

3-25-2020

## An Estimation of Distribution Algorithm Based on Multiple Elites Sampling and Individuals Differential Search

Yu Fei

1. *Min Nan Normal University, School of Physics and Information Engineering, Zhangzhou 363000, China;;*

Ruifeng Wu

2. *East China Jiaotong University, School of Software, Nanchang 330013, China;*

Wei Bo

2. *East China Jiaotong University, School of Software, Nanchang 330013, China;*

Yinglong Zhang

1. *Min Nan Normal University, School of Physics and Information Engineering, Zhangzhou 363000, China;;*

*See next page for additional authors*

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research, Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

---

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

---

# An Estimation of Distribution Algorithm Based on Multiple Elites Sampling and Individuals Differential Search

## Abstract

**Abstract:** An estimation distribution algorithm based on the multiple elites sampling and the individuals differential search (EDA-M/D) is proposed. In EDA-M/D, the elites carry out the sampling to generate the offspring independently and enhance the exploration. Meanwhile, the variance of the population distribution is selected to control the sampling radius. Thus, the target of the population can be gradually transitioned from exploration to exploitation. If the elite population stagnates, the nonelites will choose the mean value of the elites distribution  $\mu$  and the population historical best solution as the two exemplars to execute a differential search operator, and then help the population jump out of a potential local optimum. Based on the adaptive strategy, two generation methods for the offspring, i.e., basing on the multiple elites sampling and the differential search, can be hybridized. Hence, the macro information of population and the micro information of individuals can be organically integrated. Experimental results show that EDA-M/D outperforms the other peer algorithms in the algorithm stability and the global optimal search capability.

## Keywords

estimation of distribution algorithm, multiple elites sampling, differential search, gene rectification

## Authors

Yu Fei, Ruifeng Wu, Wei Bo, Yinglong Zhang, and Xuewen Xia

## Recommended Citation

Yu Fei, Wu Ruifeng, Wei Bo, Zhang Yinglong, Xia Xuewen. An Estimation of Distribution Algorithm Based on Multiple Elites Sampling and Individuals Differential Search[J]. Journal of System Simulation, 2020, 32(3): 382-393.

## 多精英采样与个体差分学习的分布估计算法

喻飞<sup>1</sup>, 吴瑞峰<sup>2\*</sup>, 魏波<sup>2</sup>, 张应龙<sup>1</sup>, 夏学文<sup>1</sup>

(1. 闽南师范大学物理与信息工程学院, 福建 漳州 363000; 2. 华东交通大学软件学院, 江西 南昌 330013)

**摘要:** 提出了基于多精英采样和差分搜索的分布估计算法 EDA-M/D (Estimation distribution algorithm based on multiple elites sampling and individuals differential search)。EDA-M/D 利用多精英个体独立采样生成子代来提升算法全局搜索能力, 利用精英群体分布的  $\sigma^2$  约束采样半径, 实现种群从全局搜索逐步过度到局部搜索。当精英群体停滞时, 劣势个体借助精英群体的  $\mu$  和种群历史最优解进行差分搜索, 帮助种群跳出局部最优解。通过多精英采样与差分搜索的自适应协同实现种群宏观信息与个体微观信息的有机融合。实验结果表明 EDA-M/D 在稳定性和搜索能力方面均表现出明显的优势。

**关键词:** 分布估计算法; 多精英采样; 差分搜索; 基因修复

中图分类号: TP301

文献标识码: A

文章编号: 1004-731X (2020) 03-0382-12

DOI: 10.16182/j.issn1004731x.joss.18-0836

## An Estimation of Distribution Algorithm Based on Multiple Elites Sampling and Individuals Differential Search

Yu Fei<sup>1</sup>, Wu Ruifeng<sup>2\*</sup>, Wei Bo<sup>2</sup>, Zhang Yinglong<sup>1</sup>, Xia Xuewen<sup>1</sup>

(1. Min Nan Normal University, School of Physics and Information Engineering, Zhangzhou 363000, China;

2. East China Jiaotong University, School of Software, Nanchang 330013, China)

**Abstract:** An estimation distribution algorithm based on the multiple elites sampling and the individuals differential search (EDA-M/D) is proposed. In EDA-M/D, the elites carry out the sampling to generate the offspring independently and enhance the exploration. Meanwhile, the variance of the population distribution  $\sigma^2$  is selected to control the sampling radius. Thus, the target of the population can be gradually transited from exploration to exploitation. If the elite population stagnates, the nonentities will choose the mean value of the elites distribution  $\mu$  and the population historical best solution as the two exemplars to execute a differential search operator; and then help the population jump out of a potential local optimum. Based on the adaptive strategy, two generation methods for the offspring, i.e., basing on the multiple elites sampling and the differential search, can be hybridized. Hence, the macro information of population and the micro information of individuals can be organically integrated. Experimental results show that EDA-M/D outperforms the other peer algorithms in the algorithm stability and the global optimal search capability.

