解, 它也许是靠近局部最优点。

因此, 种群不是向最优点集中, 而是向一些局部极小点集中。随着进化的进行, 提前被选择的局部极小点附近的种子进化逐步完善, 优秀种子重抽样的增多, 使得一些靠近全局最优点的种子反而比靠近局部最优点的种子要差而进入选择池的机会减小, 加剧了适应度函数不准确带来的影响, 造成了早熟现象。

因此, 第一, 适应度函数设计时要适当考虑自变量, 有必要采取措施减小重复种子数, 增加后续的次优种子进入选择池的机会。

第二, 目前通常结合其他局部搜索技术或采用一些启发式算子来提高遗传算法的性能。尽管遗传算法是一种全局搜索技术, 但是遗传算法的交叉算子和变异算子(尤其是一些启发式交叉和变异算子)实际上都是针对自变量设计的, 具有局部性质。比如, 实数编码启发式交叉算子计算公式一般设计为 $x_1' = x_1 + rand \times (x_1 - x_2)$, 其中, x_1 的适应度值优于 x_2 。如果 x_1, x_2 不在同一局部极小点附近, 则这种交叉产生的新种子没有“启发”意义, 反而破坏了两个父代种子的基因模式, 其效果相当于变异, 就好像自然界牛和马杂交不会产生出一个更优秀的动物一样。因此, 根据目标函数值设计的选择算子与根据自变量设计的交叉和变异算子设计思路不一致。这种不一致减慢了遗传算法的收敛速度, 函数值相近的局部极小点越多, 影响越大。

第三, 目前对于种群大小没有一种合适的确定方法, 事实上, 在实际应用时, 增大种群数, 经常出现收敛速度反而减慢的情况。由于种群数增大, 种子分布更加分散, 启发式算子失

收稿日期:2007-10-15; 修回日期:2007-12-11。 基金项目:国家 973 规划项目(2002CB312200)。

作者简介:朱灿(1973-), 男, 湖南双峰人, 讲师, 博士研究生, 主要研究方向:演化算法; 梁昔明(1967-), 男, 湖南汨罗人, 教授, 博士生导师, 主要研究方向:工业过程稳态优化控制、大规模优化理论与算法。

去作用的可能性增大。

所以,本文提出将种群分成两个子种群的方法。通过以当前最优点为中心,将一定欧氏距离内的种子归入当前最优种群,以下简称种群 1,而将其其他种群归入次优种群集合,以下简称种群 2。对种群 1 里的种子在该欧氏距离界定的范围内按常规遗传算法进化;对种群 2 里的种子避免重复种子出现。其实质是通过设定一自适应的距离来隔离一个随当前最优变化的小生境,在小生境内部按小生境技术常规遗传算法进化来稳定选择压力,再在小生境外部通过大概率的变异和交叉算子来实现种群的多样性,以此来平衡“种群多样性”和“选择压力”。

2 改进的遗传算法

2.1 种群划分及进化策略

种群结构定义如图 1。

历史最优种子	
种群1	当代最优种子
	种群1其他种子
种群2种子(次优种子集合)	

图 1 种群结构

用编号为 1 的种子保存已经搜索结束的历史最优种子。其初值为初始种群的第一个种子,并随着进化逐渐改进,实际上是当前已经搜索完毕的一个局部极小点。历史最优不点不参与选择、交叉及变异等遗传操作。当前代最优种子 \bar{x} 是除去历史最优种子外的当前代最优种子。

$$\bar{x} = \min f(x_i); i = 1, 2, \dots, \text{popsize} \quad (1)$$

根据种子到 \bar{x} 的距离,将余下种子分为种群 1 和种群 2。两子种群的距离界限值 $dist$, 小于等于 $dist$ 的种子进入子种群 1, 大于 $dist$ 的种子进入种群 2。我们希望 \bar{x} 的 $dist$ 领域只包含一个局部极小点。但由于无法确定两个局部极小点之间的距离,因此用 $dist$ 来界定两个种群是不准确的。如果 $dist$ 比实际距离大,也就是说,种群 1 里还有超过 1 个局部极小点的种子,随着进化的进行, $dist$ 会逐渐减小,直至减小到剩下的种子都集中在一个局部极小点周围。每进化一代,将两个种群混合重组,重新划分成两个种群继续进化。固定两个种群的选择池大小,实际上也就是为了确定种群 1 里的种子数目,从而确保种群 2 有足够的种子数量来确保其多样性。

