

基于广义凸下界估计的多模态差分进化算法*

张贵军, 何洋军, 郭海锋, 冯远静, 徐建明

(浙江工业大学 信息工程学院, 浙江 杭州 310023)

通讯作者: 张贵军, E-mail: zgj@zjut.edu.cn

摘要: 针对多模态优化问题, 提出了基于广义凸下界估计模型的改进差分进化算法. 首先, 基于模型变换方法将原优化问题转变为单位单纯形约束条件下的严格递增射线凸优化问题; 其次, 基于广义凸理论, 利用差分进化算法中更新个体的适应度知识, 建立原优化问题广义凸下界估计模型, 设计实现了基于 N -叉树的估计模型快速计算方法; 进而, 综合考虑原问题目标值与其估计值之间的差异, 提出一种基于有偏采样的小生境指标, 并设计区域进化树更新策略来保证算法的局部搜索能力. 数值实验结果表明, 提出的算法能够有效地发现并维持一定数量的满意解模态, 动态地实现全局模态搜索到模态内局部增强的自适应平滑过渡. 对于给出的测试问题, 能够发现所有的全局最优解以及一些较好的局部极值解.

关键词: 多模态优化; 广义凸; 下界估计; 差分进化; 小生境

中图法分类号: TP301 **文献标识码:** A

中文引用格式: 张贵军, 何洋军, 郭海锋, 冯远静, 徐建明. 基于广义凸下界估计的多模态差分进化算法. 软件学报, 2013, 24(6): 1177-1195. <http://www.jos.org.cn/1000-9825/4323.htm>

英文引用格式: Zhang GJ, He YJ, Guo HF, Feng YJ, Xu JM. Differential evolution algorithm for multimodal optimization based on abstract convex underestimation. Ruan Jian Xue Bao/Journal of Software, 2013, 24(6): 1177-1195 (in Chinese). <http://www.jos.org.cn/1000-9825/4323.htm>

Differential Evolution Algorithm for Multimodal Optimization Based on Abstract Convex Underestimation

ZHANG Gui-Jun, HE Yang-Jun, GUO Hai-Feng, FENG Yuan-Jing, XU Jian-Ming

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Corresponding author: ZHANG Gui-Jun, E-mail: zgj@zjut.edu.cn

Abstract: In this paper, a modified differential evolution algorithm, which is based on abstract convex lower approximation, is proposed for multimodal optimization. First, the original bound constrained optimization problem is converted to an increasing convex along rays (ICAR) function over a unit simplex by using the projection transformation method. Second, based on abstract convex theory, the study builds a lower approximation to original optimization problem by using a finite subset of biased sampling points comes from the population replacement scheme in the basic DE algorithm. Some properties of underestimation model are analyzed theoretically, and an N -ary tree data structure have also been designed and implemented to solve them. Furthermore, considering the difference between the original and its underestimated function values, the paper proposes a niche identify indicator based on biased DE sampling procedure, and also design a regional phylogenetic tree replacement strategy to enhance the exploitation capacity in niche. Experimental results confirm that the proposed algorithm can distinguish between the different attraction basins, and safeguard the consequently discovered solutions effectively. For the given benchmark problems, the proposed algorithm can find all the global optimal solutions and some good local minimum solutions.

Key words: multimodal optimization; abstract convex; underestimation; differential evolution; niching

* 基金项目: 国家自然科学基金(61075062, 50908213); 浙江省自然科学基金(LY13F030008); 浙江工业大学重中之重学科开放基金(20120811)

收稿时间: 2011-12-21; 修改时间: 2012-04-18, 2012-06-25; 定稿时间: 2012-09-03

许多优化问题在限定的解空间中往往存在不止一个全局最优解和数目众多的局部极值解.在机械设计、电磁设计、电力系统规划以及蛋白质结构预测等领域,除了要求得到一个全局最优解之外,还需要得到其他全局最优解和一些质量较好的局部极值解^[1-3].例如:在机械和电气设计中,设计者不但需要找到所有振幅最大的共振频率,而且是最好能找到振幅在某一特定阈值以上的所有共振频率,从而消除大幅值的共振对系统带来的消极影响^[1];在蛋白质结构预测领域,由于力场模型的复杂性和不精确性,全局最优解并不一定就是问题的满意解,有时局部极值解才真正与蛋白质实测基态构型吻合^[2].

多模态优化实质上就是设计一种全局优化算法,能够求出问题所有全局最优解和尽可能多的局部极值解.对于这一类多模态优化问题,Quasi-Newton^[4],Nelder-Mead^[5]及 Hooke-Jeeves^[6]等局部搜索方法基本上不可能得到全局最优解;简单的遗传算法(genetic algorithm,简称 GA)^[7]、差分进化算法(differential evolution,简称 DE)^[8]以及粒子群算法(particle swarm optimization,简称 PSO)^[9]等基于群体的元启发式方法虽然在某些情况下能够求解得到某一个全局最优解,但是选择压力^[10]的存在,使其无法得到其他全局最优解和一些质量较好的局部最优解.针对多模态优化问题,国内外学者借鉴生物学中小生境的概念,结合 GA,DE,PSO 等群体进化算法,相继提出了预选择机制策略、排挤策略、确定性排挤策略^[11]、概率排挤策略^[12]、适应度共享策略^[13]、清洗策略^[14]、聚类策略^[15]、交配限制策略^[16]、岛模型策略^[17]以及其他策略^[18-20].这些策略在一定程度上能够得到多个满意解,但是仍然存在诸如共享半径、清洗半径、早熟收敛以及计算复杂性等问题^[21,22].为进一步解决这些问题,近年来提出了一些上述基本算法的改进策略^[23-26],以及基于拓扑种群分割^[27]和双目标种群的多模态优化算法^[28].

一般来讲,多模态优化算法需要重点解决两个问题:(1) 同峰极值点判断问题,即如何判别新发现的极值点是否与已求出的极值点属于同一峰;(2) 不同峰极值点保持问题,即如何将处于不同峰的极值点能够在后续的进化过程中稳定的保存下来.此外,由于在进化过程所形成的种群规模相对于整个群体规模来说是很小的,因此,算法设计中还应考虑在维持群体多样性的基础上,能够提供一定的选择压力来提高生境内部的收敛速度.本文在 DE 群体进化算法基础上,利用群体中更新个体适应度信息构建多模态目标函数的广义凸下界估计模型,并对群体进行拓扑分类,综合考虑上下界偏差值提出一种表示种群拥挤程度的生境指标,从而保证形成协同进化的种群,进一步采用拓扑区域进化树更新策略来保证生境内部的局部搜索能力.

1 广义凸下界估计模型

1.1 研究动机

凸分析理论的重要研究结果之一是任意下半连续的凸函数是它所有仿射弱函数的上包络,即凸性=线性+上包络^[29].多模态优化本质上是一类非凸函数优化问题,基于广义凸理论,我们一般可以将其表示为:广义凸=分段线性+上包络^[30].考虑到 GA,DE,PSO 等群体算法本质上是一种有偏的可行域空间采样算法,因此,我们可以通过构建基于有偏采样点的广义凸下界估计模型来探知群体算法在进化过程中所形成的小生境.以一个简单的一维多模态函数 $f(x)=\sin(2\pi x)$, $x \in [-2, 0.5]$ 来说明,如图 1 所示.

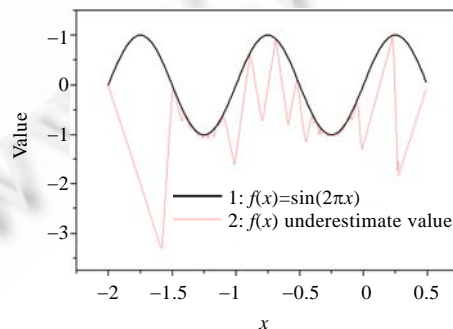


Fig.1 Underestimation curve of multimodal objective function from DE-biased sampling points

图 1 DE 算法有偏采样点形成的低估曲面

图 1 中,曲线 1 表示函数 $f(x)$ 的真实值,曲线 2 表示基本 DE 算法前 60 次有偏采样构建的下界曲线估计值.可以看出,在两个全局最优解附近区域(即吸引区)构建的低估曲线基本上和曲线 1 吻合,而其他部分估计值与实际值的偏差很大.实际上,在某个区域采样点越多,构建的低估曲面也就越精确.当某个采样点的估计值达到预定义的一个阈值 N_e 后,可以认为该模态已经吸纳了足够多的个体,仅通过区域进化树更新策略来保证同一吸引区内个体的替换,从而避免基本 DE 算法由于贪婪替换策略而导致一些较好模态丢失,这也就意味着每一个已经发现的模态均能够稳定地维持一定数量子群体.

1.2 模型建立

考虑如下多模态优化问题:

$$\min f(x), x \in [a, b] \subset R^n \tag{1}$$

其中, $x=[x_1, x_2, \dots, x_n]^T$ 为边界约束可行域 $[a, b]$ 上 n 维连续优化变量, $a=[a_1, a_2, \dots, a_n]^T$ 和 $b=[b_1, b_2, \dots, b_n]^T$ 分别表示矩形可行域的上界和下界向量, $f(\cdot)$ 为定义在可行域 $[a, b]$ 上的多模态目标函数,在限定的解向量空间可能存在多个全局最优解和大量的局部极值解.

对原优化变量 $x=[x_1, x_2, \dots, x_n]^T$ 线性变换:

$$\begin{cases} x'_i \equiv (x_i - a_i) / \sum_{i=1}^n (b_i - a_i), & i = 1, 2, \dots, n \\ x'_{n+1} \equiv 1 - \sum_{i=1}^n x'_i \end{cases} \tag{2}$$

其中, $x'_1, x'_2, \dots, x'_{n+1}$ 为线性转换变量.

显然, $\sum_{i=1}^n (b_i - a_i) > 0, x_i - a_i \geq 0, i = 1, 2, \dots, n$, 可知 $x'_i \geq 0, i = 1, 2, \dots, n, x'_{n+1} \geq 0$, 即

$$x' \equiv [x'_1, x'_2, \dots, x'_{n+1}]^T \in S \equiv \left\{ x' \in R_+^{n+1} : \sum_{i=1}^{n+1} x'_i = 1 \right\} \tag{3}$$

其中, $R_+^{n+1} \equiv \{x' = (x'_1, x'_2, \dots, x'_{n+1}) \mid x'_i \geq 0, i = 1, 2, \dots, n+1\}$, S 表示单位单纯形区域.对公式(2)进行反变换,可得到:

$$x_i = x' \sum_{i=1}^n (b_i - a_i) + a_i, i = 1, 2, \dots, n \tag{4}$$

将公式(4)带入公式(1),可得:

$$\min \bar{f}(x'), x' \in S \subset R_+^{n+1} \tag{5}$$

其中, $\bar{f}(\cdot)$ 为定义在 $n+1$ 维单位单纯形区域 S 上的多模态目标函数.

