文章编号：1000－8152(2004)04－0651－04

排挤小生态遗传算法的改进方法

谭竹梅，余晓峰，郭观七
（湖南理工学院 机械与电气工程系，湖南 岳阳 414000）

摘要：提出了基于搜索空间聚类分析的聚类排挤小生态遗传算法.通过分析适应值曲面的拓扑结构和扩大相似个体的搜索范围，聚类排挤可确定搜索空间的局部性，减少排挤的替换错误并抑制种群的遗传漂移；通过结合确定性替换和概率替换策略，聚类排挤提高了并行局部爬山能力和并行子种群维持能力.对不同多峰问题的仿真优化结果表明，聚类排挤小生态遗传算法的有效峰数量、平均峰值比和全局最优解比等综合性能一致地优于适应值共享、简单确定性排挤和概率排挤等小生态遗传算法.

关键词：遗传算法；小生态；排挤；聚类分析
中图分类号：TP301　　**文献标识码**：A

Improvement of niching genetic algorithms using crowding

TAN Zhu-mei, YU Xiao-feng, GUO Guan-qi
(Department of Mechanical and Electrical Engineering, Hunan Institute of Science and Technology, Yueyang Hunan 414000, China)

Abstract: A class of niching genetic algorithms using clustering crowding is proposed. By analyzing topology of fitness landscape and extending the space for searching similar individual, clustering crowding can determine the locality of search space more accurately, thus decreasing the replacement errors of crowding and suppressing genetic drift of the population. The integration of deterministic and probabilistic crowding increases the capacity of both parallel local hill-climbing and maintaining multiple subpopulations. The experimental results optimizing various multimodal functions show that, the performances such as the number of effective peaks, average peak ratio and global optimum ratio of genetic algorithms using clustering crowding are uniformly superior to that of the genetic algorithms using fitness sharing, simple deterministic crowding and probabilistic crowding.

Key words: genetic algorithm; niche; crowding; clustering analysis

1 引言(Introduction)

小生态技术是抑制进化计算遗传漂移现象的方法.在流行的小生态算法中，共享[1](fitness sharing, SH)存在小生态半径[2]参数化问题，且计算成本高.确定性排挤[3](deterministic crowding, DC)存在相似个体的判断错误和替换错误，遗传漂移的抑制能力较弱.虽然概率排挤[4](probabilistic crowding, PC)引入了低适应值物种的恢复压，但没有解决替换错误问题.因此，DC和PC均不能真正克服遗传漂移现象，难以满足搜索复杂多峰问题的应用要求.本文中研究基于搜索空间聚类分析的排挤小生态技术(clustering crowding, CC)，提出一套公平地评估小生态算法性能的测度准则，通过多峰问题仿真优化实验比较不同小生态遗传算法的综合性能.

2 搜索空间的聚类分析(Clustering analysis of search space)

山谷函数可用来模糊地确定搜索空间中任意两点是否属于相同的峰.一维山谷函数的例子如图1所示.图中 e_1 和 e_2 之间存在其适应值同时小于 e_1 和 e_2 适应值的内部点，二者属于两个不同的峰；e_3 和 e_4 之间不存在适应值同时小于二者适应值的中间点，二者属于相同的峰.由于 e_5 和 e_6 之间的点的适应值大于或小于二者的适应值，因此，应用山谷函数判断两点的峰所属关系时，其准确性取决于内部点的数量和位置.一般地，当适应值曲面具有较高的峰密度时，需要离散地选取多个点才能做出准确的判断.

在CC算法中，内部点的数量选为1.令 e_1 和 e_2 分别表示两个端点的坐标值，那么连线上内部点 i 可

收稿日期：2003－04－25；收修改稿日期：2003－11－19.
基金项目：湖南省教育厅科学研究基金项目(2001C380, 2002A052).

可按公式

$$i = e_1 + 0.5(e_2 - e_1)$$

计算,这样选择的内部点已能满足分析绝大多数测试函数曲面拓扑的准确性要求.对于高密度的多峰问题,虽然山谷函数不能百分之百地保证适应值曲面拓扑分析的准确性,但与根据距离确定相似个体的测度技术相比较,山谷函数显著地减少了判断错误,继而可达到减少排挤替换错误的目的.