**Keywords:** estimation of distribution algorithm; multiple elites sampling; differential search; gene rectification

## 引言

分布估计算法 (Estimation of distribution



收稿日期: 2018-12-17 修回日期: 2019-05-13;  
基金项目: 国家自然科学基金(61663009, 61762036, 61806204, 61876136), 江西省自然科学基金(20171B AB202012), 福建省本科高校重大教育教学改革研究项目(FBJG20180015), 江西省交通厅科研项目(2017D0038);  
作者简介: 喻飞(1981-), 男, 湖北钟祥, 博士, 副教授, 研究方向为计算智能及其应用, 机器学习等。

Algorithm, EDA)是进化计算领域兴起的一种新型的启发式优化算法<sup>[1-2]</sup>。由于 EDA 基本上都采用与模型相匹配的统计学习方法来学习概率模型, 这也使得 EDA 具有较好的收敛性。但当用来构建分布概率模型的父代种群分布在局部最优解附近时, 基于该模型生成的后代种群容易“早熟”收敛。为了克服 EA (evolutionary algorithms, EAs)中普遍存在的

<http://www.china-simulation.com>

问题, 许多学者对 EDA 算法提出了诸多的改良策略, 其核心思想是在更好利用种群全局信息的同时, 赋予个体更丰富的搜索行为, 合理地平衡种群多样性和收敛速度, 从而提升 EDA 在不同问题上的综合性能<sup>[3-6]</sup>。

为了平衡种群多样性和收敛性之间的矛盾, 满足算法不同时期对搜索行为的需求, 本文提出了一种基于多精英采样和差分搜索的分布估计算法(An estimation of distribution algorithm based on multiple elites sampling and individuals differential search, EDA-M/D)。在 EDA-M/D 中, 多个精英个体在其邻域内通过采样来生成子代个体, 利用宏观信息指导种群进化; 同时, 为了合理利用劣势个体, 在每代中利用精英个体的信息对劣势个体进行基因修复, 尽可能保留其良好的基因片段, 在微观层面进行搜索。另一方面, 当种群搜索出现停滞时, EDA-M/D 则利用精英群体信息及最优个体指导劣势个体进行差分学习, 在保持种群多样性的同时, 增强算法跳出局部最优的能力。EDA-M/D 通过自适应地采用两种模式生成子代种群, 有机融合种群宏观信息和个体的微观信息, 实现种群在宏观层面和微观层面进行协同搜索。

## 1 EDA

EDA 在每代的搜索过程中主要包括以下几个步骤: (1) 从当代种群中选择部分(或全部)个体; (2) 通过所选择的群体构建其分布概率模型; (3) 借助概率模型采样生成新种群。

PBIL 算法 (Population based incremental learning)<sup>[7]</sup>是由 Baluja 提出用以求解二进制优化问题, 该算法模型简单, 计算效率高, 被公认是最早的 EDA 模型。此后, Sebag 又将 PBIL 进行了扩展, 提出了适用于连续函数优化的 PBILc (Population-based incremental learning to continuous search spaces)<sup>[8]</sup>算法, 进一步丰富了其应用领域。由于本文提出的 EDA-M/D 中概率模型的初始化方式和 PBILc 算法相同, 其更新方式也类似, 下面