定理 1. 设 $\exists \bar{L} > 0$, 使得公式(5)目标函数 $\bar{f}: S \rightarrow R$ 满足:

$$\bar{L} = \inf_{x^1 \neq x^2} \frac{|\bar{f}(x^1) - \bar{f}(x^2)|}{\|x^1 - x^2\|_1}, \forall x^1, x^2 \in S \tag{6}$$

取 $\bar{M} > 2\bar{L} - \min_{x' \in S} \bar{f}(x')$, 则 $\forall y \in S$ 处生成的函数 $h^y(x')$:

$$h^y(x') = \min_{i \in \mathcal{O}(l)} l_i x'_i, \forall x' \in S \tag{7}$$

满足 $h^y(x') \leq \bar{f}(x') + \bar{M}, \forall x' \in S$. 其中, $\|x^1 - x^2\|_1 \equiv \max_{i=1, 2, \dots, n+1} |x^1_i - x^2_i|, l = (\bar{f}(y) + \bar{M}) / y, l = (l_1, l_2, \dots, l_{n+1}) \in R_+^{n+1}$, 索引集合 $\mathcal{O}(l) = \{i: l_i > 0\}$.

证明:考虑下列函数 $g: R_+^{n+1} \rightarrow R_+$:

$$g(z) = \begin{cases} \left[\bar{f}\left(z / \sum_{i=1}^{n+1} z_i\right) + \bar{M} \right] \left(\sum_{i=1}^{n+1} z_i \right), & z \neq 0 \\ 0, & z = 0 \end{cases} \tag{8}$$

因 $\min_{x' \in S} \bar{f}(x') + \bar{M} > 2\bar{L} \Leftrightarrow 2\bar{L} / (\min_{x' \in S} \bar{f}(x') + \bar{M}) < 1$, 故由文献[31]中定理 4.1 可知, $g: R_+^{n+1} \rightarrow R_+$ 为严格递增函数. 即 $\forall x^1, x^2 \in R_+^{n+1}$, 如果 $x^1 > x^2$, 则 $g(x^1) > g(x^2)$. 设 $\forall x' \in S \subset R_+^{n+1}$, 由公式(8)可知, $\bar{f}(x') + \bar{M} = g(x')$, 即 $\bar{f} + \bar{M}: S \rightarrow R_+$ 为严格递增函数.

设 $y \in S, \lambda \in (0, +\infty)$, 由于 $y \in S \Rightarrow \sum_{i=1}^{n+1} y_i = 1$, 由公式(8)可得到:

$$g(\lambda y) = \left[\bar{f} \left(\lambda y / \sum_{i=1}^{n+1} (\lambda y_i) \right) + \bar{M} \right] \left(\sum_{i=1}^{n+1} \lambda y_i \right) = \lambda [\bar{f}(y) + \bar{M}] = \lambda g(y) \tag{9}$$

即, $g(y)$ 为 $y \in S$ 处关于 $\lambda \in (0, +\infty)$ 的凸函数. 另, $g(y) = \bar{f}(y) + \bar{M}$, 故 $\bar{f} + \bar{M} : S \rightarrow R_+$ 为 $y \in S$ 处关于 $\lambda \in (0, +\infty)$ 的凸函数, 满足 $\bar{f}(\lambda x') + \bar{M} = \lambda(\bar{f}(x') + \bar{M})$.

由情形①、情形②可知, $\bar{f} + \bar{M} : S \rightarrow R_+$ 为严格递增射线凸函数(strictly increasing convex along rays, 简称 SICAR). 由广义凸理论可以得出 $\bar{f} + \bar{M} : S \rightarrow R_+$ 在 $y \in S$ 点的 H -次微分为

$$h^y(x') = l(x') - l(y) + \bar{f}(y) + \bar{M} = \min_{i \in \Theta(l)} l_i x'_i - \min_{i \in \Theta(l)} l_i y_i + \bar{f}(y) + \bar{M}, \forall x' \in S \tag{10}$$

$$l \in U = \left\{ \frac{v}{y} : v \in \partial \bar{f}_y(1) + \bar{M} \right\} \tag{11}$$

其中, $\bar{f}_y(\lambda) = \bar{f}(\lambda y)$. 另外, 由凸分析理论可知: $\partial \bar{f}_y(\lambda) \in [\bar{f}_y^-(\lambda), \bar{f}_y^+(\lambda)]$.

$$\bar{f}_y^+(\lambda) = \lim_{\beta \downarrow 0} \frac{\bar{f}_y(\lambda + \beta) - \bar{f}_y(\lambda)}{\beta} = \lim_{\beta \downarrow 0} \frac{(\lambda + \beta)\bar{f}(y) - \lambda\bar{f}(y)}{\beta} = \bar{f}(y) \tag{12}$$

$$\bar{f}_y^-(\lambda) = \lim_{\beta \downarrow 0} \frac{\bar{f}_y(\lambda) - \bar{f}_y(\lambda - \beta)}{\beta} = \lim_{\beta \downarrow 0} \frac{\lambda\bar{f}(y) - (\lambda - \beta)\bar{f}(y)}{\beta} = \bar{f}(y) \tag{13}$$

故可得出:

$$\partial \bar{f}_y(\lambda) |_{\lambda=1} = \partial \bar{f}_y(1) = \bar{f}(y) \tag{14}$$

将公式(14)代入公式(11), 可得出支持向量为

$$l = (\bar{f}(y) + \bar{M}) / y = [(\bar{f}(y) + \bar{M}) / y_1, (\bar{f}(y) + \bar{M}) / y_2, \dots, (\bar{f}(y) + \bar{M}) / y_{n+1}]^T \tag{15}$$

由此可得出:

$$\min_{i \in \Theta(l)} l_i y_i = \bar{f}(y) + \bar{M} \tag{16}$$

将公式(16)代入公式(10)可得:

$$h^y(x') = \min_{i \in \Theta(l)} l_i x'_i, \forall x' \in S \tag{17}$$

其中, $l = (\bar{f}(y) + \bar{M}) / y$.

将公式(15)代入公式(17)可知:

$$h^y(x') = (\bar{f}(y) + \bar{M}) \min \left\{ \frac{x'_1}{y_1}, \frac{x'_2}{y_2}, \dots, \frac{x'_{n+1}}{y_{n+1}} \right\} \tag{18}$$

假设

$$\bar{\lambda} = \min \left\{ \frac{x'_1}{y_1}, \frac{x'_2}{y_2}, \dots, \frac{x'_{n+1}}{y_{n+1}} \right\} = \frac{x'_\xi}{y_\xi}, \xi \in \{1, 2, \dots, n+1\} \tag{19}$$

则对于 $\forall t \in \{1, 2, \dots, n+1\}$, 满足:

$$\frac{x'_t}{y_t} \geq \bar{\lambda}, t \in \{1, 2, \dots, n+1\} \Leftrightarrow x'_t \geq \bar{\lambda} y_t, t \in \{1, 2, \dots, n+1\} \Leftrightarrow x' \geq \bar{\lambda} y \tag{20}$$

由于 $\bar{f} + \bar{M} : S \rightarrow R_+$ 为严格递增射线凸函数, 可得出:

$$\bar{f}(x') + \bar{M} \geq \bar{f}(\bar{\lambda} y) + \bar{M} = \bar{\lambda}(\bar{f}(y) + \bar{M}) = h^y(x'), \forall x' \in S \tag{21}$$

特别地, 当 $x'=y$ 时, $\bar{f}(x') + \bar{M} = h^y(x')$. □

注 1: 定理 1 给出了对于给定的多模态优化目标函数公式(5), 通过选取一个合适的常数 \bar{M} , 根据可行域采样点 $y \in S$, 我们可以构建基于条件(6)的下界估计函数 $h^y(x')$, 并满足 $\bar{f}(x') \geq h^y(x') - \bar{M}, \forall x' \in S$. 假定 $\bar{f}(x') = 1, x' \in S^2 \equiv \{x'_1 + x'_2 = 1 | x'_1 \geq 0, x'_2 \geq 0\}$, 由公式(6)可计算得到 $\bar{L} = 0$. 取 $\bar{M} = 0$, 显然 $\bar{M} = 2\bar{L} - \min_{x' \in S} \bar{f}(x')$. 给定 $y = [0.5, 0.5] \in S^2$, 则构建的下界估计函数 $h^y(x')$ 如图 2 所示.

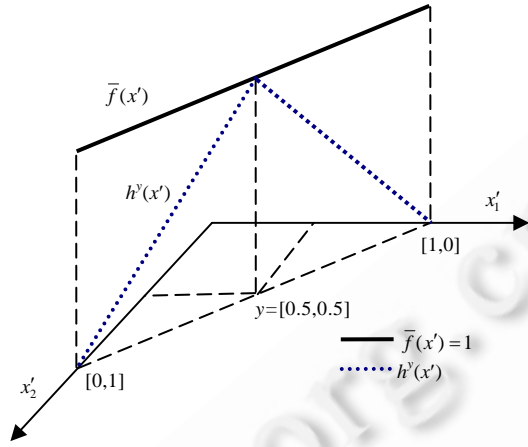


Fig.2 Underestimation function $h^y(x')$ of $\bar{f}(x')=1$ build from the sampling points $y=[0.5,0.5]$

图 2 $\bar{f}(x')=1$ 在 $y=[0.5,0.5]$ 处的下界估计函数 $h^y(x')$

推论 1. 设 $y^1, y^2, \dots, y^K \in S$ 为公式(5)已知采样点,定义以下估计模型:

$$H^K(x') = \max_{k=1, \dots, K} h^k(x') = \max_{k=1, \dots, K} \min_{i \in \Theta(l)} l_i^k x_i' \quad (22)$$

则满足:

$$\bar{f}(x') + \bar{M} \geq H^K(x'), \forall x' \in S \quad (23)$$

$$\bar{f}(x') + \bar{M} = H^K(x'), \forall x' \in \{y^1, y^2, \dots, y^K\} \quad (24)$$

其中, $l_i^k = \frac{\bar{f}(y^k) + \bar{M}}{y_i^k}$, $\Theta(l) = \{i: l_i > 0\}$, 记 $h^k(x') \equiv h^{y^k}(x'), k=1, 2, \dots, K$.