在数值实验里,作者对两个种群里的种子进行交叉,并对比了更新最优解的来源,发现:最优种子来源于种群 1 的相对比较多,来源于种群 2 的也有一部分,但来源于种群 1 和种群 2 的种子交叉的情况非常小。所以,没必要对两个种群进行交叉计算。更新历史最优解表明进化从一个局部最优解进化到一个更好的最优解。当代最优解更新来源于种群 2 说明了搜索范围在更换,种群 2 就像一个局部极小点仓库一样促使整个种群到一个更好的搜索范围里搜索。

2.2 选择算子

2.2.1 适应度函数

采用两个适应度函数来选择种子,按适应度函数 F_1 选择种子进入两个种群,按适应度函数 F_2 将种群种子排序。适应度函数 F_2 直接根据目标函数生成(本文以极小化问题为例)。

记 $norm$ 为欧氏距离函数, $precision$ 为计算精度, $popsize$, $popsize1$ 和 $popsize2$ 分别为总种群大小、种群 1 大小和种群 2 大小, k 为进化代数。

$$F_1: \begin{cases} \text{pop1: } norm(x, \bar{x}) \leq dist_k, dist_k > precision \\ \text{pop2: } norm(x, \bar{x}) > dist_k \\ \text{pop1: } norm(x, \bar{x}) > dist_k, dist_k \leq precision \\ \text{pop2: } \emptyset \end{cases} \quad (2)$$

$$dist_k = 0.5 \times \min(\max(norm(x, \bar{x})), \overrightarrow{norm(x, \bar{x})} \times (0.5popsize)) \quad (3)$$

其中, $\overrightarrow{norm(x, \bar{x})}$ 为 $norm(x, \bar{x})$ 数组升序排列。

另一个适应度函数为:

$$F_2 = f(x) \quad (4)$$

种群 1 里种群数不超过 $0.5popsize$, 如超过则按式(4)取适应度值排序最前的 $0.5popsize$ 个种子。由于种群 2 里有部分随机变异产生的种子,因此其适应度函数值很低,进化为最优的可能性小,种群 2 $pop2tmp$ 按式(4)排序,取前 $0.25popsize$ 个种子进入种群 2。

2.2.2 交配池大小及精英保存策略

两个种群里的种子按式(2)排序后,按轮盘赌方法进入交配池,轮盘赌选择概率计算公式为:

$$fit(i) = \frac{q(1-q)^{i-1}}{1-(1-q)^n} \quad (5)$$

交配池总大小标记为 $psize = popsize$, $n1$, $n2$ 分别为两种群交配池大小,显然,交配池大小不同于种群的大小,其计算公式为:

$$\begin{cases} n1 = 0.5popsize; n2 = 0.5popsize - 1, dist_k > precision \\ n1 = popsize - 1, n2 = 0, dist_k \leq precision \end{cases} \quad (6)$$

精英保存策略既要保存当前代最优种子,又要保存历史最优种子,而种群 2 实际上是保存距离当前代最优种子比较远而适应度函数值不一定比种群 1 里的种子要好的种子。当 $dist_k \leq precision$ 时,比较种群 1 最优种子与历史最优种子,如种群 1 最优种子优于历史最优种子,则用其替换历史最优种子。表明已经找到一个更优秀的局部极小点。当代最优种子按常规精英保存策略保存。历史最优种子更换公式为:

$$x(1) = \min(x(1), \bar{x}); dist_k \leq precision \quad (7)$$

2.3 交叉算子和变异算子

2.3.1 交叉变异策略

两种群的交叉算子采用启发式算子,变异算子采用一致和多重非一致变异算子。算子计算与常见遗传算法没有差别。只是对种群 1 的变异操作在 \bar{x} 的 $disk_k$ 邻域内进行,即种群 1 变异范围为: $bounds1 = (\bar{x} - dist_k, \bar{x} + dist_k)$; 而种群 2 的变异在整个编码空间变化。