将以 PBILc 为例, 对 EDA 的核心思想进行简单的介绍。

### 1.1 PBILc

PBILc 算法可视为一种基于高斯模型(Gaussian model)的实数编码 EDA。在 PBILc 中, 对选中的  $N$  个个体  $X_i=[x_{i,1}, x_{i,2}, \dots, x_{i,D}]$  ( $D$  为问题变量维数), 构建一个基于高斯分布  $N(\mu_j, \sigma_j^2)$  ( $1 \leq j \leq D$ ) 的概率模型, 然后根据该概率模型, 在整个解空间采样产生  $N$  个样本构成子代种群。可以看出, PBILc 在每代中利用  $\mu_j$  约束子代种群的中心, 利用  $\sigma_j$  控制其多样性。在理想情况下, 随着种群的进化,  $\mu_j$  将逐渐逼近全局最优解的位置, 而  $\sigma_j$  将越来越小, 从而使种群聚集在全局最优解的附近。PBILc 主要流程如下:

step 1:  $t=1$ ; 随机产生规模为  $N$  的初始种群并计算所有个体的适应值。

step 2: 依适应值选择  $K$  ( $K \leq N$ ) 个精英个体根据式(1)初始化概率模型。

$$\begin{cases} \mu_j^t = \frac{\sum_{i=1}^K x_{i,j}^t}{K} \\ \sigma_j^t = \sqrt{\frac{\sum_{i=1}^K (x_{i,j}^t - \mu_j^t)^2}{K}} \end{cases} \quad (1)$$

式中:  $x_{i,j}^t$  ( $1 \leq i \leq K$ ) 为第  $t$  代时第  $i$  个精英个体第  $j$  维的值;  $D$  为目标问题维度;  $\mu_j^t$  和  $\sigma_j^t$  分别为当前精英种群在第  $t$  代时第  $j$  维的均值与标准差。

step 3: 通过式(2)进行正态采样, 得到  $N$  个子代个体  $X_i^{t+1}=[x_{i,1}^{t+1}, x_{i,2}^{t+1}, \dots, x_{i,D}^{t+1}]$  ( $1 \leq i \leq N$ ), 其中

$$x_{i,j}^{t+1} = \mu_j^t + \sigma_j^t \cdot rand \quad (2)$$

这里,  $rand$  为符合标准高斯分布的随机数。

step 4: 评价新种群; 分别采用式(3), (4)所示的线性学习方式更新高斯分布  $N(\mu, \sigma^2)$  的均值与标准差。

$$\mu_j^{t+1} = (1 - rate) \cdot \mu_j^t + rate \cdot (x_{b1,j}^{t+1} + x_{b2,j}^{t+1} - x_{w,j}^{t+1}) \quad (3)$$

$$\sigma_j^{t+1} = (1 - rate) \cdot \sigma_j^t + rate \cdot \sqrt{\frac{\sum_{i=1}^K (x_{i,k,j}^{t+1} - \bar{x}_j^{t+1})^2}{K}} \quad (4)$$

式中:  $rate$  为学习因子, 一般取值为  $[0, 1]$  之间的实数。当  $rate$  越大时, 高斯模型越倾向于采用当代种群信息更新, 反之则更少地利用当代信息而保留更多的历史信息。 $x_{b1,j}^{t+1}$ ,  $x_{b2,j}^{t+1}$  和  $x_{w,j}^{t+1}$  分别表示种群中最优个体、次优个体和最差个体在第  $t+1$  代时第  $j$  维的值;  $x_{i,j}^{t+1}$  ( $1 \leq i \leq K$ ) 为第  $t+1$  代时第  $i$  个个体第  $j$  维的值;  $\bar{x}_j^{t+1}$  为第  $t+1$  代时  $K$  个精英个体在第  $j$  维的均值, 即  $\bar{x}_j^{t+1} = \sum_{i=1}^K x_{i,j}^{t+1} / K$ 。

step 5:  $t=t+1$ ; 若不满足停机条件, 则跳转到 step 3; 否则, 输出结果。

## 1.2 EDA 研究现状




在 EDA 研究初期, 一些学者将问题视为变量相互独立, 研究了单变量模型(也称变量独立模型)的 EDA, PBIL 和 PBILc 就属于此类。Mühlenbein 提出了另一种单变量模型的 EDA: UMDA (Univariate marginal distribution algorithm)<sup>[9-10]</sup>。随后, Larranaga 提出了应用于连续函数优化的 UMDAc<sup>[11]</sup>。Harik 等提出了 cGA (compact genetic algorithm)<sup>[12]</sup> 考虑到很多复杂工程问题中存在变量耦合的现象, 一些学者在 EDA 中引入了不同的多变量相关的概率模型用以刻画实际应用中存在的变量耦合关系。如 Bonet 用一种链式关系表征 2 个变量间的关系, 提出了 MIMIC 算法 (Mutual information maximization for input clustering)<sup>[13]</sup>, Baluja 提出了 COMIT (Combining optimizers with mutual information trees)<sup>[14]</sup> 算法。此后, 更为复杂的多变量 EDA 相继涌现。如采用基于概率图模型的 EDA<sup>[15]</sup> 来解决多变量耦合的优化问题。但该算法需要事先确定用以表示变量耦合关系的图模型, 这也使得对于很多“黑箱”优化问题不能直接采用该算法进行求解。