证明:根据定理 1 可知:

$$\bar{f}(x') + \bar{M} \geq h^y(x'), y \in S, \forall x' \in S.$$

故可得到:

$$\bar{f}(x') + \bar{M} \geq \max_{k=1, \dots, K} h^k(x') = H^K(x'), \forall x' \in S \quad (25)$$

设 $x' = y^\xi, \xi \in \{1, 2, \dots, K\}$, 则 $\bar{f}(y^\xi) + \bar{M} = h^\xi(y^\xi)$. 另外, 由公式(25)可知:

$$h^\xi(y^\xi) = H^K(y^\xi) \quad (26)$$

故 $\bar{f}(x') + \bar{M} = H^K(x'), \forall x' \in \{y^1, y^2, \dots, y^K\}$. □

注 2:推论 1 说明了如果已知 K 个采样点,我们可以构建一个满足条件(6)的广义凸下界估计函数 $H^K(x')$. 设 $\bar{f}(x')=1, x' \in S^2 \equiv \{x_1' + x_2' = 1 | x_1' \geq 0, x_2' \geq 0\}$, 取 $\bar{M} = 0$, 显然 $\bar{M} = 2\bar{L} - \min_{x' \in S} \bar{f}(x')$. $y^k \in S^2, k=1, 2, \dots, 7$ 构建的多采样点下界估计函数 $H^7(x')$ 如图 3 所示. 显然, 从图 3 中可以得出 $\bar{f}(x') \geq H^7(x'), \forall x' \in S^2$, 且 $\forall \xi \in \{1, 2, \dots, 7\}$ 均满足公式(26). 推论 1 同时也说明了采样点越多, 我们可以得到更为收紧的下界, 即 $h^7(x') \geq h^k(x'), \forall k=1, 2, \dots, 7$.

注 3:在无偏采样的前提下, 当采样点 $K \rightarrow \infty$ 时, $H^K(x') = f(x') + \bar{M}$. 以一维 Rastrigin 问题^[8]为例说明, 图 4 给出了无偏采样点数量分别为 50, 200, 2 000, 5 000 情况下低估模型的逼近结果. 从图中可以看出, 采样点越多, 逼近精度越高, 当采样点为 5 000 时, 基本上已经逼近原问题. 一维 Rastrigin 问题包含了 10 个局部极值解、1 个全局最优解. 在此, 用 Gaussian 随机数来说明有偏采样结果. 算法首先按均匀分布随机生成 $[-5, +5]$ 之间的整数, 然后以该值作为均值生成标准偏差为 0.1 的 Gaussian 随机数进行有偏采样, 采样结果如图 5 所示. 可以看出, 在某个区域生成的点越多, 那么该区域的逼近程度越精确. DE 群体优化算法实质上是一个有偏采样过程, 利用推论 1, 我们可以基于进化个体构建下界估计模型, 从而探知群体进化所形成的种群.

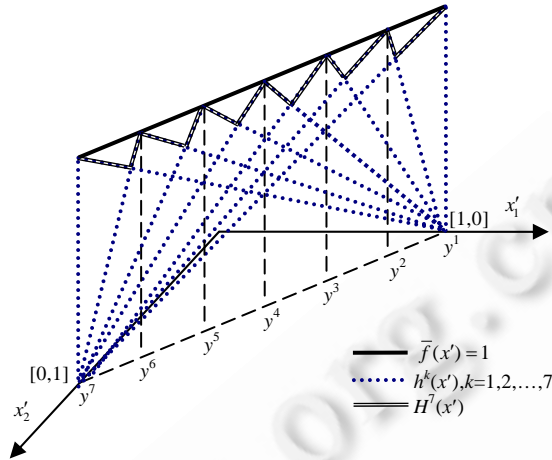


Fig.3 Underestimation function $H^7(x')$ of $\bar{f}(x')=1$ build from the sampling points $y^k, k=1,2,\dots,7$

图3 $\bar{f}(x')=1$ 在 $y^k, k=1,2,\dots,7$ 处构建的下界估计函数 $H^7(x')$

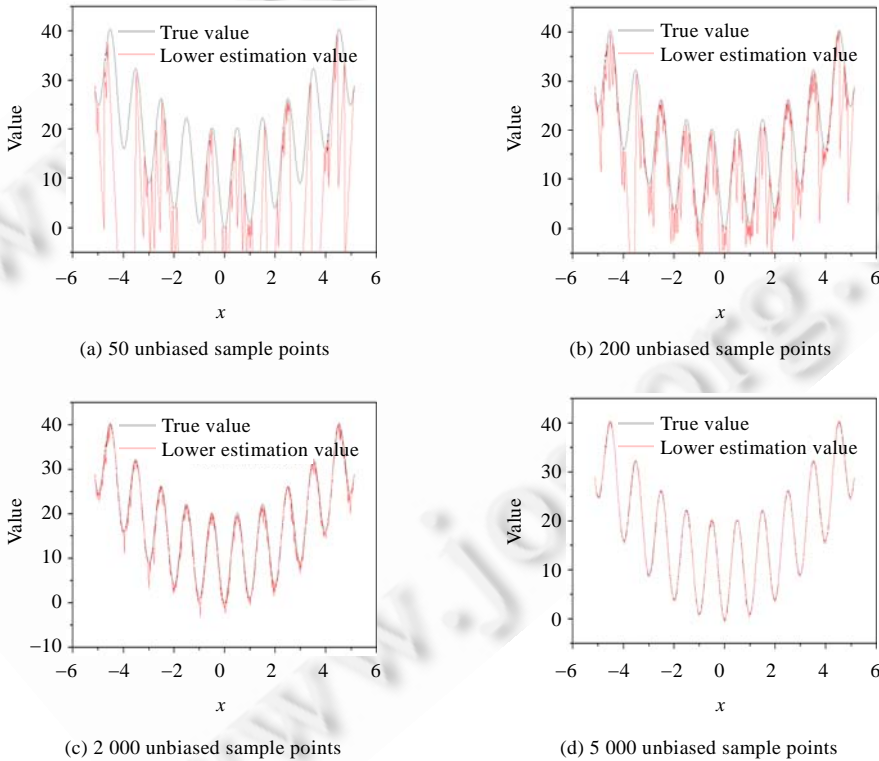


Fig.4 Underestimation curve of 1-D Rastrigin problem from unbiased sampling points

图4 无偏采样点生成的一维 Rastrigin 问题的低估曲面

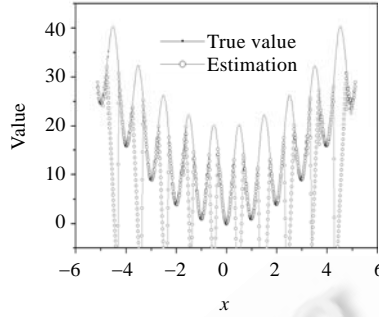


Fig.5 Underestimated curve of 1-D Rastrigin problem from 5 000 Gaussian biased sampling points

图5 5 000 个高斯有偏采样点生成的一维 Rastrigin 问题的低估模型曲面

1.3 模型求解

在 DE 群体进化过程,对于每一个新生成的个体,都需要计算下界估计模型 $H^K(x')$.因此,高效的 $H^K(x')$ 求解算法是至关重要的,在此,本文采用 Bagirov^[32]提出的割角法进行求解.

考虑 $n+1$ 维支持向量矩阵:

$$L = \begin{pmatrix} l_1^{k_1} & l_2^{k_1} & \dots & l_{n+1}^{k_1} \\ l_1^{k_2} & l_2^{k_2} & \dots & l_{n+1}^{k_2} \\ \vdots & \vdots & \vdots & \vdots \\ l_1^{k_{n+1}} & l_2^{k_{n+1}} & \dots & l_{n+1}^{k_{n+1}} \end{pmatrix} \tag{27}$$

Bagirov 证明了 $H^K(x')$ 所有的局优解对应于满足下列两个条件的矩阵集合:

- (I) $\forall i, j \in \{k_1, k_2, \dots, k_{n+1}\}, i \neq j: l_i^{k_i} < l_i^{k_j}$;
- (II) $\forall v \in \{l^1, l^2, \dots, l^K\} \setminus \{l^{k_1}, l^{k_2}, \dots, l^{k_{n+1}}\}$, 则 $\exists i \in \{k_1, k_2, \dots, k_{n+1}\}$, 使得 $l_i^{k_i} \geq v_i$.

对于满足条件(I)、条件(II)的支撑向量矩阵 $L^u (u=1, 2, \dots, U)$, 其局优值 $d(L^u)$ 和局优解 $x'_{\min}(L^u)$ 分别为

$$d(L^u) = 1 / \sum_{i=1}^{n+1} (1/l_i^{k_i^u}) \tag{28}$$

$$x'_{\min}(L^u) = d(L^u) / \text{Trace}(L^u) \tag{29}$$

其中, $\{k_1^u, k_2^u, \dots, k_{n+1}^u\}$ 为满足条件(I)、条件(II)的一种排列.

