

基于交叉模型的改进遗传算法

杨新武, 杨丽军

(北京工业大学 a. 计算机学院, b. 多媒体与智能软件技术北京市重点实验室,
c. 大规模流数据集成与分析技术北京市重点实验室, 北京 100124)

摘要: 提出一种解决早熟收敛问题的改进遗传算法. 通过最小生成树聚类将种群划分为若干个子种群, 子种群内的个体之间及不同子种群间的个体之间同时进行遗传操作. 同子种群间个体的遗传操作可以保证算法的进化方向和收敛速度, 不同子种群间个体的遗传操作可以避免近亲繁殖, 提供多样性. 分别采用二进制和实数编码, 在经典的 23 个基准函数上的对比测试结果表明, 所提出算法具有较好的收敛速度和寻优能力.

关键词: 遗传算法; 早熟收敛; 最小生成树聚类; 多样性

中图分类号: TP391

文献标志码: A

An improved genetic algorithm based on crossover model

YANG Xin-wu, YANG Li-jun

(a. College of Computer Science, b. Beijing Municipal Key Lab of Multimedia and Intelligent Software Technology, c. Beijing Key Laboratory on Integration and Analysis of Large Scale Stream Data, Beijing University of Technology, Beijing 100124, China. Correspondent: YANG Xin-wu, E-mail: yang_xinwu@bjut.edu.cn)

Abstract: An improved genetic algorithm is proposed for solving premature convergence. Firstly, the population is divided into several sub-populations by the minimum spanning tree clustering. Then, the genetic operation is performed among individuals within sub-population which ensures the evolution direction and speed, and that among individuals between different sub-populations which provides diversity by avoiding inbreeding. The experimental results on 23 benchmark functions using binary and real-valued representations show that the proposed algorithm has better convergence and faster speed to get the optimal solution.

Keywords: genetic algorithm; premature convergence; minimum spanning tree clustering; diversity

0 引言

遗传算法是一种并行的全局搜索优化方法, 通过模拟生物进化寻找最佳解. 与其他的局部搜索算法相比, 遗传算法具有更强的鲁棒性, 它能有效地探索巨大的复杂空间, 是求解优化问题的理想技术. 在资源分配、模式识别、机器学习、数据挖掘、优化和调度等领域均有大量的成功应用.

遗传算法的这种优势得益于它的操作是以种群为操作对象, 且种群本身具有维持种群多样性的优势. 种群中的每个个体都是所求问题的一个可行解, 整个种群就是一个可行解空间, 通过种群进化, 可以并行地搜索可行解空间中的可行解, 从而大大提升算法的搜索效率. 但遗传算法本身也存在着缺陷, 例如: 种群常常会出现早熟收敛^[1]、易陷入局部最优, 使算法的搜索性能大大降低. 这是因为在种群进化过程中, 随

着个体间的优胜劣汰, 使得大量的相似个体保存下来, 种群的多样性渐渐消失, 最后使得种群过早地收敛于某个个体, 由于收敛速度过快, 该个体往往不是全局最优. 此种情况下, 交叉操作的全局作用会被削弱, 从而不能有效地产生新的后代. 种群的进化操作将主要依靠局部变异来进行, 即在某一解的小的邻域范围内求解, 在这种情况下很难使得种群从局部最优解跳出来, 因而很难找到全局最优解. 这种现象在解决具有多峰的复杂问题上更为常见.

由于进化过程中种群多样性的缺失及收敛速度过快而出现的早熟收敛现象严重影响了遗传算法的搜索性能, 人们在应对早熟收敛问题上做了很多研究, 提出了许多方法, 这些方法包括参数控制^[2-7]、多种群的运用^[8-9]、双种群遗传算法^[10]、交配限制^[11-13]等. 参数控制方法主要是通过动态调整交叉概率或变

收稿日期: 2015-08-27; 修回日期: 2016-01-22.

基金项目: 国家自然科学基金重大研究计划培育项目(91546111); 北京市教委项目(PXM2015_014204_500221).

作者简介: 杨新武(1971—), 男, 副教授, 博士, 从事遗传算法、模式识别等研究; 杨丽军(1987—), 女, 硕士生, 从事遗传算法、数据挖掘的研究.

异概率等参数来改变遗传算法的搜索性能. 多种群主要是运用多个种群同时进化, 通过迁移来增加种群多样性. 双种群遗传算法可以认为是一种多种群遗传算法, 但又与其有所不同: 在多种群遗传算法中, 不同子种群的进化目标是相同的; 而在双种群中两个不同种群具有不同的进化目标, 各自有着自己的进化方向.

交配限制方法的思想是避免近亲繁殖, 该思想来源于生物学, 通过非近亲间个体的繁殖提高种群多样性从而避免早熟收敛. 徐晓艳提出了基于聚类的改进遗传算法(CGA)^[13], 其主要思想是对种群划分归类, 只选择异类间的个体进行遗传操作, 目的是避免近亲繁殖, 减少无效交叉. 该方法在一定程度上增加了种群多样性, 但是收敛速度有所下降. 针对该方法中的缺陷, 本文提出一个基于交叉模型的改进遗传算法(GACM). 该算法引进一个新的交叉模型, 根据种群中个体间的相似性度量, 将种群划分为若干个子种群, 以使每个子种群中的个体间具有较高的相似性, 不同子种群个体之间具有较低的相似性. 在进行遗传操作时, 选择同一子种群中的个体, 以及不同子种群中的个体同时进行交叉操作; 然后, 在产生的子代个体中进行生存竞争, 运用贪婪算法选出较优的个体. 同一子种群内个体之间进行遗传操作有助于优良性状的保留, 可以加快收敛速度; 不同子种群个体间的遗传操作可以引入多样性, 减少无效交叉. 两种方式产生的个体通过生存竞争, 优胜劣汰, 兼顾多样性和收敛速度, 进而有效抑制过早收敛. 分别采用二进制和实数编码, 在 23 个经典函数上进行了对比实验, 结果表明了GACM 的优越性.