由于 EDA 和 EAs 分别利用种群的宏观信息和个体的微观信息指导优化过程, 各有其优缺点, 因此, 如何有机融合这两类智能算法, 充分发挥各自的优势, 成为了近年的研究热点, 也涌现出了一些非常优秀的成果。第一种融合策略是将其它 EAs

或搜索策略作为 EDA 的辅助操作算子, 提升 EDA 在复杂多峰问题中的性能。Qiu<sup>[16]</sup> 利用不同的遗传算子在父代种群周围进行局部搜索, 开辟新的搜索区域。刘小龙将则融合了细菌的趋化性运动和分布估计算法的思想, 提出了 EDA-BFO<sup>[17]</sup>。另一种融合策略是在种群进化过程中, 通过一定的策略来选择 EDA 或 EAs 来生成子代种群。如 El-Abd 将粒子群算法 (Particle Swarm Optimization, PSO) 与 EDA 相结合, 提出了 EDA-PSO<sup>[18]</sup>。Shao<sup>[19]</sup> 则依照概率来选择使用 DE 或 EDA 来生成子代个体。2004 年, Jose Pena 将遗传算法 (Genetic Algorithm, GA) 与 EDA 进行了融合, 提出了 GA-EDA<sup>[20]</sup>。

以往的研究表明, 尽管多变量模型的 EDA 能更准确地刻画变量间的耦合关系, 但在没有先验知识的前提下要通过种群进化过程中学习出精确的多变量相关的概率分布模型是极其耗时的, 从而也降低了其实用性。因此, 在单变量模型 EDA 的基础上, 有机融合其它并行搜索策略、合理利用不同个体的隐含信息来引导种群进行搜索可以有效缓解搜索效率与求解精度之间的矛盾。本文的研究正是基于这一出发点。

## 2 EDA-M/D

EDA 中子代种群的生成是对父代群体整体信息的利用, 因而更加强调种群宏观的进化方向, 具有良好全局搜索能力。同时, 如果以精英种群构建的概率模型的均值处于全局最优解附近时, 种群也将具有较快的收敛速度。然而, 对于复杂的多峰优化问题, 基于该概率模型的采样方式容易使得所生成的子代种群过早丧失多样性, 进而出现“早熟”现象。若目标函数解空间的适应度值等高线图如图 1 所示, 当精英种群(见点 ) 分布模型的样本中心  $\mu_j^t$  (见点 ) 位于局部最优解附近, 则依此点采样生成的新个体难以跳出其所在局部最优解, 种群也容易陷入局部最优。此时, 若能以每个精英个体为中心点进行采样, 则其生成的子代个体能以更大的概率搜索到全局最优解(见点 )。



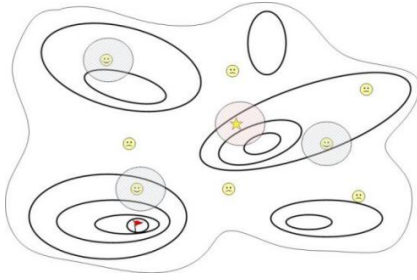


图 1 多精英个体采样与传统 EDA 采样结果对比

Fig. 1 Comparison result between multiple elites sampling method and traditional EDA sampling method

为了合理利用种群的全局信息与精英个体的优势信息, 本文提出了 EDA-M/D。该算法一方面利用部分精英个体的统计信息构建其分布概率模型, 基于该模型, 多个精英个体在其邻域内并行采样来生成子代个体, 从而在宏观层面指导种群进化; 同时, 为了合理利用劣势个体, 在每代中利用精英个体的信息对劣势个体进行基因修复, 尽可能保留其良好的基因片段, 在微观层面进行搜索。另一方面, 当种群搜索出现停滞时, EDA-M/D 则利用精英群体信息及最优个体指导劣势个体进行差分学习, 增强算法跳出局部最优的能力。EDA-M/D 通过自适应地采用两种学习模式生成子代种群, 实现两种模型有机融合, 进而指导种群在宏观层面和微观层面进行协同搜索。