2.3.2 自适应变异率

本方法种群 1 主要是用于提高平均适应度,提高收敛速度,而种群 2 的主要作用是保持种群多样性。所以种群 1 的进化以交叉为主,而种群 2 的进化以变异为主,其中种群 2 里要避免重复种子出现。所以,种群 1 里的交叉变异率依常规遗传算法确定,种群 2 里的变异率根据重复种子个数来确定。在轮盘赌选择时,将进入种群 2 交配池的重复种子置于交配池的后部,统计重复种子个数,用交叉和变异算子产生相应个数的不同种子,替换种群 2 后部的重复种子。

2.4 算法描述

对当前种群 1 按常规遗传算法进化,对种群 2 采用大概率变异、大概率交叉产生新的种子。每进化一代,将两种群混合起来,重新计算当前最优种子,重新划分两种群并继续进化。

本算法种群以实数编码,以目标函数为适应度函数。算法描述如下:

- 1) 确定种群规模 $popsiz$, 初始化种群;
- 2) 按式(1) 计算当前最优种子, 并计算编号为 $2 \sim popsiz$ 的种子到当前最优种子的欧氏距离;
- 3) 按式(3) 计算距离 $dist_k$, 如果 $dist_k \leq precision$, 则按式(7) 计算并替代历史最优点;
- 4) 按式(2) 生成种群1 和临时种群2, 并将临时种群2 里的种子按式(4) 排序并选择 $0.25 \times popsiz$ 个种子生成种群2;
- 5) 按式(6) 计算交配池规模; 并用轮盘赌方法产生种子进入一个或两个交配池; 对于种群2, 将重复种子置于交配池的后部, 统计重复种子个数;
- 6) 对两种群或一种群分别进行交叉和变异运算;
- 7) 检验是否进化结束, 如是则结束; 否则将种群1 和种群2 混合, 返回2)。

3 数值实验

测试函数1(Schaffer 函数):

$$\min f_1 = \frac{\sin^2 \sqrt{x^2 + y^2} - 0.5}{[1 + 0.001(x^2 + y^2)]^2} - 0.5;$$

$$-100 \leq x, y \leq 100$$

测试函数2(Shubert 函数):

$$\min f_2 = \sum_{i=1}^5 i \cos[(i+1)x + i] \times \sum_{i=1}^5 i \cos[(i+1)y + i];$$

$$x_i \in [-10, 10]$$

测试函数3(multi-modal 函数):

$$\min f_3 = (\cos(2\pi x) + \cos(2.5\pi x) - 2.1) \times (2.1 - \cos(3\pi y) - \cos(3.5\pi y));$$

$$(x, y) \in [0, 3] \times \cos(3.5\pi y)$$

测试函数4(六峰驼背函数):

$$\min f_4 = (4 - 2.1x^2 + x^4/3)x^2 + xy + (-4 + 4y^2)y^2;$$

$$(x, y) \in [-3, 3] \times [-2, 2]$$

测试函数5(Colville 函数):

$$\min f_5 = \sum_{i=1}^{n-1} [100 \times (x_i^2 - x_{i+1})^2 + (1 - x_i)^2];$$

$$-5.12 \leq x_i \leq 5.12, n = 20$$

测试函数说明参看文献[1,10,11]。种群数取100, 种群1 的交叉概率取0.8, 变异概率取0.03; 种群2 交叉概率取0.2, 变异概率自适应取值。重复运行本算法20次, 计算结果见表1。值得说明的是, 本文尝试在本算法开始前, 单独运行常规遗传算法数次, 以便使种群形成几个不同极小点附近的集合, 但数值实验证明该方法没有显著效果。

通过对测试函数1、5与文献[1], 测试函数2、3、4与文献[10] 计算结果比较分析, 本文提供的方法比文献中方法平均

收敛速度稍快(除了测试函数2 收敛较二阶段法慢外), 但数值更稳定, 基本上达到精确解。

表1 测试函数数值结果分析表

函数名/已知解	成功率/%	平均解	收敛代数
$f_1(-1)$	100	1.000000	187
$f_2(-186.7300)$	100	-186.730910	154
$f_3(-16.0910)$	100	-16.091720	36
$f_4(-1.0316)$	100	-1.031628	26
$f_5(0)$	100	0.000013	482