1 交叉模型描述

交叉操作作为遗传算法的主要操作算子, 对算法的性能有着直接的影响. 因此, 算法成功的关键就是设计一个好的交叉操作. 文献[13]中的交叉操作仅限于异类间个体, 虽然可以减小无效交叉和增加多样性, 但没有考虑到仅有差异个体间的操作不能保证整个种群的收敛方向和收敛速度. 因此, 本文同时将同类个体间的遗传操作考虑进来, 并对两种操作产生的个体加入竞争机制, 设计一种新的交叉算子模型(CM), 即子种群内个体之间及子种群间个体之间同时进行交叉操作, 对产生的子代个体进行优胜劣汰, 选择其中适应度较强的个体作为下一代进化的个体. 子种群内相似个体之间的近亲繁殖在一定程度上可以保护优秀的基因模式, 维持个体的优良性状, 加快收敛速度, 但其缺点是如果过多地保护优良个体就会导致多样性缺失, 进而造成早熟收敛. 因此, 本文也同时让不同子种群间个体之间进行交叉操作, 不同子种群间个体之间具有一定的差异性, 能够避免近亲繁殖, 维持

种群多样性, 从而两者之间的合作与竞争关系可以同时兼顾到种群多样性与收敛速度之间的平衡.

交叉模型(CM)的算法步骤如下:

Step 1: 计算种群中任意两个个体间的欧氏距离, 构建邻接矩阵 D ;

Step 2: 利用普利姆算法求解邻接矩阵 D 的最小生成树 T ;

Step 3: 计算 T 的平均权值 \bar{W} , 取阈值 V 为 T 中小于 $\delta \times \bar{W}$ 的最大权值 ($0 < \delta < 1$);

Step 4: 遍历 T , 查找大于 V 的所有边并将其断开, 得到若干个子连通图;

Step 5: 遍历各子连通图, 得到子类, 并对子类编号保存;

Step 6: 轮盘赌选择一个个体 x_1 , 记下该个体所属的子种群的编号 i , 从子种群 i 中选择一个最好的个体 x_2 ;

Step 7: 选择与子种群 i 距离最远的子种群 j , 随机选择子种群 j 中的一个体 y ;

Step 8: 个体 x_1 与 x_2 进行交叉操作产生的个体的集合为 X ;

Step 9: 选择个体 x_1 与 x_2 中, 与个体 y 距离较远的个体, 假设为 x_2 , 个体 x_2 与个体 y 进行交叉操作产生的个体集合为 Y ;

Step 10: 通过贪婪算法选择集合 X 和集合 Y 中较优的个体作为子代个体.

2 GACM 算法描述

遗传算法主要包括 5 个基本要素: 参数编码, 初始群体的产生, 适应度函数的设计, 遗传操作(选择、交叉、变异)的设计, 控制参数的设定(群体规模、交叉概率、变异概率). 以下对几个要素进行介绍.

2.1 参数编码

编码是遗传算法解决实际问题时首先要解决的问题. 它对染色体的结构和解码方式起决定性的作用, 而且后续的交叉、变异等操作也会受它的影响^[14].

遗传算法的优点之一就是迭代计算不是针对变量本身, 而是变量编码后的个体, 通过对个体施加遗传操作从而求得全局最优解. 因此, 编码的长度及适当的编码策略将直接影响遗传算法的精度与效率^[14].

为了验证交叉模型(CM)不受编码机制的影响, 算法中分别采用二进制编码和实数编码. 所谓二进制编码, 即染色体为若干个为 0 或 1 的字符串; 所谓实数编码就是直接使用问题的变量进行编码, 染色体 x 的形式为 $x = (x_1, x_2, \dots, x_n)$, $x_i \in D_i$, $D_i \in R$, $i = 1, 2, \dots, n$. 其中: n 为变量的维数, D_i 为每个变量的定义域, R 为全体实数.

对于二进制编码而言,常用的交叉操作包括单点交叉、两点交叉、多点交叉等,在这里采用最常用的单点交叉.假设个体 A 与 B 为待交叉的两个父代个体,“|”为交叉点的位置,则有 $A: 1010|0100, B: 0110|001$; 单点交叉后的子代个体为 $A': 10100011, B': 01100100$. 对于实数编码而言,常用的交叉有平坦交叉和线性交叉等,在这里采用平坦交叉.假设 $p_1 = (x_{11}, x_{12}, \dots, x_{1n}), p_2 = (x_{21}, x_{22}, \dots, x_{2n})$, 为两个父代个体,所谓平坦交叉就是两个父代个体产生一个子代个体 $c = (x_1, x_2, \dots, x_n)$, 其中 x_i 为在区间 $[x_{1i}, x_{2i}]$ (假设 $x_{1i} > x_{2i}$) 中产生的一个均匀随机数.