## 2.1 多精英采样

在第  $t$  代时, EDA-M/D 首先对种群按适应值排序:  $f(X_{i_1}^t) \leq f(X_{i_2}^t) \leq \dots \leq f(X_{i_N}^t)$ ,  $N$  为种群规模。选择其中的  $K$  ( $K < N$ ) 个精英个体  $\mathbf{X}_{i_k}^t = [x_{i_k,1}^t, x_{i_k,2}^t, \dots, x_{i_k,D}^t]$ , ( $k=1, 2, \dots, K$ ) 按式(5)构建当前精英群体分布概率模型。不失一般性, 本文以最小化问题作为研究对象, 将个体对应的函数值作为其适应值, 即函数值越大, 适应值越差。

$$\begin{cases} \mu_j^t = \frac{\sum_{i=1}^K x_{i_k,j}^t}{K} \\ \sigma_j^t = \sqrt{\frac{\sum_{i=1}^K (x_{i_k,j}^t - \mu_j^t)^2}{K}} \end{cases} \quad (5)$$

为了合理利用历史精英群体的搜索经验, 上述概率模型需要向历史概率分布进行学习, 以防止当

前精英群体过快收敛所造成的“早熟”现象。分布概率模型的学习过程如式(6), (7)所示。其中,  $rate$  为学习因子, 取值在 0 到 1 之间。

$$\mu_j^{t+1} = (1 - rate) \cdot \mu_j^t + rate \cdot \frac{\sum_{k=1}^K x_{i_k,j}^{t+1}}{K} \quad (6)$$

$$\sigma_j^{t+1} = (1 - rate) \cdot \sigma_j^t + rate \cdot \sqrt{\frac{\sum_{k=1}^K (x_{i_k,j}^{t+1} - \bar{x}_j^{t+1})^2}{K}} \quad (7)$$

构建了当代精英群体的分布概率模型后, 选择当前种群中的  $E$  ( $E < N$ ) 个精英个体  $\mathbf{X}_{i_k}^t$ , ( $i=1, 2, \dots, E$ ) 以自身位置作为采样中心, 以  $\sigma_j^{t+1}$  为标准差进行高斯采样, 生成新个体  $\mathbf{TmpX}_{i_k} = [tmpx_{i_k,1}, tmpx_{i_k,2}, \dots, tmpx_{i_k,D}]$ 。具体如式(8)所示。需要注意的是, 这里  $E$  的取值并非一定要等于  $K$ 。

$$tmpx_{i_k,j} = \begin{cases} \text{Guass}(x_{i_k,j}^t, \sigma_j^t), & \text{if } (rand \leq Lp) \\ x_{i_k,j}^t, & \text{else} \end{cases} \quad (8)$$

式中:  $Lp$  用来控制精英个体  $\mathbf{X}_{i_k}^t$  第  $j$  个基因位执行采样操作的概率;  $\sigma_j^{t+1}$  用来控制采样的范围。生成个体  $\mathbf{TmpX}_{i_k}$  后, EDA-M/D 采用贪心策略进行保留, 即子代个体  $\mathbf{TmpX}_{i_k}$  的适应值  $f(\mathbf{TmpX}_{i_k})$  比其父代个体  $\mathbf{X}_{i_k}^t$  的适应值更优时, 保留该子代个体, 抛弃相应父代个体。具体策略如式(9)所示:

$$\mathbf{X}_{i_k}^{t+1} = \begin{cases} \mathbf{X}_{i_k}^t, & \text{if } (f(\mathbf{TmpX}_{i_k}) \geq f(\mathbf{X}_{i_k}^t)) \\ \mathbf{TmpX}_{i_k}, & \text{else} \end{cases} \quad (9)$$

## 2.2 劣势个体基因片段修复

尽管其它的  $N-E$  个劣势个体适应值较差, 但若简单地将其抛弃有可能使其蕴含的有益信息丢失, 造成计算资源的浪费。因此, EDA-M/D 借助劣势个体与精英个体的杂交实现劣势个体基因片段的修复, 从而达到保留其有益信息的目的。