Beliakov^[33]随后证明了局优解 $x'_{\min}(L^u)$ 同时也唯一地确定了一个子区域 S^{L^u} .

$$S^{L^u} = \{x'_1, x'_2, \dots, x'_{n+1} \mid x'_j > x'_i, i, j \in \{1, 2, \dots, n+1, i \neq j\}, u=1, 2, \dots, U\} \tag{30}$$

且满足

$$S = \bigcup_{u=1}^U S^{L^u}, S^{L^i} \cap S^{L^j} = \emptyset, i \neq j, i, j \in \{1, 2, \dots, U\} \tag{31}$$

即,将单位单纯形可行域 S 划分为了 U 个互不交叉的子区域 S^{L^u} . 特别地,当 $\forall x' \in S^{L^u}$ 时,有

$$H^K(x') = \max\{l_1^{k_1^u} x'_1, l_2^{k_2^u} x'_2, \dots, l_{n+1}^{k_{n+1}^u} x'_{n+1}\}, \forall x' \in S^{L^u}, u \in \{1, 2, \dots, U\} \tag{32}$$

枚举所有满足条件(I)、条件(II)的支撑向量矩阵 $L^u, u \in \{1, 2, \dots, U\}$, 其计算复杂度为 $O(n(C_{K-1}^n))$; Batten^[34]进一步证明了 $L^u, u \in \{1, 2, \dots, U\}$ 位于一个有向无环图上,通过检查图的叶结点 V^{K-1} 发现满足条件的所有组合,从而可以将计算复杂度降至 $O(n|V^{K-1}|)$, $|V^{K-1}|$ 表示 $H^{K-1}(x')$ 局优解数量,而且还证明了包含新的支撑向量 l^K 并满足条件的矩阵可从 V^{K-1} 中刚删除的叶结点得到(当 $v=l^K$ 时可能导致 V^{K-1} 中某些叶结点不满足条件(II)); Beliakov^[35]将历次结点 V^{n+1}, \dots, V^{K-1} 构造一棵度数最多为 $n+1$ 的树 $T^K = \bigcup_{k=n+1, \dots, K} V^k$, 这样可将计算复杂度降至 $\log_{n+1}(|V^K|)$.

Batten, Beliakov 等学者主要解决了如何能够有效地枚举得到所有满足条件(I)、条件(II)的支撑向量矩阵 L^u ,

而本文研究的 $H^K(x')$, $x' \in S$ 求解方法则是如何根据给定 $x' \in S$ 查询某个子区域 S^{L_u} , $u=1, 2, \dots, U$, 使得 $x' \in S^{L_u}$, 二者是不同的. 一旦我们找到 x' 所位于的区域 S^{L_u} (唯一地对应于一个满足条件(I)、条件(II)的矩阵 L^u), 利用公式(29), 公式(32)就可以非常廉价地计算出 $H^K(x')$ 以及距离 x' 最近的估计模型局部极值解 $x'_{\min}(L^u)$. 同时, 利用公式(28)还可以快速地计算得到该局部极值 $d(L^u)$. 为此, 我们首先给出以下定理.

定理 2. 设 $x^{s'} \in S^{L_u}$, 记 $l^s = ((\bar{f}(x^{s'}) + \bar{M})/x_1^{s'}, (\bar{f}(x^{s'}) + \bar{M})/x_2^{s'}, \dots, (\bar{f}(x^{s'}) + \bar{M})/x_{n+1}^{s'})$, 则 $l_i^s > l_i^{k_u}$, $\forall i \in \Theta(l)$.

证明: 设当前采样点为 x^1, x^2, \dots, x^K , 支撑向量为 l^1, l^2, \dots, l^K , 且形成 S^{L_u} 的支撑向量排列为 $l^{k_1^u}, l^{k_2^u}, \dots, l^{k_{n+1}^u}$, 对应的采样点依次为 $x^{k_1^u}, x^{k_2^u}, \dots, x^{k_{n+1}^u}$. 因 $x^{s'} \in S^{L_u}$, 由公式(30)可知 $x^{s'} \neq x^{k_1^u}, x^{k_2^u}, \dots, x^{k_{n+1}^u}$. 另外, 根据推论 1 可知:

$$\bar{f}(x^{s'}) + \bar{M} > H^K(x^{s'}) = \max_{k=1, \dots, K} \min_{i \in \Theta(l)} l_i^k x_i^{s'} = \max\{l_1^{k_1^u} x_1^{s'}, l_2^{k_2^u} x_2^{s'}, \dots, l_{n+1}^{k_{n+1}^u} x_{n+1}^{s'}\} \quad (33)$$

设 $l^{K+1} = l^s = ((\bar{f}(x^{s'}) + \bar{M})/x_1^{s'}, (\bar{f}(x^{s'}) + \bar{M})/x_2^{s'}, \dots, (\bar{f}(x^{s'}) + \bar{M})/x_{n+1}^{s'})$ 为由 $x^{s'}$ 生成的第 $K+1$ 个支撑向量, 则

$$\bar{f}(x^{s'}) + \bar{M} = \max_{k=1, \dots, K} \min_{i \in \Theta(l)} l_i^k x_i^{s'} \quad (34)$$

由公式(33)、公式(34)可得出:

$$\max_{k=1, \dots, K} \min_{i \in \Theta(l)} l_i^k x_i^{s'} > \max\{l_1^{k_1^u} x_1^{s'}, l_2^{k_2^u} x_2^{s'}, \dots, l_{n+1}^{k_{n+1}^u} x_{n+1}^{s'}\} \quad (35)$$

又因为

$$\max_{k=1, \dots, K} \min_{i \in \Theta(l)} l_i^k x_i^{s'} = \max\{\max_{k=1, \dots, K} \min_{i \in \Theta(l)} l_i^k x_i^{s'}, \min_{i \in \Theta(l)} l_i^{K+1} x_i^{s'}\} \quad (36)$$

可得到:

$$\min_{i \in \Theta(l)} l_i^{K+1} x_i^{s'} > \max\{l_1^{k_1^u} x_1^{s'}, l_2^{k_2^u} x_2^{s'}, \dots, l_{n+1}^{k_{n+1}^u} x_{n+1}^{s'}\} \quad (37)$$

即 $\forall i \in \Theta(l), l_i^s = l_i^{K+1} > l_i^{k_u}$. □

注 4: 定理 2 说明了如果 $x^{s'} \in S^{L_u}$, 那么以 $x^{s'}$ 生成的支撑向量 l^s 会使 L^u 不满足条件(II). 也就是说, 当我们需要判断某个点 $x^{s'}$ 是否位于当前某个区域 S^{L_u} , $u=1, 2, \dots, U$ 时, 首先判断 l^s 的存在是否会导致 L^u 不合法. 如果我们采用一种非线性信息结构——树来描述历次结点 $V^{m+1}, V^{m+2}, \dots, V^K$ 的层次结构, 根据文献[35]的研究结果, 如果新加入的支撑向量 l^s 使 L^u 不满足条件(II), 则 L^u 所有的父结点均不满足条件(II); 如果某个层次的父结点满足条件(II), 则该结点下包含的子树均满足条件(II), 可以不用测试. 这样, 子区域 S^{L_u} , $u=1, 2, \dots, U$ 查询的平均复杂度就可以从 $O(U)$ 降至 $O(\log_{n+1} U)$.

2 算法设计

2.1 基本思想

一般情况下, 基本 DE 算法在进化初期能够很好地发现多模态优化问题的绝大多数较好模态(即小生境), 但问题是由于其采用贪婪替换策略, 较大的选择压力使得某些较好模态不能在后续的进化过程中稳定地保留下来. 从进化计算的角度来讲, 一方面我们希望能够找到一批较好模态, 这样同时也能提高发现全局最优解的概率; 另一方面, 我们也希望能够提高算法的局部收敛速度. 但这两个方面是一对矛盾体, 综合权衡两方面的因素, 是设计合理多模态优化算法的关键. 本文根据前面给出的定理及推论, 利用基本 DE 算法迭代过程形成的有偏采样点, 对原问题进行下界逼近. 通过计算上下界之间的差异程度来判断采样点的聚集程度, 进而判断其是否位于一些已经形成的较好模态. 一旦确定某个新采样点位于某个好的模态, 就可以采用区域进化树更新策略来增强模态内的局部搜索能力; 否则, 仍然采用贪婪替换策略来实现模态的大范围搜索. 这样, 算法就可以动态地实现从基于有偏采样点的大范围模态搜索到模态内局部增强的自适应平滑过渡.

2.2 小生境性能指标

提出的小生境性能指标主要针对多模态优化中同峰极值点判断和不同峰极值点保持两个问题.

定义 1. 设 $x \in [a, b] \in \mathbb{R}^n$, 其中, $[a, b]$ 为多模态优化问题 $f(x)$ 的 n 维可行域, 则在 x 处的小生境指标 $N_{\text{niche}}(x)$ 为

$$N_{\text{niche}}(x) = \frac{f(x) + \bar{M} - d(L^u)}{f(x) + \bar{M}} \quad (38)$$

其中, L^u 为转换变量.

$x' = [(x_1 - a_1) / \sum_{i=1}^n (b_i - a_i), (x_2 - a_2) / \sum_{i=1}^n (b_i - a_i), \dots, (x_n - a_n) / \sum_{i=1}^n (b_i - a_i), 1 - \sum_{i=1}^n (x_i - a_i) / \sum_{i=1}^n (b_i - a_i)]^T$ 位于子区域 S^{L^u} 所对应的支撑向量矩阵, $d(L^u)$ 为子区域 S^{L^u} 的唯一局部极值, 常数 $\bar{M} > 0$.

注 5: 定义 1 在几何意义上表示给定变量 x 的函数值 $f(x) + \bar{M}$ 与其对应的低估子区域 S^{L^u} 局部极值 $d(L^u)$ 之间的差异程度. 常数 \bar{M} 对多模态算法有一定影响, 理论上讲应该满足定理 1 的条件, 但是在实际应用中只要选取一个足够大的数即可. 另外, $N_{\text{niche}}(x)$ 的计算是简单高效的. 首先对 x 按照公式(2)进行线性变换得到 x' , 然后按照定理 2 以对数复杂度查询得到子区域 S^{L^u} , 进而直接利用公式(28)显式计算得到 $d(L^u)$.

给定小生境指标阈值 $N_e > 0$, 当 $N_{\text{niche}}(x) < N_e$ 时, 我们认为, 在 x 点附近吸纳了足够多的个体, 已经形成一定规模的种群. 当基本 DE 算法^[8]执行选择、交叉、变异操作之后, 如新生成的测试个体 x_{Trial} 优于对应的目标个体 x_{Target} 且 $N_{\text{niche}}(x) \geq N_e$ 时, 则执行常规的更新操作 $x_{\text{Target}} = x_{\text{Trial}}$, 同时将 x_{Trial} 作为下一个采样点构建支撑向量进而更新系统维护的 N 叉树结构, 以获得更为精确的低估模型; 否则丢弃. 以 Periodic 问题^[36]为例说明 $N_{\text{niche}}(x)$ 能够有效地解决多模态优化中同峰极值点判断和不同峰极值点保持两个问题. Periodic 问题在原点(0,0)处包含 1 个全局最优解以及 48 个局部极值解. 图 6 给出了群体规模 $\text{popSize} = 1000$, $\bar{M} = 1000$, $N_e = 0.01, 0.5, 10, 1000$ 时, 执行该更新策略运行到 100 代时得到的结果.