2.2 初始群体的产生

遗传算法是在群体上操作的,因此在确定编码之后,就应该设计如何产生初始种群.在本文中,每一个染色体(即个体)的基因都是从其对应的定义域中随机产生,循环种群规模次后得到初始种群.

2.3 适应度函数的设计

适应度的概念在遗传算法中被用来表示个体趋近于最优解的优良程度^[14-15],高适应度的个体被遗传到下一代的概率较大,低适应度的个体被遗传到下一代的概率相对较小.

目标函数反映的是问题最初的求解目标,有时目标函数就是适应度函数,但是在演化计算中的某些选择算子(如轮盘赌选择)需要适应度函数是非负值,这时需要将目标函数转换成符合要求的适应度函数.

由于算法中采用轮盘赌选择,而经典的 23 个测试函数是最小化问题,并且这些测试函数中有的函数值为负值,需要对其转换.为了与文献[13]对比,依然采用如下转换形式:

$$F(f(x)) = \begin{cases} C_{\max} - f(x), & f(x) < C_{\max}; \\ 0, & f(x) \geq C_{\max}. \end{cases} \quad (1)$$

其中: C_{\max} 为一个恰当大的值,通常取值为当前代 $f(x)$ 的最大值.

2.4 遗传操作算子

选择、交叉和变异是遗传算法中的主要操作算子,下面对其进行介绍.

2.4.1 选择算子

所谓选择就是在某种规则下,优良个体被选择直接复制到下一代或者被配对交叉产生新个体,将新个体遗传到下一代,而劣质个体被淘汰的过程^[16].

本文选择算子采用轮盘赌选择法,也称比例选择法,其主要思想是个体被选中的概率与它的适应度成正比.因此在轮盘赌选择中,个体 i 被选中的概率为

$$p_s = f_i / \sum_{i=1}^N f_i. \quad (2)$$

其中: f_i 为个体 i 的适应度值; N 为种群中的个体数目,即种群规模.该选择算子虽然具有一定的随机性,但能够保证一定的进化度,使得适应度较高的个体被选择的概率更大,从而可使遗传算法向好的方向进化.

2.4.2 交叉算子

交叉算子在遗传算法中具有全局搜索的作用,是产生新个体的主要操作.在二进制编码下采用基于 CM 的单点交叉,在实数编码下采用基于 CM 的平坦交叉.

2.4.3 变异算子

变异算子在遗传算法中具有局部搜索的作用,决定算法的求精能力.基本位变异,即依概率对染色体的某个基因位进行变异,是遗传算法中常用的方法.在二进制编码下采用基本位变异,即 0 变 1, 1 变 0; 在实数编码下,采用的变异算子为随机变异,即某位基因变异为其对应定义域内的一个随机数.

2.5 GACM 算法伪代码

文献[13]中的 CGA 算法,采用自适应最小生成树聚类对种群进行划分,通过异类个体的交叉操作来增加多样性,避免近亲繁殖,减少无效交叉,但 CGA 只考虑了不同子种群间的操作,而忽略了子种群内的交叉操作,不能保证收敛速度和进化方向.对此,本文提出一个交叉模型(CM),该模型对每一代种群进行聚类,然后同时考虑同类个体之间与异类个体之间的遗传操作产生新的种群个体.具体操作如下:

Step 1: 对种群进行最小生成树聚类.

1) 计算种群中任意两两个体之间的欧氏距离,构建邻接矩阵 D ;

2) 利用普利姆算法求解 D 的最小生成树 T ;

3) 计算 T 的平均权值 \bar{W} , 取阈值 V 为 T 中小于 $\delta \times \bar{W}$ 的最大权值 ($0 < \delta < 1$);

4) 遍历 T , 查找大于 V 的所有边并将其断开,得到若干个子连通图,每个子连通图即为一个子类,对子类编号保存.

Step 2: 依交叉概率轮盘赌选择一个待交叉个体 p_{i1} , 并确定该个体的类别 C_i , 从 C_i 中选择最好的个体 p_{i2} 作为另一个待交叉个体,进行同类个体间的交叉操作; 选择距离 C_i 最远的一个类 C_j , 从中随机选择一个待交叉个体 p_{j1} , 与 p_{i1}, p_{i2} 中同 p_{j1} 差异较大的个体(假设为 p_{i1}) 进行异类间个体的交叉操作; 从两种操作产生的新个体中选择较优的个体作为子代个体.

Step 3: 判断是否达到种群规模.是则停止,否则转 Step 2.

在上述算法操作中,同类别内的个体之间差异性小,类内个体间的交叉可在一定程度上保护优秀的

基因模式, 维持个体的优良性状, 加快收敛速度; 异类个体之间差异性大, 异类个体间的交叉有助于维持种群的多样性. 因此, 两种遗传操作的同时进行可以兼顾收敛速度和种群多样性. 基于交叉模型的遗传算法(GACM)的具体流程如图 1 所示.