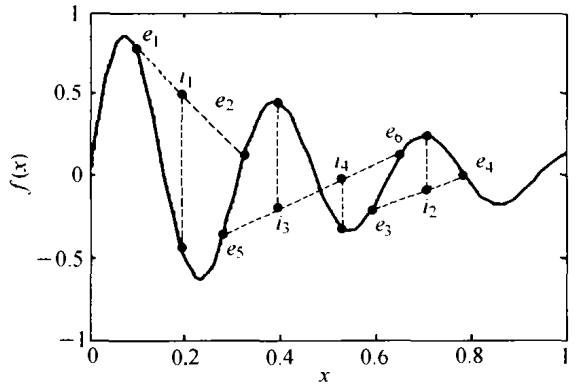

图 1 山谷函数的采样示意图

Fig. 1 Sampling map of hill-valley function

3 聚类排挤小生态遗传算法(Niching genetic algorithms using clustering crowding)

CC算法的伪代码如下:

ALGORITHM CC

Generate randomly a population P of size μ;

While (not fulfill stopping criterion)

$A = \{1,2,\cdots,\mu\}$; $B = \emptyset$; $P' = \emptyset$;

For ($k = 1; k \leqslant \mu/2; k = k + 1$)

i = a random integer in set $A - B$; $B = B \cup \{i\}$;

j = a random integer in set $A - B$; $B = B \cup \{j\}$;

(C_i, C_j) = mutation $\circ$ recombination (P_i, P_j);

$P' = P' \cup (C_i, C_j)$;

For ($i = 1; i \leqslant \mu; i = i + 1$)

Find out the nearest $P_j \mid j \in [1,\mu]$ to P'_i;

Analyze if P'_i and P_j belong to an identical peak by hill-valley function;

If (P'_i and P_j belong to an identical class and $f(P'_i) > f(P_j)$)

P'_i replace P_j;

Elseif ($\kappa \leqslant f(P'_i) / (f(P'_i) + f(P_i))$)

P'_i replace P_j;

End of the algorithm.

其中,$\emptyset$ 表示空集,$f > 0$ 表示目标问题的适应值函数,κ 表示[0,1]区间上均匀分布的随机变量,每次引用都采样一个新的随机数.

CC算法首先随机地生成规模为 μ 的初始种群 P,然后将全体个体随机地(不放回采样)配成 $\mu/2$ 对父代,所有配对的父代经重组和变异操作后生成子代个体种群 P'.对每个子代个体 $P'_i \mid i \in [1,\mu]$,从父代种群 P 中找到与之距离最近的个体 $P_j \mid j \in [1,\mu]$,然后应用山谷函数分析 P'_i 和 P_j 是否属于相同的峰,如果二者属于相同的峰且 P'_i 的适应值大于 P_j 的适应值,P'_i 替换 P_j.如果 P'_i 和 P_j 属于不同的峰,P'_i 以与其适应值成正比的概率替换 P_j.

4 性能准则(Performance criteria)

在设计小生态进化算法时,不但需要考虑算法并行地搜索多个局部最优解的能力,还需考虑所找到的局部最优解和全局最优解的质量(精度).因此,使用下列指标来测度算法的综合性能.

1) 有效峰数量(number of effective peaks maintained, NEC). 当搜索空间中某个峰在种群中存在至少一个元素,且该元素的适应值至少达到该峰所对应的局部极值的 80% 时,这样的峰称为一个有效峰.种群维持的有效峰的数量表示算法搜索到的有效局部最优解的个数,是对算法的并行搜索能力的测度.

2) 平均峰值比(average peak ratio, APR).种群中每个有效峰存在一个适应值最大的元素,该元素称为有效局部最优解,种群中全体有效局部最优解的适应值之和除以搜索空间中全体真实局部最优解的适应值之和称为平均峰值比,该指标是对种群中的多个有效局部最优解的平均质量的测度,理想值被规范化为 1.

3) 全局最优解比(global optimum ratio, GOR). 种群中适应值最大的有效局部最优解称为有效全局最优解,有效全局最优解的适应值与真实全局最优解的适应值之比称为全局最优解比,该指标是对种群中的有效全局最优解的质量的测度,理想值被规范化为 1.

5 优化实验(Optimization experiments)

5.1 测试函数(Test functions)

为测试和比较不同小生态算法的并行局部搜索能力和搜索速度(用 NEC, APR 和 GOR 等 3 个性能指标表示),采用下列 3 个不同难度的常用多峰函数:

$$f_1(x) = e^{-2(\ln 2)\left(\frac{x-0.08}{0.834}\right)^2} \sin^6(5\pi(x^{0.75} - 0.05)),$$

$$f_2(r,d,h)=\begin{cases}h-\dfrac{2hd^2}{r^2}, & d<\dfrac{r}{2},\\ \dfrac{2h(d-r)^2}{r^2}, & \dfrac{r}{2}\leqslant d<r,\\ 0, & \text{其他},\end{cases}$$

$$f_3(x_0,x_1,\cdots,x_{29})=\sum_{i=0}^{4}u\Big(\sum_{j=0}^{5}x_{6i+j}\Big),$$

f_1[5] 的定义域为[0,1],该函数有 5 个非均匀分布的不等高的峰,其局部极大点分别位于 $x=0.080$, 0.247,0.451,0.681 和 0.934,相应的极大值分别为 1.0.948,0.770,0.503 和 0.250.