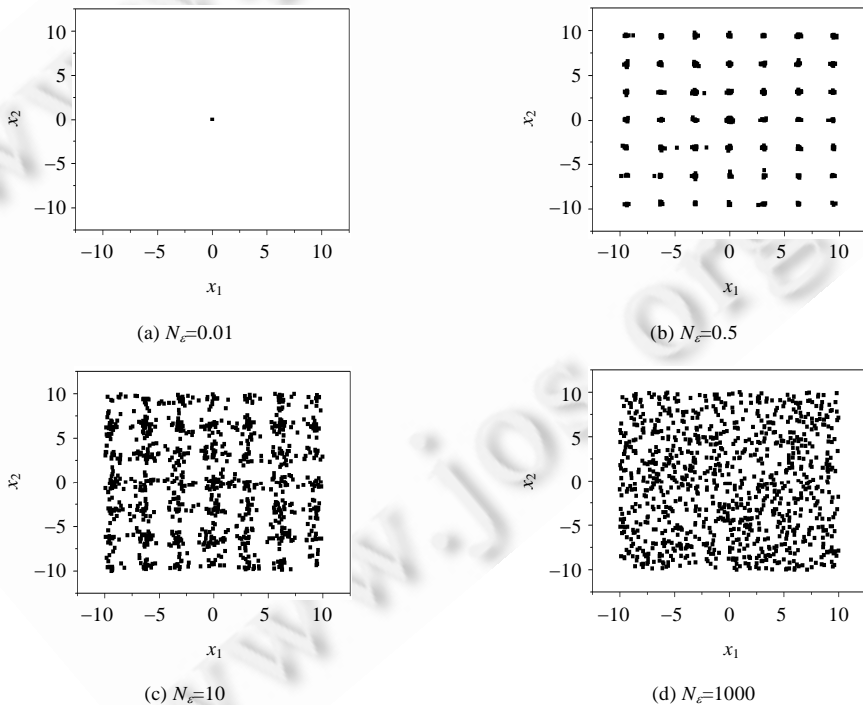


Fig.6 Population distribution in 100th generation using different niche index

图 6 在不同小生境指标阈值下第 100 代群体分布情况

可以看出, 当 $N_e = 0.01$ 时, 1 000 个体在第 69 代就已经收敛于唯一的全局最优解(0,0). 实际上, 当 $N_e = 0$, 算法与基本 DE 算法相同, 其选择压力驱使所有的个体收敛于唯一的一点; 当 $N_e = 0.5, 10$ 时可以看出, 均能保持 49 个种群, 并且 N_e 越大, 群体的多样性越好, 但同时搜索能力也逐渐减弱; 特别地, 当 $N_e = 1000$ 时, 基本上丧失进化能力. 这

样,通过选择适当的控制参数 N_e ,就可以在在一定程度上控制算法的多模态特性.对于本文研究的问题,实验表明, $N_e \approx 0.5 \sim 1$ 是一个较好的选值范围.图 7 给出了当 $N_e=0.5$,算法运行至 100 代时 1 000 个个体(已按目标值从小到大进行排序)的低估值和真实值之间的差异情况(群体中个体按函数值升序排列).Periodic 问题除了一个 $f(0,0)=0.9$ 的全局最小值之外,还包括了 48 个局部极值为 1 的局优解.可以看出,1 000 个体绝大多数已经收敛至极值(0.9 和 1.0)附近,有 30% 个体位于全局最优解所在的模态,70% 的个体位于其他 48 个局优解所位于的局部模态(如图 6(b)所示).同时,每个模态均生成了足够精度的广义凸下界估计曲面.

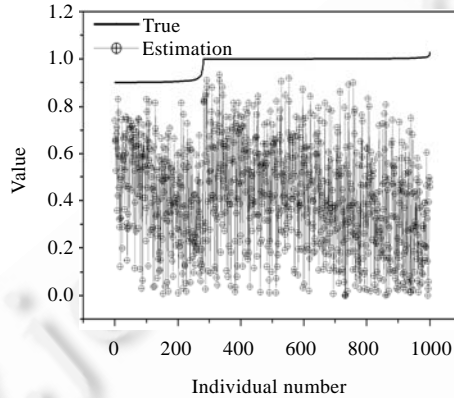


Fig.7 True and estimated value of 1 000 individuals' numbers in 100th generation
图 7 进化至 100 代时,1 000 个体的函数值和估计值

2.3 区域进化树更新策略

区域进化树更新策略目的是为了增强小生境内部的局部搜索能力.以 $n=2$ 的多模态优化问题为例说明.假设基于第 2.2 节描述的小生境指标 DE 算法依次更新了 5 个有偏采样点 x^1, x^2, x^3, x^4, x^5 ,按公式(2)进行线性转换得到在单位单纯形可行域 S 的 5 个采样点(维数增加 1,为 $n+1$) x^1, x^2, x^3, x^4, x^5 用来构建广义凸下界估计曲面,构建过程如图 8 所示.由第 1.3 节理论分析过程可知,结点 V^i 、支撑向量矩阵 L^i 以及生成子区域 S^i 存在一一对应关系,故 V^i 也就可看成 S^i ,且满足以下关系(由于 $n=2$,故树中每个结点最多有 3 个孩子结点):

- $V^4 = V^{51} \cup V^{52} \cup V^{53}$, (x^1 采样点分割形成);
- $V^{51} = V^{61} \cup V^{62}$, (x^2 采样点分割形成);
- $V^{53} = V^{71} \cup V^{72}$, (x^3 采样点分割形成);
- $V^{62} = V^{81} \cup V^{82}$, (x^4 采样点分割形成);
- $V^{71} = V^{91} \cup V^{92} \cup V^{93}$, (x^5 采样点分割形成).

从而可归纳得到通过 5 个采样点,整个可行域 S 分割成 8 个子区域(对应于所有叶子结点),且满足:

- $S = V^4 = V^{61} \cup V^{82} \cup V^{82} \cup V^{52} \cup V^{91} \cup V^{92} \cup V^{93} \cup V^{72}$;
- $V^{61} \cap V^{82} \cap V^{82} \cap V^{52} \cap V^{91} \cap V^{92} \cap V^{93} \cap V^{72} = \emptyset$.

借鉴生物学中进化树概念,在算法实现中保存每一个结点,根据子区域准包含关系可以得到一棵区域进化树.在区域进化树上每个结点代表一个子区域,父结点和子结点表示区域细分(或称进化)过程和一定的包含关系.区域进化树更新策略步骤表示如下(C++伪代码描述):

```

regionUpdating(tree,target,trial,depth) //区域进化树更新算法
输入:支撑向量矩阵树 tree、目标个体 target、测试个体 trial、区域更新深度 depth;
输出:更新状态 status.
Step 1. subNode=FindSubRegionNode(tree,target); //发现 target 位于的子区域结点 subNode.
Step 2. containedStatus=IsContainedFromNode(subNode,trial); //判断 subNode 是否包含 trial.
    
```

```

Step 3. if (containedStatus==TRUE) //如果 subNode 对应子区域包含测试个体.
Step 3a. target=trial; //执行更新操作.
Step 3b. status=TRUE; //设置更新状态为 TRUE.
Step 3c. goto Step 7; //转至 Step 7,退出.
Step 4. depth=depth-1; //区域更新深度减 1.
Step 5. if (depth>0) //如果 updateDepth 大于 0.
Step 5a. subNode=FindParentNode(subNode); //查找 subNode 父结点.
Step 5b. if(subNode==rootNode) goto Step 3a; //如果 subNode 为区域树根结点,转至 Step 3a.
Step 5c. goto step 2; //转至 Step 2.
Step 6. status=FALSE; //设更新状态为 FALSE.
Step 7. EXIT; //退出.
    
```

注 6:根据定理 2 可知,Step 1 中,FindSubRegionNode 函数的计算复杂度为 $O(\log_{n+1}U)$,其中, U 为支撑向量矩阵 $tree$ 当前所有的树叶结点数(即当前子区域数);Step 2 根据公式(30)进行判断.

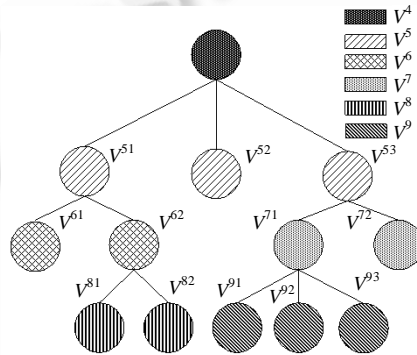


Fig.8 Schematic diagram of regional phylogenetic tree construction process ($n=2$)

图 8 区域进化树构建过程示意图($n=2$)

2.4 算法描述

整体算法流程描述如下(极小值优化):

Step 0. 算法初始化.设置群体规模 $popSize$ 、变异因子 $mutatorFactor$ 、交叉因子 $crossFactor$ 、常数 \bar{M} 、小生境指标阈值 N_ϵ 、区域更新深度 $depth$ 、终止代数 gen 、问题维数 n ;初始群体 $POP=\{x^1, x^2, \dots, x^{popSize}\}$;在 $n+1$ 维单纯形可行域空间 S 生成 $n+1$ 个初始采样点 $x^1=[1,0,\dots,0]^T, x^2=[0,1,0,\dots,0]^T, \dots, x^{n+1}=[0,0,\dots,1]^T$,根据 $n+1$ 初始采样点建立第 1 个支撑向量矩阵 L ,以之作为根结点 $rootNode$ 构造支撑向量矩阵树 $tree$,初始化 $g=0$.