3 实验结果与分析

为了验证本文提出的交叉模型的有效性, 分别

在二进制编码和实数编码情况下, 对标准遗传算法 SGA、CGA^[13] 及本文算法 GACM, 在 23 个经典函数上对算法进行对比实验. 23 个函数的基本信息见表 1, 函数的其他详细信息见文献[17]. 实验环境: 操作系统为 Windows 7, 处理器 Inter(R) Core(TM) i5-3470 CPU @ 3.20 GHz, 内存 4GB, 算法运行环境为 VS2005. 算法中用的参数及含义如表 2 所示.

表 1 23 个测试函数信息

function	f_{\min}	D	S
$f_1(x) = \sum_{i=1}^D x_i^2$	0	30	$[-100, 100]^D$
$f_2(x) = \sum_{i=1}^D x_i + \prod_{i=1}^D x_i $	0	30	$[-10, 10]^D$
$f_3(x) = \sum_{i=1}^D \left(\sum_{j=1}^i x_j \right)^2$	0	30	$[-100, 100]^D$
$f_4(x) = \max\{ x_i , 1 \leq i \leq D\}$	0	30	$[-100, 100]^D$
$f_5(x) = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$	0	30	$[-30, 30]^D$
$f_6(x) = \sum_{i=1}^D (\lfloor x_i + 0.5 \rfloor)^2$	0	30	$[-100, 100]^D$
$f_7(x) = \sum_{i=1}^D ix_i^4 + \text{random}[0, 1)$	0	30	$[-1.28, 1.28]^D$
$f_8(x) = -\sum_{i=1}^D (x_i \sin \sqrt{x_i})$	-12 569.5	30	$[-500, 500]^D$
$f_9(x) = 10D + \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i))$	0	30	$[-5.12, 5.12]^D$
$f_{10}(x) = 20 + e - 20 \exp\left(-\frac{1}{5} \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i)\right)$	0	30	$[-32, 32]^D$
$f_{11}(x) = 1 + \sum_{i=1}^D x_i^2 / 4000 - \prod_{i=1}^D \cos(x_i / \sqrt{i})$	0	30	$[-600, 600]^D$
$f_{12}(x) = \frac{\pi}{D} (10 \sin^2(\pi y_1) + \sum_{i=1}^{D-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})]) (y_n - 1)^2 + \sum_{i=1}^D (u(x_i, 10, 100, 4))$	0	30	$[-50, 50]^D$
$f_{13}(x) = 0.1 \left[\sin^2(3\pi x_1) + (x_D - 1)^2 (1 + \sin^2(2\pi x_D)) + \sum_{i=1}^{D-1} (x_i - 1)^2 (1 + \sin^2(3\pi x_{i+1})) \right] + \sum_{i=1}^D u(x_i, 5, 100, 4)$	0	30	$[-50, 50]^D$
$f_{14}(x) = \left(1/500 + \sum_{j=1}^{25} \left(j + \sum_{i=1}^2 (x_i - a_{ij})^6 \right)^{-1} \right)^{-1}$	1	2	$[-65.536, 65.536]^D$
$f_{15}(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4} \right]^2$	0.000 307 5	4	$[-5, 5]^D$
$f_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	-1.031 628 5	2	$[-5, 5]^D$
$f_{17}(x) = \left(x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6 \right) + 10 \left(1 - \frac{1}{8\pi} \right) \cos x_1 + 10$	0.398	2	$[-5, 10] \times [0, 15]$
$f_{18}(x) = [1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	3	2	$[-2, 2]^D$
$f_{19}(x) = -\sum_{i=1}^4 c_i \exp \left[-\sum_{j=1}^4 a_{ij} (x_j - p_{ij})^2 \right]$	-3.86	4	$[0, 1]^D$
$f_{20}(x) = -\sum_{i=1}^4 c_i \exp \left[-\sum_{j=1}^6 a_{ij} (x_j - p_{ij})^2 \right]$	-3.32	6	$[0, 1]^D$
$f_{21}(x) = -\sum_{i=1}^7 [(x - a_i)(x - a_i)^T + c_i]^{-1}$	-10	4	$[0, 10]^D$
$f_{22}(x) = -\sum_{i=1}^7 [(x - a_i)(x - a_i)^T + c_i]^{-1}$	-10	4	$[0, 10]^D$
$f_{23}(x) = -\sum_{i=1}^{10} [(x - a_i)(x - a_i)^T + c_i]^{-1}$	-10	4	$[0, 10]^D$

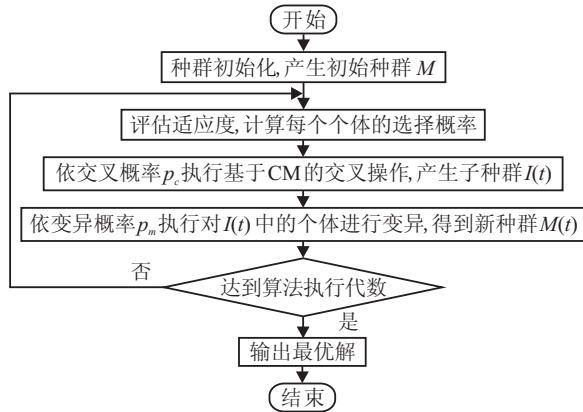


图 1 算法 GACM 的流程

表 2 算法中的参数

参数名称	参数含义	参数值
popsize	种群规模	100
varnumber	变量维数	30
perchrom	每个维数的编码长度	15
maxstring	总编码长度	varnumber × perchrom
p_c	交叉概率	1.0
p_m	变异概率	1.0/maxstring
factor	聚类系数	0.99