4 结语

本文分析了遗传算法早熟的成因, 根据种子到当前最优点的距离将种群结构划分为历史最优种子、当前最优种子和次优种子集合。通过界约束最优种群遗传算法来提高局部搜索速度; 通过固定两种群交配池大小来实现稳定最优种群数; 在次优种群进化计算里对常规轮盘赌选择算子也做了改进, 避免出现重复种子, 并通过保存次优种群增强了种群多样性。数值实验证明本方法数值稳定, 能显著避免早熟现象。

参考文献:

- [1] 袁晓辉, 袁艳斌, 王乘, 等. 一种新型的自适应混沌遗传算法[J]. 电子学报, 2006, 34(4): 708-712.
- [2] 洪露, 穆志纯, 王岗罡. 一种改进型混合遗传算法的分析[J]. 工业仪表与自动化装置, 2005, 34(3): 35-37.
- [3] 胡妙娟, 胡春, 钱锋. 遗传算法中选择策略的分析[J]. 计算机与数字工程, 2006, 34(3): 1-3, 57.
- [4] 苏小红, 杨博, 王亚东. 基于进化稳定策略的遗传算法[J]. 软件学报, 2003, 14(11): 1863-1868.
- [5] JANG W-H. Genetic/Quadratic search algorithm for plant economic optimizations using a process simulator[J]. Computers & Chemical Engineering, 2006, 30(2): 285-294.
- [6] 王秀坤, 赫志, 张晓峰. 一种改进的最优保存遗传算法[J]. 小型微型计算机系统, 2005, 26(4): 833-835.
- [7] 何新贵, 梁久祯. 利用目标函数梯度的遗传算法[J]. 软件学报, 2001, 12(7): 981-986.
- [8] 史亮, 李海鹰, 杨俊安, 等. 基于主动进化的遗传算法[J]. 小型微型计算机系统, 2004, 25(5): 790-793.
- [9] 车明, 孙晓华, 韩倩倩. 遗传算法与生物界进化相比存在的不足及改进[J]. 微处理机, 2006, 27(2): 53-54.
- [10] GUO CHONG-HUI, LU YU-CHANG, TANG HUAN-WEN. Modified evolutionary algorithm for global optimization[J]. Journal of Systems and Electronics, 2004, 15(1): 1-6.
- [11] (美)米凯利维茨 Z. 演化程序——遗传算法与数据编码的结合[M]. 周家驹, 何险峰, 译. 北京: 科学出版社, 2002: 257-260.

(上接第934页)

- [5] F'ARBER C. Sampling-Algorithmen f'ur stochastische Grammatiken zur Vorhersage vonRNA-Sekund'arstrukturen mit Pseudoknoten[D]. Frankfurt am Main, Germany: J. W. Goethe-Universit', Institut f'ur Informatik, 2005.
- [6] TAN GUANG-MING, FENG SHENG-ZHONG, SUN NING-HUI. An optimized and efficiently parallelized dynamic programming for RNA secondary structure prediction[J]. Journal of Software, 2006, 17(7): 1501-1509.
- [7] SCHMITZ M, STEGER G. Base-pair probability profiles of RNA secondary structures[J]. Computer Applications in the Biosciences, 1992, 8(4): 389-399.
- [8] SCHUSTER P. How to search for RNA structures: Theoretical concepts in evolutionary biotechnology [J]. Journal of Biotechnology, 1995, 41: 239-257.
- [9] KIRKPATRICK S, GELATT C D, VECCHI M P. Optimization by simulated annealing[J]. Science, 1983, 220: 671-680.
- [10] van BATENBURG F H D, GULTYAEV A P, PLEJJ C W A. An APL-programmed genetic algorithm for the prediction of RNA secondary structure[J]. Journal of Theoretical Biology, 1994, 174: 269-280.
- [11] 王翼飞, 史定华. 生物信息学——智能计算算法及其应用[M]. 北京: 化学工业出版社, 2006: 172-210.