Step 1. 对每一个目标个体 $x^i \in R^n (i=1,2,\dots, popSize)$ 做如下处理:

- 1a. 设置 $i=1$;
- 1b. 任意选取 3 个个体 $\{x^a, x^b, x^c | a, b, c \in \{1,2,\dots, popSize\}, a \neq b \neq c \neq i\}$;
- 1c. 对 $\{x^a, x^b, x^c\}$ 做变异操作,生成变异个体 \hat{x}^i ;
- 1d. 对目标个体 x^i 和变异个体 \hat{x}^i 执行交叉操作,生成测试个体 \bar{x}^i ;
- 1e. $i=i+1$;
- 1f. 如果 $i \leq popSize$,转至步骤 1b;

Step 2. 对目标个体 $x^i \in R^n$ 和对应的测试个体 $\bar{x}^i \in R^n (i=1,2,\dots, popSize)$ 逐个做如下更新操作:

- 2a. 设置 $i=1$;

- 2b. 如果 $f(\bar{x}^i) \geq f(x^i)$, 转至步骤 2i;
- 2c. 按公式(2)将 $x^i, \bar{x}^i \in R^n$ 转换至单纯形空间 $x^i, \bar{x}^i \in S^{n+1}$;
- 2d. 按公式(38)计算测试个体 $\bar{x}^i \in R^n$ 小生境性能指标 $N_{\text{niche}}(\bar{x}^i)$;
- 2e. 如果 $N_{\text{niche}}(\bar{x}^i) \geq N_e$, 转至步骤 2h;
- 2f. 执行 $\text{status}=\text{regionUpdating}(\text{tree}, x^i, \bar{x}^i, \text{depth})$ 过程;
- 2g. 如果 $\text{status}=\text{FALSE}$; 转至步骤 2i;
- 2h. 执行更新操作 $x^i = \bar{x}^i$, 以 \bar{x}^i 为新的采样点构建支撑向量 l^i , 并更新 tree 结构;
- 2i. $i=i+1$;
- 2j. 如果 $i \leq \text{popSize}$, 转至步骤 2b;

Step 3. $g=g+1$;

Step 4. 如果 $g \leq \text{gen}$, 转至步骤 Step 1;

Step 5. 结果输出, 退出.

注 7:

- 1) 步骤 1c、步骤 1d 中, DE 算法变异、交叉操作见文献[8]; 步骤 2h 中, tree 结构更新过程参见文献[35];
- 2) 步骤 2d 和步骤 2f 中均需要执行 FindSubRegionNode 操作, 在具体实现中只要执行一次即可;
- 3) 步骤 2b、步骤 2e 和步骤 2g 说明, 广义凸下界低估曲面的构建过程是有偏的. 有偏采样点构建过程不仅可以探知并保持一些较好模态, 同时还可以减少支撑向量的数目, 进而降低对内存空间的消耗.

3 数值研究

3.1 测试问题

为测试本文所提出算法的性能, 本文针对 4 个典型的多模态优化问题进行了数值研究, 算法实现采用 C++ 语言. 4 个问题及优化结果分别描述如下:

问题 1: Shubert 函数(2 维)

$$\min f(x_1, x_2) = \sum_{i=1}^5 i \cos[(i+1)x_1 + i] \cdot \sum_{i=1}^5 i \cos[(i+1)x_2 + i], x_1, x_2 \in [-10, 10] \quad (39)$$

Shubert 函数^[37]是一个多峰目标函数, 可行域中存在分布极不均匀的 760 个局部极值解, 其中包含最优值为 -186.731 的 18 个全局最小解; 局部极值为 -123.577 的 18 个次优解, 局部极值为 -79.411 的 18 个次次优解. 参数设置: $\text{mutatorFactor}=0.5, \text{crossFactor}=0.1, \text{gen}=1000, \text{popSize}=2000, N_e=0.5, \bar{M}=30000, \text{depth}=D \times 0.2$. 其中, D 支撑向量矩阵树的总高度. 仿真结果显示, 常规 DE 算法基本上在 100 代左右就会收敛至其中某几个全局最优解. 图 9 给出了提出算法运行至 1 000 代时群体的分布情况. 图 9(a) 给出了优化变量 $x_1, x_2 \in [-10, 10]$ 的平面分布情况; 为直观起见, 图 9(b) 给出了在放大区域 $x_1, x_2 \in [-3, 1]$ 的目标值三维图. 从图中可以看出, 截至 1 000 代, 算法能够可靠地得到全部 18 个等值的全局最优解、18 个等值的次最优解以及 18 个等值的次次最优解; 除此以外, 还得到了 17 个较差的局优解. 图 9(b) 显示了放大区域中 8 个全局(或局部)最优解的分布情况. 图 10 给出了运行至 1 000 代时, 2 000 个个体的真实目标值和广义凸下界估计值, 其中, 个体按照目标值升序排列(群体中个体按目标值升序排列). 对照图 9, 可以从图 10 中得出以下结论: 约有 62.5% 个体位于 18 个全局最优解所在模态, 约 25% 的个体位于 18 个次优解所在的模态, 约 12.5% 的个体位于 18 个次次优解所在的模态, 每个模态基本包含了与其目标值成一定比例的个体数目; 另外, 由于下界估计值基本上接近实际的函数目标值, 导致大部分个体的 $N_{\text{niche}}(x) < N_e$, 故这些个体在后续进化过程仍然可以保留下来.

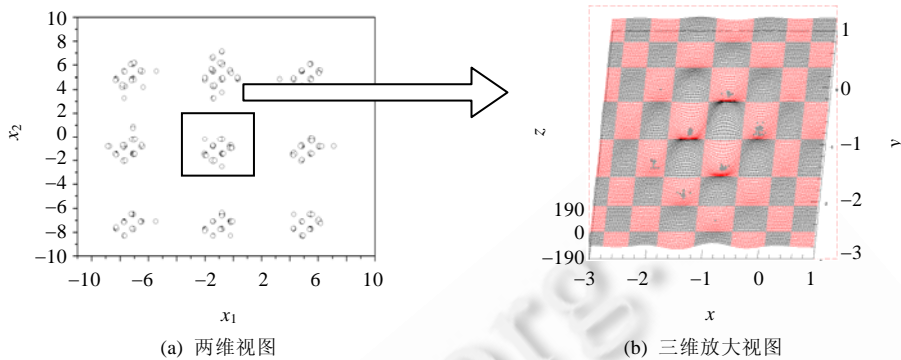


Fig.9 Population distribution in 1000th generation

图9 运行至1000代时群体分布情况

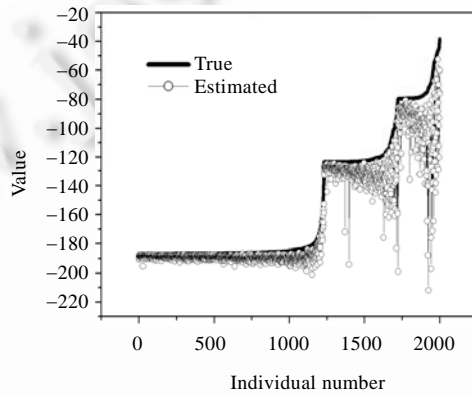


Fig.10 True and estimated values of 2000 individuals' numbers in 1000th generation

图10 1000代时,2000个体的函数值和估计值

理想的多模态优化算法首先是快速的全局探测(exploration)满意解所处的模态;然后,在稳定维持较好模态的前提下,能够实现模态内的局部搜索(exploitation)能力.如何实现全局探测和局部搜索过程的可靠切换,是多模态优化算法设计考虑的重要问题,同时也是目前研究的难点问题.针对基本 DE 算法的群体替换环节,提出算法含两部分替换操作:

- (1) $N_{\text{niche}}(\bar{x}^i) \geq N_e$ 情况下的全局探测置换(exploration replacement),见算法设计步骤 2e;
- (2) $N_{\text{niche}}(\bar{x}^i) < N_e$ 情况下的局部置换(exploitation replacement),见算法设计步骤 2f、步骤 2g.

图 11 分别给出了算法运行至 1000 代时,探测置换和局部置换数量之间的比较结果.

从图中可以看出,在算法迭代 40 次左右,探测置换数量已达到 8000.测试结果表明,这时,基本上形成了一定数量的模态;在随后的优化过程中,模态内的局部搜索能力逐渐增强,探测置换数量逐渐减弱,并在 200 代左右两者置换数量基本持平.这说明了提出算法针对 Shubert 函数能够较为有效地实现多模态优化中同峰极值点判断和不同峰极值点保持问题.

图 12 给出了迭代过程中生成支撑矩阵树的结点数、叶子结点数及高度.

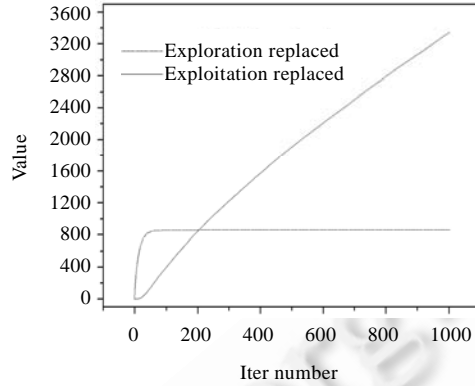
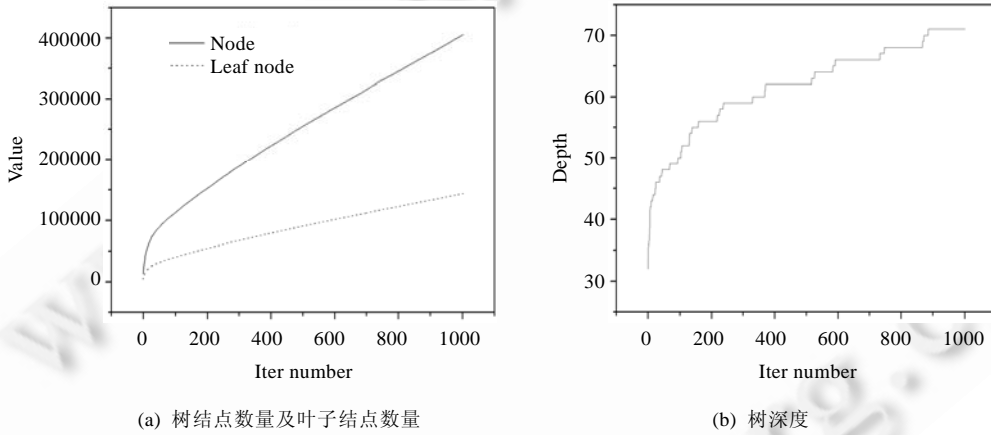


Fig.11 Comparison results of exploitation and exploration replaced numbers in 1 000 generations
图 11 1 000 次迭代过程中,全局探测置换和局部增强置换数量比较结果



(a) 树结点数量及叶子结点数量

(b) 树深度

Fig.12 Size of node, leaf node and height of N -tree
图 12 支撑矩阵 N -叉树的结点、叶子结点及高度

问题 2:Multi-Gaussian 函数(2 维)

$$\min f(x_1, x_2) = -\sum_{i=1}^5 a_i \exp(-((x_1 - b_i)^2 + (x_2 - c_i)^2) / d_i^2), x_1, x_2 \in [-2, 2] \quad (40)$$

其中, $a_i, b_i, c_i, d_i, i=1, 2, 3, 4, 5$, 见表 1.