3.1 二进制编码下的实验

算法 SGA、CGA、GACM 的遗传操作说明如表 3 所示。算法运行 50 次，实验评价指标为最优平均值 avg 及标准差 stdev，实验结果如表 4 所示。从表 4 中可以看出，在 23 个经典函数上，GACM 的平均值及标准差都好于或接近于 SGA 和 CGA，结果有力地证明了本文的交叉模型对 CGA 的改进是有效的，算法的性能得以提高。

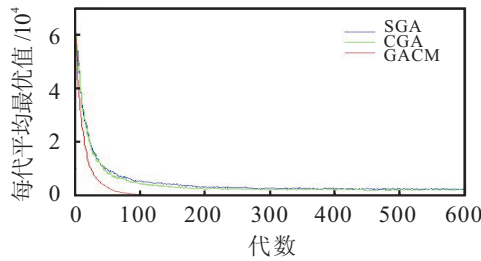
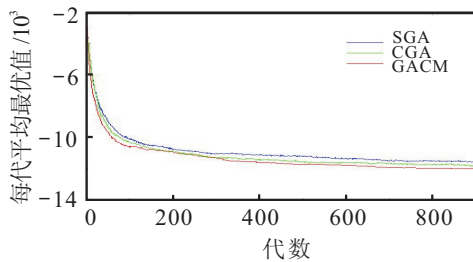
表 3 算法 SGA、CGA、GACM 的遗传操作

算法	选择算子	交叉算子	变异算子
SGA	轮盘赌	单点交叉	基本位变异
CGA	轮盘赌	异类个体间的单点交叉	基本位变异
GACM	轮盘赌	基于 CM 单点交叉	基本位变异

为了进一步说明 CM 确实改进了遗传算法的搜索性能，在高维单峰函数 f_1 和高维多峰函数 f_8 上进行 10 次实验，分别运行 600 代、900 代，求得每代平均最优值来验证改进后算法的搜索效率，如图 2、图 3 所示。显然，算法 GACM 的收敛速度好于 CGA。

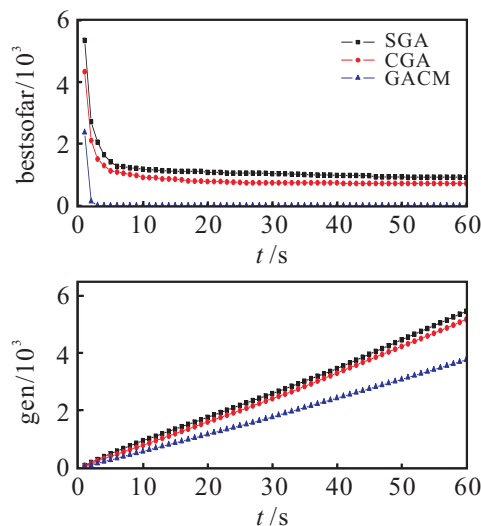
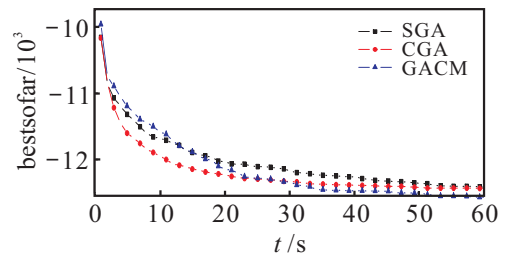
表 4 二进制编码下 SGA、CGA、GACM 的实验结果

function/optimal	gen	SGA		CGA		GACM	
		avg	stdev	avg	stdev	avg	stdev
$f_1/0$	1500	1.27e+02	1.90e+02	8.49e+02	1.43e+02	2.82e-04	1.46e-05
$f_2/0$	2000	5.81e+01	4.77e+00	5.51e+01	4.63e+00	9.16e-03	0
$f_3/0$	5000	2.09e+04	2.42e+03	2.24e+04	2.35e+03	4.84e+03	1.59e+03
$f_4/0$	5000	2.56e+01	2.50e+00	2.94e+01	2.94e+00	5.28e-01	2.29e-01
$f_5/0$	20000	7.12e+05	1.41e+05	5.69e+05	1.13e+05	4.27e+01	3.33e+01
$f_6/0$	1500	1.10e+03	1.86e+02	8.66e+02	1.43e+02	0	0
$f_7/0$	3000	9.98e-01	2.08e-01	9.09e-01	1.87e-01	5.07e-02	1.17e-02
$f_8/-12569.5$	9000	-12430.0	3.85e+01	-12464.8	1.32e+01	-12567.9	2.67e-01
$f_9/0$	5000	2.41e+01	4.09e+00	2.14e+01	3.02e+00	1.27e+01	5.18e+00
$f_{10}/0$	1500	4.71e+00	1.70e-01	3.83e+00	2.15e-01	1.33e-02	6.49e-02
$f_{11}/0$	2000	1.05e+01	1.43e+00	8.26e+00	1.05e+00	4.51e-02	4.59e-02
$f_{12}/0$	1500	5.22e+05	2.85e+05	4.37e+05	2.86e+05	4.51e-01	1.70e-01
$f_{13}/0$	1500	2.51e+06	1.01e+06	2.24e+06	8.76e+05	1.63e+00	2.69e-01
$f_{14}/1$	100	1.128	2.23e-01	1.139	2.58e-01	1.028	7.49e-02
$f_{15}/3.075e-04$	4000	9.77e-04	1.87e-04	1.05e-03	2.13e-04	1.20e-03	2.92e-03
$f_{16}/-1.0316285$	100	-1.024900	8.35e-03	-1.025881	5.13e-03	-1.031593	1.27e-04
$f_{17}/0.398$	100	0.401	3.83e-03	0.400	2.39e-03	0.398	3.71e-05
$f_{18}/3$	100	3.194178	2.25e-01	3.150050	1.51e-01	3.00005	2.37e-04
$f_{19}/-3.86$	100	-3.86	8.05e-04	-3.86	3.31e-04	-3.86	1.70e-05
$f_{20}/-3.32$	200	-3.27	5.29e-02	-3.29	4.28e-02	-3.27	5.66e-02
$f_{21}/-10$	100	-5.35	3.14e+00	-7.26	3.21e+00	-6.04	3.19e+00
$f_{22}/-10$	100	-6.12	3.43e+00	-7.62	3.46e+00	-7.00	3.12e+00
$f_{23}/-10$	100	-5.83	3.10e+00	-8.44	3.17e+00	-6.12	3.46e+00