劣势个体  $\mathbf{X}_{i_k}^t = [x_{i_k,1}^t, x_{i_k,2}^t, \dots, x_{i_k,D}^t]$ ,  $k=(E+1, E+2, \dots, N)$  在执行基因修复时, 首先会从本代精英群体中任选一精英个体  $\mathbf{X}_{i_m}^t$  ( $1 \leq m \leq E$ ) 为样本点通过高斯采样获得基因, 按照概率  $Lp$  优化劣势个体的基因片段, 得到候选个体  $\mathbf{TmpX}_{i_k} = [tmpx_{i_k,1}, tmpx_{i_k,2}, \dots, tmpx_{i_k,D}]$ 。具体如式(10)所示:

$$tmpx_{i_k,j} = \begin{cases} \text{Guass}(x_{i_m,j}^t, \sigma_j^t), & \text{if } (rand \leq Lp) \\ x_{i_k,j}^t, & \text{else} \end{cases} \quad (10)$$

式中： $L_p$  用以控制劣势个体第  $j$  个基因位执行修复操作的概率。类似的，对新生成的子个体也采用式(9)的贪心策略保留个体。

EDA-M/D 在上述 2 个策略中均采用式(9)所示的贪心保留策略可用图 2 来表示。可以看出，通过当代精英个体并行独立采样得到新的子个体后，若采用 EDA 常用的保留策略，则距离全局最优解最近的精英个体(即图中左下角的精英个体)采样得到的子代个体会因为劣于另外 2 个精英个体及相应采样得到的子个体而被抛弃掉，这也使得子代种群远离了全局最优解而陷入了局部最优。反之，若采用式(9)所示的保留策略，距离全局最优解附近的精英个体采样得到的子代个体尽管适应值并非属于精英群体，但依然会被保留下来，从而保证了种群的全局搜索能力。

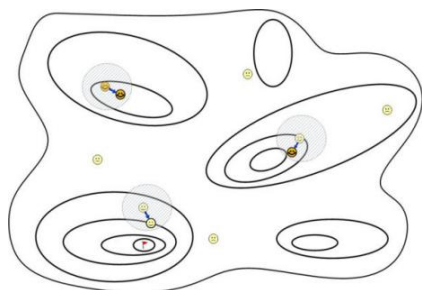


图 2 贪心保留策略示意图  
Fig. 2 Process of greedy reservation strategy

### 2.3 劣势个体差分学习

通过上述 MES 和 NGR 策略生成第  $t+1$  代的个体后，若新种群中的  $E$  个精英的适应值与  $t-1$  代中的  $E$  个精英的适应值完全相同，则说明基于上述子代生成策略无法挖掘有益信息指导种群的搜索，本文将这种情况视为精英群体出现停滞。当精英群体连续停滞代数  $stay$  达到设定阈值  $staymax$  时， $N-E$  个劣势个体  $X_{i_k}^t (k = E + 1, E + 2, \dots, N)$  向式(5)所示的精英分布概率模型的  $\mu_j^{t+1}$  和当前种群最优个体  $X_{best}^{t+1} = [x_{best,1}^{t+1}, x_{best,2}^{t+1}, \dots, x_{best,D}^{t+1}]$  进行差分学习来产生新的个体  $TmpX_{i_k}$  学习方式如式(11)所示。

该学习方式的意义在于：当劣势个体已经无法从 NGR 策略中获得更多有益信息时，则以精英种

群中心和最优个体作为学习目标，利用隐含信息来引导种群进行搜索。通过差分学习方式探索其路径上可能存在的最优解。同时，在进化后期 NDL 更利于种群的开采工作。

### 2.4 EDA-M/D 流程

综上所述，EDA-M/D 流程可描述如下：

step 1: 初始化： $t=1, fes=0, stay=0$ ，随机初始化种群  $Pop = [X_1^t, X_2^t, \dots, X_N^t]$ ， $N$  为种群规模；设置  $rate, staymax, Li, Lp$  及  $max\_FEs$ ；

step 2: 评价第  $t$  代种群，依适应值排序，满足  $f(X_{i_1}^t) \leq f(X_{i_2}^t) \leq \dots \leq f(X_{i_N}^t)$ ； $fes=fes+N$ ；

step 3: 记录种群最优个体  $X_{best}^t = X_{i_1}^t$ ；

step 4: 选择最优的  $K$  个个体  $X_{i_k}^t (1 \leq k \leq K)$

按照式(5)初始化概率模型；

step 5: (MES 策略)：对第  $t$  代中的  $E$  个精英个体  $X_{i_k}^t (1 \leq k \leq E)$  利用式(8)~(9)生成  $E$  个子个体  $X_{i_k}^t$ ；