Table 1 Data for multi-Gaussian problem
表 1 Multi-Gaussian 问题参数

i	a_i	b_i	c_i	d_i
1	0.5	0.0	0.0	0.1
2	1.2	1.0	0.0	0.5
3	1.0	0.0	-0.5	0.5
4	1.0	-0.5	0.0	0.5
5	1.2	0.0	1.0	0.5

Multi-Gaussian 函数^[37]在可行域中包含 1 个全局最优解(-1.29695)、4 个局部最优解以及 1 个鞍点,而且最小值和局部极值之间差距很小.参数设置除 $popSize=1000$ 之外,与前述参数设置相同.图 13 给出了算法运行至 1 000 代时群体的分布情况,可以看出,算法找到了所有的全局最优解、局优解以及鞍点.统计显示:有 12.8% 个体位于全局最优解所在的模态;由于在鞍点区域存在较为平坦的目标函数曲面,在该区域目标值差别极小(如图

13(b))所示,因此在该模态吸引了将近 80%的个体.

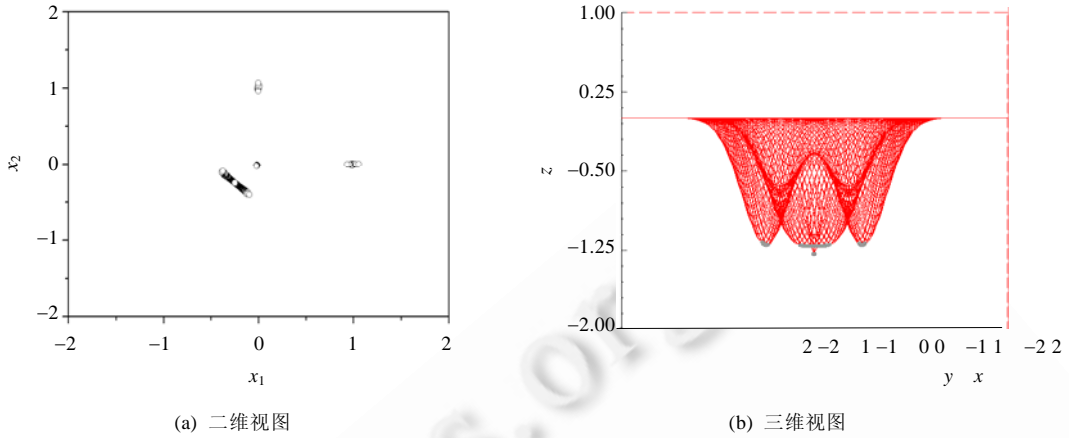


Fig.13 Population distribution in 1000th generation

图 13 运行至 1 000 代时群体分布情况

问题 3:Periodic 函数(2 维)

$$\min f(x_1, x_2) = 1 + \sin^2 x_1 + \sin^2 x_2 - 0.1 \exp(-x_1^2 - x_2^2), x_1, x_2 \in [-10, 10] \quad (41)$$

Periodic 函数^[36]在可行域中包含 1 个全局最优解(目标值为 0.9)、48 个目标值为 1 的局部最优解.49 个全局(或局部)最优解平均分布在可行域中.算法参数设置与 Multi-Gaussian 问题相同.图 14 给出了算法运行至 1 000 代时群体的分布情况,可以看出,算法找到了 1 个全局最优解以及 48 个等值的局部最优解.统计显示,约有 25%个体位于全局最优解所在模态,75%个体位于其他 48 个局优解所在的模态.

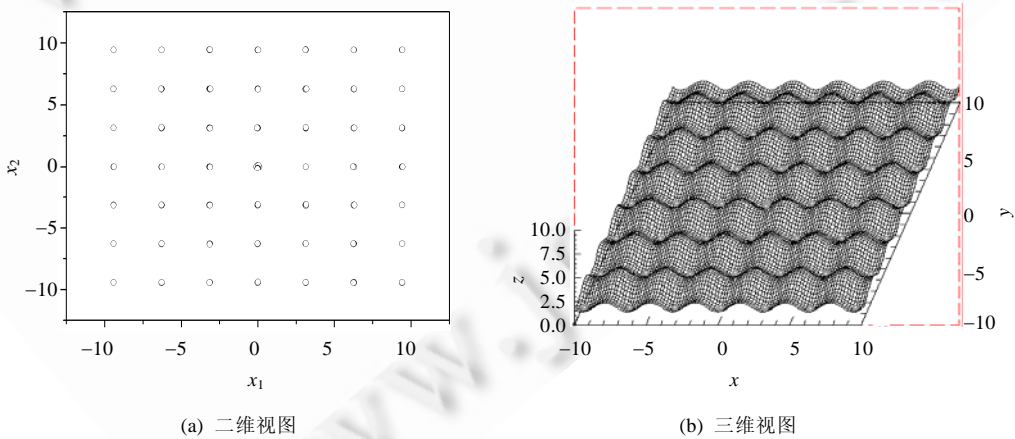


Fig.14 Population distribution in 1000th generation

图 14 运行至 1 000 代时群体分布情况

问题 4:Rastrigin 函数(10 维)

$$\min f(x_1, x_2, \dots, x_n) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)], x_i \in [-5.12, 5.12], i = 1, 2, \dots, n \quad (42)$$

Rastrigin 函数^[8]在可行域中包含 1 个全局最优解(目标值为 0),在其可行域空间中存在数目众多的局部极值解(局部最优解数目未知).在此,我们取 $n=10$,即维数为 10.参数设置与 Multi-Gaussian 问题相同.运行至 1 000 代

时,算法找到了 1 个全局最优解以及若干个局部极值解,如图 15 所示(群体中个体按目标值升序排列).统计分析约有 33.3%的个体收敛至函数值为 0 的全局最优解;约有 31.8%的个体收敛至若干个次全局最优解,其局部极值为 0.994 96;约有 13.5%的个体收敛到函数值为 1.989 92 的若干个次次全局最优解;其余 21.4%的个体分别收敛至局部极值为 2.984 88, 3.979 83, 4.974 79, 5.969 75, 6.964 71, 7.959 66, 8.954 6, 9.949 56, 10.944 5, 11.939 5, 12.934 4, 13.929 4, 15.919 2, 16.914 2, 17.909 2, 18.904 1, 19.899 1, 24.873 7, 25.868 7, 26.863 6, 28.853 6, 31.838 5, 33.828 3, 41.787 9 等局部极值解上.

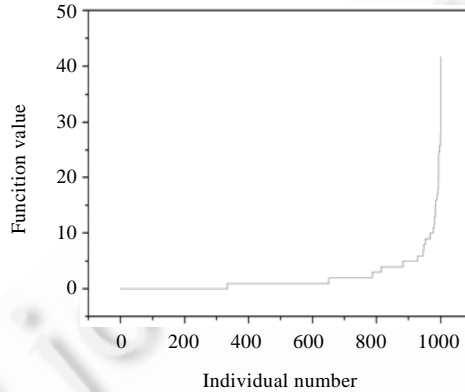


Fig.15 Rastrigin function values of 1 000 individuals in 1000th generation

图 15 运行至 1 000 代时,1 000 个体的函数值

3.2 算法比较

为了验证本文所提出的广义凸下界估计多模态差分进化算法(ACDE)的有效性,另外,采用经典的 SGA^[13]和 CGA^[11]算法对上述 4 个问题进行求解.算法比较结果列于表 2(30 次运行平均结果,终止代数均为 1 000).

Table 2 Comparison results of three multimodal algorithms

表 2 3 种多模态算法比较结果

Problem	Algorithm	Pr (%)	Pa	Problem	Algorithm	Pr (%)	Pa
Shubert (18+18)	ACDE	94.40	30.113	Multi-Gaussian (1+5)	ACDE	100	0.053
	SGA	76.11	18.275		SGA	82.0	0.139
	CGA	69.44	<0.001		CGA	71.67	0.017
Periodic (1+48)	ACDE	91.84	0.560	Rastrigin (1+20)	ACDE	80.95	7.840
	SGA	65.31	1.276		SGA	67.14	4.540
	CGA	66.04	<0.001		CGA	63.81	0.610

在表 2 中,(A+B)表示我们一共需要定位 A+B 个极值点,其中,A 为全局最优解的个数,B 为较为满意局部最优解的个数.本文中,我们需要定位 Shubert 函数的 18 个全局最优解以及 18 个次全局最优解,Multi-Gaussian 函数的 1 个全局最优解和 5 个局部最优解(其中包括 1 个鞍点),Periodic 函数的 1 个全局最优解和 48 个次全局最优解.由于 Rastrigin 函数在其可行域中存在大量数目未知的局部解,由此设定需要定位 1 个全局最优解和 20 个局部最优解.

Pr 表示峰比值(peak ration)指标,即

$$\text{Pr} = \frac{C}{A+B} \times 100\% \quad (43)$$

其中,A 为给定问题全局最优解的个数,B 为给定问题较为满意的局部最优解个数,C 为算法得到的全局和局部最优解数个数.