图2 f_1 的实验结果(1)图3 f_8 的实验结果(1)

由于GACM在每次迭代时加入了聚类操作,选择同子类内的个体及不同子类内的个体进行交叉,相对于标准遗传算法SGA增加了计算量,但从相同代数的实验结果可以看出,GACM在收敛速度和寻优能力上都有了提高。

为了进一步证明算法的性能,又分别在高维单峰函数 f_1 和高维多峰函数 f_8 上进行了以时间为横坐标的性能恒量实验。实验运行10次,进化停止的条件是60s。每次实验都记录了每间隔1s种群所进化的代数gen,以及到此刻此代数为止搜索到的当前最优解bestsofar,然后取10次结果的代数和当前最优解的平均值,所得到的实验结果如图4、图5所示。从图4和图5中可以明显地看出,在相同时间内,GACM运行的代数比SGA、CGA要少,说明聚类操作确实增加了每次迭代的计算量;但还可以发现,GACM在相同的时间内以较少的运行代数搜索到的最优解是3种方法中最佳的。

图4 f_1 的实验结果(2)图5 f_8 的实验结果(2)

对于图5中前20s内GACM搜索的结果不如其他两个算法好,而之后的结果很快超越前两者的现象,其原因是由于GACM算法添加了额外的计算量,20s内进化的代数不够。此外, f_8 是高维多峰函数,存在很多个局部最优值。一方面是由于计算量大进化的代数不够;另一方面是由于陷入了某个极值。20s后GACM算法在搜索中有了一次跳跃,跳出局部极值之后搜索加速,这在一定程度上证明了GACM较好地兼顾了收敛速度和种群多样性。综上分析,GACM虽然增加了计算量,但是在寻优的质量上有了提高,即可以找到更好的解。

3.2 实数编码下的实验

为了进一步验证交叉模型的有效性,在实数编码下进行了实验,算法SGA、CGA与GACM的遗传操作说明如表5所示。算法运行50次,实验评价指标为最优平均值avg及标准差stdev,实验结果如表6所示。从表6中可以看出,在23个经典函数上,GACM的平均值和标准差依然好于或接近于SGA和CGA。

表5 算法SGA、CGA、GACM的遗传操作

算法	选择算子	交叉算子	变异算子
SGA	轮盘赌	平坦交叉	随机变异
CGA	轮盘赌	异类个体间的平坦交叉	随机变异
GACM	轮盘赌	基于CM平坦交叉	随机变异

为了进一步说明CM确实改进了遗传算法的搜索性能,在高维单峰函数 f_1 及高维多峰函数 f_8 上进行10次实验,分别运行600代、900代,求得每代平均最优值来验证改进后算法的搜索效率,如图6、图7所示。可见,在实数编码下,算法GAAM的收敛速度依然明显好于CGA,说明CM确实改进了算法的搜索性能。

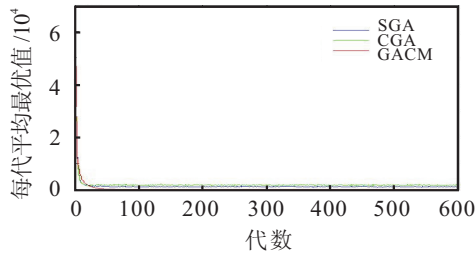


图 6 f_1 的实验结果 (3)

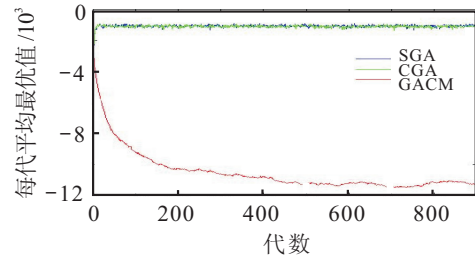


图 7 f_8 的实验结果 (3)