step 6: (NGR 策略)：对第  $t$  代中的  $N-E$  个劣势个体  $X_{i_k}^t (E < k \leq N)$  利用式(9)~(10)生成  $N-E$  个子个体  $X_{i_k}^{t+1}$ ；

step 7:  $t=t+1$ ；评价第  $t$  代种群，依适应值排序，满足  $f(X_{i_1}^t) \leq f(X_{i_2}^t) \leq \dots \leq f(X_{i_N}^t)$ ； $fes=fes+N$ ；

step 8: 若  $f(X_{i_k}^t) = f(X_{i_k}^{t-1}) (1 \leq k \leq E)$ ，

$stay=stay+1$ ；否则  $stay=0$ ；

step 9: (NDL 策略)：若  $stay < staymax$ ，转至 step 11；否则，对  $N-E$  个劣势个体  $X_{i_k}^t (E < k \leq N)$  利用式(9)，式(11)生成  $N-E$  个个体  $X_{i_k}^t$ ；评价新生成的  $N-E$  个个体， $fes=fes+N-E$ ； $stay=0$ ；

$$tmpx_{i_k,j} = \begin{cases} x_{i_k,j}^{t+1} + rand_1(\mu_j^{t+1} - x_{i_k,j}^{t+1}) + \\ rand_2(x_{best,j}^{t+1} - x_{i_k,j}^{t+1}), & \text{if}(rand \leq Li) \\ x_{i_k,j}^{t+1}, & \text{else} \end{cases} \quad (11)$$

step 10: 对种群依适应值排序，满足  $f(X_{i_1}^t) \leq f(X_{i_2}^t) \leq \dots \leq f(X_{i_N}^t)$ ；

step 11: 按照式(6)~(7)更新高斯概率模型；

step 12: 若未满足停机条件，转至 step 5；否则，结束并输出结果。

学习完成后, 采用式(9)所示的贪心策略保留更优的子代个体实验。

### 3 仿真实验

#### 3.1 实验设置

为了验证 EDA-M/D 的综合性能, 本文选择了 6 种已有算法在 CEC2017 测试集<sup>[21]</sup>上进行了对比实验, 包括 PBILc<sup>[8]</sup>, UMDAc<sup>[11]</sup>, EDA-BFO<sup>[17]</sup>, GA-EDA<sup>[20]</sup>, EDA-PSO<sup>[18]</sup>, AIL-DE/EDA<sup>[19]</sup>。其中, PBILc 和 UMDAc 是单变量模型中求解连续域优化问题最有代表性的 EDA 算法, 可以为 EDA-M/D 性能表现提供参考基准。EDA-BFO 结合细菌觅食算法并采用 EDA 提升种群多样性。GA-EDA 按比率融合了 GA 和 EDA, 从两种方法中获得有益的指导信息。EDA-PSO 则通过自适应的选择比率协调 PSO 和 EDA 的搜索行为。这 3 种混合 EDA,

能全面反映出混合不同 EAs 的 EDA 在该测试集上的表现; AIL-DE/EDA 则在融合差分进化算法 (Differential Evolution, DE) 的同时, 还加入了自适应增量学习策略协调算法全局收敛性和局部寻优能力, 是目前 EAs 研究领域的一种发展趋势。实验选取的 6 种算法一定程度上反映了当前 EDAs 的研究方向及成果, 能为 EDA-M/D 综合性能提供全面客观的参照。

通过实验, EDA-M/D 中的参数取  $rate=0.6$ ,  $Li=0.4$ ,  $Lp=0.4$ ,  $staymax=5$ ,  $N=30$ ,  $K=N/3$ ,  $E=N/6$ 。对比算法涉及的重要参数设置均参照原文推荐值。每个测试函数的维数  $D=30$ , 最大评价次数  $max\_FEs=10\ 000 \times D$ 。实验中, 每个算法对 30 个测试函数独立运行 50 次, 评价指标包括 50 次求解结果的均值 (Mean) 及中位数 (Median), 最优结果均用阴影背景来标记。对测试集中 4 类不同函数的实验结果分别如表 1~4 所示。