Pa 表示峰精度(peak accuracy)指标,即

$$Pa = \sum_{i=1}^C |f(c^i) - f(x^i)| \quad (44)$$

其中, C 为算法得到的全局和局部最优解数个数, c^i 为第 i 个已定位的峰位置, x^i 为距离 c^i 最近的解向量, $f(\cdot)$ 表示对应向量的函数目标值。

从表 2 中可以看出:本文提出的 ACDE 算法在峰比值指标方面均优于 SGA 和 CGA 算法,在 30 次运行过程中,均能定位得到 Shubert 问题的 18 个全局最优解以及其他 3 个问题的 1 个全局最优解;同时,算法还能得到 4 个问题的绝大多数满意局优解;另外,由于算法定位得到了更多的满意局优解,由公式(43)可知,在峰精度指标方面稍劣于其他两种算法。

总体上来讲,ACDE 算法无需共享半径先验知识,能够有效发现并维持一定数量的满意解,较好地实现了全局模态搜索到模态内局部增强的自适应平滑过渡.算法采用的区域进化树更新策略在原 DE 算法基础上引入时间复杂度为 $O(\log_{n+1}U)$,其中, U 为支撑向量矩阵当前所有的树叶结点数(即当前子区域数)。

4 结 论

在理论研究基础上,本文提出一种基于广义凸下界估计的多模态差分进化算法,动态实现可行域范围模态探测和模态内的局部增强过程.数值研究结果显示,该算法能够有效地发现并维持一定数量的满意解模态.下一步研究工作可结合局部搜索方法,从而减小支撑向量规模,降低对内存空间消耗;将算法应用到诸如蛋白质结构预测等实际问题的求解.另外,该方法还可以方便地应用到遗传算法、蚁群系统、粒子群等群体优化算法中。

致谢 在此,我们向对本文的工作给予支持和建议的所有评审专家和同行表示由衷的感谢。

References:

- [1] Beasley D, Bull DR, Martin RR. A sequential niche technique for multimodal function optimization. *Evolutionary Computation*, 1993,1(2):101–125. [doi: 10.1162/evco.1993.1.2.101]
- [2] Bradley P, Misura KMS, Baker D. Toward high-resolution de novo structure prediction for small proteins. *Science*, 2005,309(54742): 1868–1871. [doi: 10.1126/science.1113801]
- [3] Woo DK, Choi JH, Ali M, Jung HK. A novel multimodal optimization algorithm applied to electromagnetic optimization. *IEEE Trans. on Magnetics*, 2011,47(6):1667–1673. [doi: 10.1109/TMAG.2011.2106218]
- [4] Fletcher R, Powell MJD. A rapidly convergent descent method for minimization. *Computer Journal*, 1963,6(2):163–168. [doi: 10.1093/comjnl/6.2.163]
- [5] Nelder JA, Mead R. A simplex method for function minimization. *Computer Journal*, 1965,7(7):308–313. [doi: 10.1093/comjnl/7.4.308]
- [6] Lewis RM, Torczon V, Trosset MW. Direct search methods: Then and now. *Journal of Computational and Applied Mathematics*, 2000,124(1-2):191–207. [doi: 10.1016/S0377-0427(00)00423-4]
- [7] Khoo KG, Suganthan PN. Structural pattern recognition using genetic algorithms with specialized operators. *IEEE Trans. on Systems, Man, and Cybernetics*, 2003,33(1):156–165. [doi: 10.1109/TSMCB.2003.808185]
- [8] Storn R, Price K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 1997,11(4):341–359. [doi: 10.1023/A:1008202821328]
- [9] Kennedy J, Eberhart, R. Particle swarm optimization. In: *Proc. of the IEEE Int'l Conf. on Neural Networks*. New York: IEEE Press, 1995. 1942–1948. [doi: 10.1109/ICNN.1995.488968]
- [10] Rogers A, Prügell-Bennett A. Generic drift in genetic algorithm selection schemes. *IEEE Trans. on Evolutionary Computation*, 1999, 3(4):298–303. [doi: 10.1109/4235.797972]
- [11] Mahfoud SW. Crowding and preselection revisited. In: *Manner R, Manderick B, eds. Proc. of the Conf. on Parallel Problem Solving from Nature*. Amsterdam: Elsevier Science, 1992. 27–36.

- [12] Mengsheel OJ, Goldberg D. Probabilistic crowding: deterministic crowding with probabilistic replacement. In: Wolfgang B, ed. Proc. of the Genetic and Evolution Computation Conf. San Fransisco: Morgan Kaufmann Publishers, 1999. 409–416.
- [13] Miller BL, Show MJ. Genetic algorithms with dynamic niche sharing for multimodal function optimization. In: Proc. of the IEEE Int'l Conf. on Evolutionary Computation. New York: IEEE Press, 1996. 786–791. [doi: 10.1109/ICEC.1996.542701]
- [14] Lee CG, Cho DH, Jung HK. Niching genetic algorithm with restricted competition selection for multimodal function optimization. IEEE Trans. on Magnetics, 1999,35(3):1722–1725. [doi: 10.1109/20.767361]
- [15] Li JP, Balazs ME, Parks GT, Clarkson PJ. A species conserving genetic algorithm for multimodal function optimization. Evolutionary Computation, 2002,10(3):207–234. [doi: 10.1162/106365602760234081]
- [16] Shir OM, Bäck T. Dynamic niching in evolution strategies with covariance matrix adaptation. In: Proc. of the Conf. on Genetic and Evolution Computation (GECCO 2005). New York: IEEE Press, 2005. 2584–2591. [doi: 10.1109/CEC.2005.1555018]
- [17] Gustafson S, Burke EK. The speciating island model: An alternative parallel evolutionary algorithm. Journal of Parallel Distributed Computing, 2006,66(8):1025–1036. [doi: 10.1016/j.jpdc.2006.04.017]
- [18] Das S, Maity S, Qu BY, Suganthan PN. Real-Parameter evolutionary multimodal optimization—A survey of the state-of-the-art. Swarm and Evolutionary Computation, 2011,1(2):71–88. [doi: 10.1016/j.swevo.2011.05.005]
- [19] Yu XJ, Wang ZJ. The fitness sharing genetic algorithm with self-adaptive control of peaks radii. Acta Automatica Sinica, 2002, 28(5):816–820 (in Chinese with English abstract).
- [20] Li MQ, Kou JS. Coordinate multi-population genetic algorithms for multi-modal function optimization. Acta Automatica Sinica, 2002,28(4):497–504 (in Chinese with English abstract).
- [21] Guo GQ, Yu SY. Genetic drift analysis of recombination. Ruan Jian Xue Bao/Journal of Software, 2003,14(11):1875–1881 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/14/1875.htm>
- [22] Rönkkönen J. Continuous multimodal global optimization with differential evolution based methods [Ph.D. Thesis]. Lappeenranta: Lappeenranta University of Technology, 2009.
- [23] Cioppa AD, Stefano CD, Marcelli A. Where are the niches? Dynamic fitness sharing. IEEE Trans. on Evolutionary Computation, 2007,11(4):453–465. [doi: 10.1109/TEVC.2006.882433]
- [24] Wong KC, Wu CH, Mok RKP, Peng CB, Zhang ZL. Evolutionary multimodal optimization using the principle of locality. Information Sciences, 2012,194(1):138–170. [doi: 10.1016/j.ins.2011.12.016]
- [25] Li MQ, Lin D, Kou JS. A hybrid niching PSO enhanced with recombination-replacement crowding strategy for multimodal function optimization. Applied Soft Computing, 2012,12(3):975–987. [doi: 10.1016/j.asoc.2011.11.032]
- [26] Vitela JE, Castanos O. A sequential niching memetic algorithm for continuous multimodal function optimization. Applied Mathematics and Computation, 2012,218(17):8242–8259. [doi: 10.1016/j.amc.2011.05.051]
- [27] Stoean, C, Preuss M, Stoean R, Dumitrescu D. Multimodal optimization by means of a topological species conservation algorithm. IEEE Trans. on Evolutionary Computation, 2010,14(6):842–864. [doi: 10.1109/TEVC.2010.2041668]
- [28] Yao J, Kharna N, Grogono P. Bi-Objective multipopulation genetic algorithm for multimodal function optimization. IEEE Trans. on Evolutionary Computation, 2010,14(1):80–102. [doi: 10.1109/TEVC.2009.2017517]
- [29] Boyd S, Vandenberghe L. Convex Optimization. New York: Cambridge University Press, 2004. 67–112.
- [30] Andramonov M, Rubinov A, Glover B. Cutting angle methods in global optimization. Applied Mathematical Letters, 1999,12(3): 95–100. [doi: 10.1016/S0893-9659(98)00179-7]
- [31] Rubinov A, Andramonov M. Lipschitz programming via increasing convex-along-rays functions. Optimization Methods and Software, 1999,10(6):763–781. [doi: 10.1080/10556789908805740]
- [32] Bagirov AM, Rubinov AM. Global minimization of increasing positively homogeneous functions over the unit simplex. Annals of Operations Research, 2000,98(1-4):171–187. [doi: 10.1023/A:1019204407420]
- [33] Beliakov G. Cutting angle method—A tool for constrained global optimization. Optimization Methods and Software, 2004,19(2): 137–151. [doi: 10.1080/10556780410001647177]
- [34] Batten LM, Beliakov G. Fast algorithm for the cutting angle method of global optimization. Journal of Global Optimization, 2002, 24(2):149–161. [doi: 10.1023/A:1020256900863]

- [35] Beliakov G. Geometry and combinatorics of the cutting angle method. *Optimization*, 2003,52(4):379–394. [doi: 10.1080/02331930310001611556]
- [36] Price WL. Global optimization by controlled random search. *Journal of Optimization Theory and Applications*, 1983,40(3):333–348. [doi: 10.1007/BF00933504]
- [37] Ali MM, Khompatraporn C, Zabinsky ZB. A numerical evaluation of several stochastic algorithms on selected continuous global optimization test problems. *Journal of Global Optimization*, 2005,31(4):635–672. [doi: 10.1007/s10898-004-9972-2]

附中文参考文献:

- [19] 于歆杰,王赞基.自适应调整峰半径的适应值共享遗传算法.自动化学报,2002,28(5):816–820.
- [20] 李敏强,寇纪淞.多模态函数优化的协同多群体遗传算法.自动化学报,2002,28(4):497–504.
- [21] 郭观七,喻寿益.重组的遗传漂移分析.软件学报,2003,14(11):1875–1881. <http://www.jos.org.cn/1000-9825/14/1875.htm>



张贵军(1974—),男,山西阳泉人,博士,教授,主要研究领域为生物信息学、智能信息处理、全局优化理论及算法设计.

E-mail: zgi@zjut.edu.cn



冯远静(1976—),男,博士,副教授,主要研究领域为智能优化.

E-mail: fyjing@zjut.edu.cn



何洋军(1988—),男,硕士生,主要研究领域为智能优化.

E-mail: yjunhe@yahoo.com.cn



徐建明(1970—),男,博士,副教授,主要研究领域为智能控制.

E-mail: xumj@zjut.edu.cn



郭海锋(1977—),男,博士,副教授,主要研究领域为交通控制工程,人工智能.

E-mail: guohf@zjut.edu.cn