表 6 实数编码下 SGA、CGA、GACM 的实验结果

function/optimal	gen	SGA		CGA		GACM	
		avg	stdev	avg	stdev	avg	stdev
$f_1/0$	1500	4.33e+02	5.47e+01	6.44e+02	7.64e+01	11.1e+02	2.48e+00
$f_2/0$	2000	7.48e+00	3.80e-01	9.57e+00	5.97e-01	5.75e-01	7.25e-02
$f_3/0$	5000	1.04e+03	1.48e+02	1.54e+03	1.85e+02	3.06e+02	5.10e+01
$f_4/0$	5000	5.76e+00	3.14e-01	8.86e+00	4.71e-01	1.22e+00	1.35e-01
$f_5/0$	20000	1.59e+04	2.63e+03	2.82e+04	5.46e+03	3.62e+02	5.60e+01
$f_6/0$	1500	4.52e+02	6.42e+01	6.46e+02	6.94e+01	1.33e+01	2.43e+00
$f_7/0$	3000	3.16e-02	6.59e-03	5.69e-02	1.39e-02	2.37e-02	5.81e-03
$f_8/-12569.5$	9000	-2884.7	4.30e+02	-2699.5	3.41e+02	-12267.3	7.26e+01
$f_9/0$	5000	9.75e+01	8.29e+00	1.16e+02	6.96e+00	6.47e+00	1.54e+00
$f_{10}/0$	1500	4.66e+00	1.99e-01	5.52e+00	2.43e-01	1.10e-01	1.36e-02
$f_{11}/0$	2000	4.97e+00	4.14e-01	6.60e+00	7.73e-01	1.10e+00	1.88e-02
$f_{12}/0$	1500	5.44e+00	7.75e-01	7.80e+00	8.93e-01	4.45e-01	1.15e-01
$f_{13}/0$	1500	2.57e+01	4.20e+00	1.06e+02	1.07e+02	2.38e+00	4.48e-01
$f_{14}/1$	100	1.04	1.96e-01	1.43	4.68e-01	1	5.42e-04
$f_{15}/3.075e-04$	4000	6.557e-04	1.37e-04	7.600e-04	1.47e-04	4.999e-04	6.01e-05
$f_{16}/-1.0316285$	100	-1.027447	4.56e-03	-1.028420	3.33e-03	-1.031364	3.00e-04
$f_{17}/0.398$	100	0.407	5.67e-03	0.405	6.31e-03	0.398	7.63e-04
$f_{18}/3$	100	3.09	9.28e-02	3.13	1.27e-01	3.01	1.65e-02
$f_{19}/-3.86$	100	-3.89	1.31e-03	-3.86	2.30e-03	-3.86	2.54e-05
$f_{20}/-3.32$	200	-3.21	2.54e-02	-3.28	1.67e-02	-3.29	4.92e-02
$f_{21}/-10$	100	-6.61	3.70e+00	-6.75	1.27e+00	-8.24	2.86e+00
$f_{22}/-10$	100	-9.98	1.67e+00	-7.80	1.16e+00	-8.53	2.91e+00
$f_{23}/-10$	100	-10.13	1.59e+00	-7.10	1.29e+00	-9.68	2.12e+00

为了进一步证明算法的性能, 在实数编码下又分别在高维单峰函数 f_1 及高维多峰函数 f_8 上进行了以时间为横坐标的性能衡量实验. 实验运行 10 次, 进化停止的条件是 60 s, 每次实验都记录了每间隔 1 s 种

群所进化的代数 gen , 以及到此刻此代数为止搜索到的当前最优解 $bestsofar$; 然后取 10 次结果的代数和当前最优解的平均值. 所得到的实验结果如图 8、图 9 所示. 从图 8、图 9 中可以明显看出, 在相同时间内,

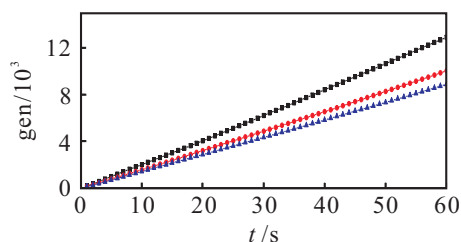
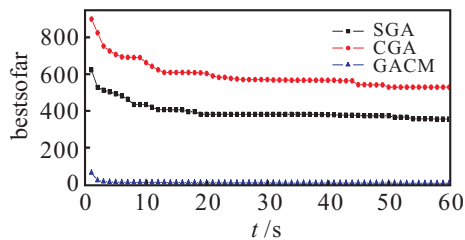


图 8 f_1 的实验结果 (4)

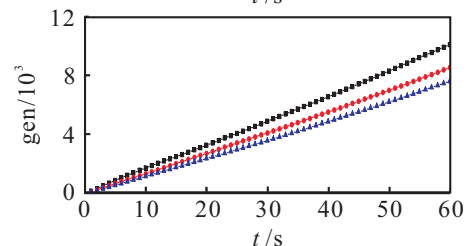
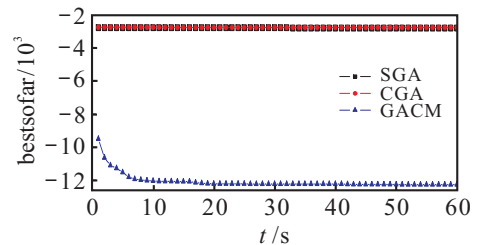


图 9 f_8 的实验结果 (4)