f_2[6] 称为 Bell 函数,其中 r 表示铃铛状圆锥体底面的半径,h 表示高度,d 表示点到圆锥体底面中心的距离.通过改变搜索空间中圆锥体的数量、中心位置、大小和高度,该函数可提供不同的优化复杂度.在本文的优化实验中,30 个圆锥体底面随机地分布于定义域为 $x\in[0,1]$ 的二维空间中,每个圆锥体底面半径和圆锥体高度分别为区间[0.02,0.1]和[0.1,1]上的随机数.高度为 1 的最高圆锥体底面中心的坐标值为(0.76,0.61),在该点 Bell 函数取全局最大值 1.

f_3[7] 的自变量为长度为 30 位的二进制字符串,该字符串被均匀地分割成 5 个长度均为 6 位的子串,f_3 的函数值定义成 5 个子函数 $u(x)$ 的适应值之和.子函数 $u(x)$ 的自变量为 6 位二进制子串中"1"的位数,$u(x)$ 分别在点 $x=0$ 和 $x=6$ 取全局最大值 1;在点 $x=1$ 和 $x=5$ 取全局最小值 0;在点 $x=2$ 和 $x=4$ 取近似为 0.4 的中间值;在点 $x=3$ 取局部最大值 0.640576. f_3 被故意地设计成欺骗性的有极多峰的函数,该函数共有 5153632 个局部极值[8],其中有 32 个为全局最大值点,全局最大点的函数值等于 5,其它局部极值点的函数值位于 3.203 到 4.641 之间.

5.2 测试算法(Test algorithms)

对 4 种不同的小生态遗传算法进行测试,4 种算法分别为 SH,DC,PC 和 CC 小生态遗传算法.SH 采用适应值比例选择和通用随机采样,共享函数值调节指数[6] $\alpha=1$.所有算法均采用表现型的欧几里得距离测度,采用均匀随机变异和两点交叉算子,变异概率 $p_m=0.001$,交叉概率 $p_c=1$(这是 DC,PC 和 CC 隐含的交叉概率),其它运行参数如表 1 所示,其中,CC 的函数评估次数已包含了山谷函数进行拓扑分析所需要的评估次数.

表 1　测试算法对不同函数的非公共参数
Table 1　Uncommon parameters of test algorithms

函数	种群规模 μ	编码长度 l	小生态半径 σ_s(SH)	评估次数 E
f_1	30	30	0.1	10000
f_2	100	2×30	0.1	40000
f_3	100	30	0.2	50000

5.3 测试结果(Test results)

对每一个函数,4 种算法各进行 100 次独立的运行,每次运行完成表 1 设定的函数评估次数.实验结果用每次独立运行终止时的 NEC,APR 和 GOR 表示,100 次运行的算术平均值如表 2 所示.

表 2　100 次独立运行的性能指标算术平均值
Table 2　Average values of performance criteria for 100 independent runs

函数	性能指标	SH	DC	PC	CC
f_1	NEC	2.79	4.65	2.80	5.00
	APR	0.706	0.941	0.579	1.000
	GOR	0.994	0.993	0.992	1.000
f_2	NEC	10.47	14.22	4.41	27.37
	APR	0.455	0.656	0.201	0.945
	GOR	0.961	0.987	0.951	0.981
f_3	NEC	2.38	14.95	0.66	27.89
	APR	0.074	0.467	0.021	0.872
	GOR	1.000	1.000	0.963	1.000

CC 在 1×10^4 次的函数评估时间内均可靠地搜索到 f_1 的 5 个局部最优解.APR 和 GOR 值近似地等于 1 的事实说明由 CC 搜索到的所有有效局部最优解和有效全局最优解的质量接近真实的局部最优解和全局最优解.

CC 在 4×10^4 次的函数评估时间内平均地搜索到 f_2 的 30 个局部最优解中的 27 个,APR 值已达到 0.945,说明每个有效局部最优解的质量均接近真实的局部最优解.因此,CC 形成和维持有效峰的能力显著地优于 DC,SH 和 PC 算法.虽然 4 种算法的 GOR 指标比较接近,这并不表示它们具有相近的全局最优解搜索能力,这是因为计算 GOR 时并未判断有效全局最优解和真实全局最优解是否属于相同的峰,因此,有可能将某个具有最大适应值的有效局部最优解错误地当成有效全局最优解.分析 100 次运行终止时的种群分布可以发现,DC,SH 和 PC 算法分别有 6,20 和 33 次未生成全局有效峰,不同程度地存在丢失真实全局最优解的现象,但 CC 在 100 次运行中尚未发现一次类似的情况,因此,只有 CC 的 GOR 值才是全局最优解质量的真实反映.

Mahfoud[7]指出,无论采用多大的种群规模,在150万次函数评估时间内,SH均不能找到f_3的全部32个全局最优解.Mahfoud采用种群规模为600的DC和并行爬山法的混合算法,平均花费1.01×10^5次函数评估找到了全部32个全局最优解.Sareni[5]应用种群规模为100的清除过程,2×10^4次函数评估平均只找到了15个全局最优解,在同样的运行条件下,DC平均只找到了0.43个全局最优解.如表2所示,CC在5×10^4次的函数评估时间内平均地搜索到28个全局最优解,其全局最优解的形成和维持能力分别是DC,SH和PC的2~42倍.

观察有效峰数量平均值的动态曲线(限于篇幅,图略)还可发现,对于3个不同的优化问题,只有CC的动态曲线表现出先单调上升,后稳定的特征,其他3种算法的曲线或者在上升到一定阶段后呈现出下降趋势,或者呈现出明显的波动.这说明,只有CC能维持稳定的平衡态.对于3个测试函数,在平衡状态下,种群中所维持的有效峰数量分别为5,29.31和29.63(实际的局部最优解和全局最优解依次为5,30和32个),达到平衡态所需要的函数评估次数依次为4.4×10^3,5.76×10^4和9.13×10^4.

6 结束语(Conclusions)

CC通过扩大相似个体的搜索范围和应用山谷函数分析适应值曲面拓扑结构来提高相似个体判断的准确性,达到减少替换错误和抑制遗传漂移的目的.实验结果表明,CC能够自适应地、高效地形成和维持稳定的子种群,实现对搜索空间不同区域的并行搜索.

CC结构简单,应用时无需任何附加控制参数,是低成本的隐式自适应小生态技术.对SH,DC,PC和CC的综合性能进行测试和比较,结果表明,CC是最优的小生态算法.

当然,由于适应制曲面的拓扑分析需要函数评估的开销,与DC和PC相比较,在总的函数评估次数相同的前提下,CC用于优化搜索的评估次数较少,所以并行局部爬山速度较低,尚需研究改进这一缺陷的并行局部搜索技术.

参考文献(References):

[1] GOLDBERG D E, RICHARDSON J. Genetic algorithms with sharing for multimodal function optimization [C]// *Proc of the 2nd Int Conf on Genetic Algorithms*. Hillsdale, NJ: Lawrence Erlbaum, 1987: 41 - 49.

[2] JELASITY M, DOMBI T. GAS, a concept on modeling species in genetic algorithms [J]. *Artificial Intelligence*, 1998, 99(1): 1 - 19.

[3] MAHFOUD S W. Crowding and preselection revisited [C]// *Parallel Problem Solving from Nature*-2. Amsterdam: Elsevier, 1992: 27 - 36.

[4] MENGSHOEL O J, GOLDBERG D E. *Probabilistic crowding: deterministic crowding with probabilistic replacement* [R]. Urbana-Champaign: University of Illinois, IlliGAL Report No. 99004, 1999.

[5] SARENI B, KLÄHENBüHL L. Fitness sharing and niching methods revisited [J]. *IEEE Trans on Evolutionary Computation*, 1998, 2(3): 97 - 106.

[6] GAN J, WARWICK K. Dynamic niche clustering: a fuzzy variable radius niching technique for multimodal optimization in GAs [C]// *Proc of* 2001 *IEEE Int Conf on Evolutionary Computation*. Piscataway, NJ: IEEE Press, 2001: 215 - 222.

[7] MAHFOUD S W. *Niching methods for genetic algorithms* [D]. Urbana-Champaign: University of Illinois, 1995.

作者简介:

谭竹梅 (1963—),女,副教授,主要研究方向为进化算法、故障诊断、专家系统,E-mail: zhumei_tan@hotmail.com;

余晓峰 (1962—),男,主要研究方向为最优化、进化计算、软件工程;

郭观七 (1963—),男,教授,博士,主要研究方向为进化计算、自适应控制、人工智能,E-mail: guanqi_guo@hotmail.com.