GACM 运行的代数比 SGA、CGA 要少, 这说明了聚类操作确实增加了每次迭代的计算量; 但还可以发现, GACM 在相同的时间内以较少的运行代数搜索到的最优解则是 3 种方法中最佳的, 从而更有力地证明了 GACM 具有较好的收敛速度和寻优能力。

4 结 论

本文针对遗传算法中常见的早熟收敛现象, 提出了一个基于聚类的交叉模型 CM, 进而提出了基于交叉模型的改进遗传算法 GACM, 并在 23 个函数上进行了性能测试, 且与标准遗传算法 SGA 和文献 [13] 中算法 CGA 做了比较. 从实验结果可以看出, 无论在何种编码体制下 CM 都有效改善了 CGA 算法的搜索性能, 从而验证了 CM 的有效性; 同时, GACM 的实验结果好于 SGA 和 CGA, 从而表明本文算法改善了遗传算法的搜索性能。

参考文献(References)

- [1] Park T, Ryu K R. A dual population genetic algorithm with evolving diversity[C]. IEEE Congress on Evolutionary Computation. Singapore: IEEE, 2007: 3516-3522.
- [2] Back T. The interaction of mutation rate, selection, and self-adaptation within a genetic algorithm[C]. Parallel Problem Solving from Nature. Brussels, 1992: 87-96.
- [3] 黄江波, 付志红. 基于自适应遗传算法函数优化与仿真[J]. 计算机仿真, 2011, 28(5): 237-240.
(Huang J B, Fu Z H. Simulation and adaptive genetic algorithm used in function optimization[J]. Computer Simulation, 2011, 28(5): 237-240.)
- [4] 杨华芬. 一种改进的遗传算法在函数优化中的应用[J]. 重庆工商大学学报: 自然科学版, 2009, 26(2): 148-151.
(Yang H F. Application of a modified genetic algorithm in function optimization[J]. J of Chongqing Technology and Business University: Natural Science, 2009, 26(2): 148-151.)
- [5] 吴志远, 邵惠鹤, 吴新余. 一种新的自适应遗传算法及其在多峰值函数优化中的应用[J]. 控制理论与应用, 1999, 16(1): 127-129.
(Wu Z Y, Shao H H, Wu X Y. A new adaptive genetic algorithm & its application in multimodal function optimization[J]. Control Theory & Applications, 1999, 16(1): 127-129.)
- [6] 蒋冬初, 林亚平. 遗传算法在求解函数优化中的最优参数研究[J]. 计算机工程与科学, 2005, 27(10): 50-52.
(Jiang D C, Lin Y P. Research on the best parameters of genetic algorithms to solve functional optimization problems[J]. Computer Engineering & Science, 2005, 27(10): 50-52.)
- [7] 陈长征, 王楠. 遗传算法中交叉和变异概率选择的自适应方法及作用机理[J]. 控制理论与应用, 2002, 19(1): 41-43.
(Chen C Z, Wang N. Adaptive selection of crossover and mutation probability of genetic algorithm and its mechanism[J]. Control Theory & Applications, 2002, 19(1): 41-43.)
- [8] Cantu-Paz E. A survey of parallel genetic algorithms[J]. Calculateurs Paralleles Reseaux Et Systems Repartis, 1997, 10(4): 429-449.
- [9] Whitley D, Rana S, Heckendorn R B. The island model genetic algorithm: On separability, population size and convergence[J]. J of Computing & Information Technology, 1998, 7(1): 33-47.
- [10] Park T, Ryu K R. A dual-population genetic algorithm for adaptive diversity control[J]. IEEE Trans on Evolutionary Computation, 2010, 14(6): 865-884.
- [11] Eshelman L J, Schaffer J D. Preventing premature convergence in genetic algorithms by preventing incest[C]. Proc of Int Conf on Genetic Algorithms. San Diego: Morgan Kaufmann, 1991: 115-122.
- [12] Craighurst R, Martin W N. Enhancing GA performance through crossover prohibitions based on ancestry[C]. Proc of the 6th Int Conf on Genetic Algorithms. Pittsburgh: Morgan Kaufmann, 1995: 130-135.
- [13] 徐晓艳. 基于聚类思想的改进混合遗传算法[D]. 北京: 北京工业大学计算机学院, 2013.
(Xu X Y. The improved hybrid genetic algorithm based on clustering[D]. Beijing: School of Computer Science, Beijing University of Technology, 2013.)
- [14] 周明, 孙树栋. 遗传算法原理及应用[M]. 北京: 国防大学出版社, 1999: 32-44.
(Zhou M, Sun S D. Genetic algorithms theory and applications[M]. Beijing: National Defense University Press, 1999: 32-44.)
- [15] 雷德明. 多维实数编码遗传算法[J]. 控制与决策, 2000, 15(2): 239-241.
(Lei D M. Multidimensional real-coded genetic algorithm[J]. Control and Decision, 2000, 15(2): 239-241.)
- [16] 陈国良, 王煦法, 庄镇泉, 等. 遗传算法及其应用[M]. 北京: 人民邮电出版社, 1996: 76-80.
(Chen G L, Wang X F, Zhuang Z Q, et al. Genetic algorithms and its applications[M]. Beijing: Posts & Telecom Press, 1996: 76-80.)
- [17] Yao X, Liu Y, Lin G. Evolutionary programming made faster[J]. IEEE Trans on Evolutionary Computation, 1999, 3(2): 82-102.

(责任编辑: 李君玲)