表 1 对比算法在 CEC2017 测试集单峰函数( $f_1 \sim f_3$ )上的实验结果

Tab. 1 Comparison results of 7 peer algorithms on unimodal functions ( $f_1 \sim f_3$ ) in CEC2017 test suite

| 函数    | 性能     | EDA-M/D  | UMDAc    | PBILc    | BOFEDA   | GAEDA    | PSOEDA   | AIL-DE/ED |
|-------|--------|----------|----------|----------|----------|----------|----------|-----------|
| $f_1$ | Mean   | 3.46E+03 | 5.09E+10 | 6.73E+03 | 4.35E+10 | 2.38E+07 | 1.07E+10 | 3.47E+09  |
|       | Median | 2.02E+03 | 5.15E+10 | 4.46E+03 | 4.33E+10 | 2.06E+07 | 9.94E+09 | 3.46E+09  |
| $f_2$ | Mean   | 4.22E+08 | 2.53E+39 | 7.00E+16 | 1.67E+48 | 1.44E+19 | 1.33E+38 | 2.13E+38  |
|       | Median | 3.05E+03 | 4.18E+38 | 4.19E+09 | 1.44E+46 | 1.19E+16 | 5.65E+32 | 6.12E+36  |
| $f_3$ | Mean   | 1.64E+01 | 1.10E+05 | 1.97E+04 | 2.01E+05 | 2.04E+04 | 9.03E+04 | 2.74E+05  |
|       | Median | 2.91E-05 | 1.13E+05 | 1.39E+04 | 2.02E+05 | 1.97E+04 | 8.79E+04 | 2.68E+05  |

表 2 对比算法在 CEC2017 测试集多峰函数( $f_4 \sim f_{10}$ )上的实验结果

Tab. 2 Comparison results of 7 peer algorithms on multimodal functions ( $f_4 \sim f_{10}$ ) in CEC2017 test suite

| 函数       | 性能     | EDA-M/D  | UMDAc    | PBILc    | BOFEDA   | GAEDA    | PSOEDA   | AIL-DE/ED |
|----------|--------|----------|----------|----------|----------|----------|----------|-----------|
| $f_4$    | Mean   | 3.20E+01 | 9.68E+03 | 8.81E+01 | 1.22E+04 | 1.45E+02 | 1.34E+03 | 4.54E+02  |
|          | Median | 3.41E+01 | 9.69E+03 | 8.79E+01 | 1.19E+04 | 1.43E+02 | 1.09E+03 | 4.60E+02  |
| $f_5$    | Mean   | 2.31E+01 | 4.30E+02 | 3.71E+01 | 4.69E+02 | 3.74E+01 | 2.78E+02 | 2.21E+02  |
|          | Median | 2.39E+01 | 4.32E+02 | 2.62E+01 | 4.62E+02 | 3.71E+01 | 2.74E+02 | 2.24E+02  |
| $f_6$    | Mean   | 4.11E-05 | 8.47E+01 | 5.31E-03 | 1.00E+02 | 2.88E-01 | 6.45E+01 | 1.04E+01  |
|          | Median | 1.37E-07 | 8.52E+01 | 1.41E-06 | 9.96E+01 | 2.29E-01 | 6.55E+01 | 1.04E+01  |
| $f_7$    | Mean   | 5.16E+01 | 1.49E+03 | 1.88E+02 | 1.10E+03 | 8.35E+01 | 7.24E+02 | 2.78E+02  |
|          | Median | 4.84E+01 | 1.50E+03 | 1.91E+02 | 1.11E+03 | 8.07E+01 | 7.16E+02 | 2.80E+02  |
| $f_8$    | Mean   | 2.64E+01 | 3.86E+02 | 3.01E+01 | 4.46E+02 | 3.55E+01 | 2.42E+02 | 2.21E+02  |
|          | Median | 2.34E+01 | 3.89E+02 | 2.46E+01 | 4.44E+02 | 3.58E+01 | 2.37E+02 | 2.22E+02  |
| $f_9$    | Mean   | 1.62E+00 | 1.40E+04 | 2.50E-14 | 1.73E+04 | 1.04E+02 | 6.97E+03 | 1.15E+03  |
|          | Median | 8.95E-02 | 1.42E+04 | 0.00E+00 | 1.71E+04 | 8.55E+01 | 6.55E+03 | 1.15E+03  |
| $f_{10}$ | Mean   | 3.65E+03 | 7.26E+03 | 9.33E+02 | 8.25E+03 | 2.16E+03 | 7.71E+03 | 7.52E+03  |
|          | Median | 4.21E+03 | 7.31E+03 | 8.78E+02 | 8.28E+03 | 2.17E+03 | 7.87E+03 | 7.62E+03  |



表 3 对比算法在 CEC2017 测试集混合函数( $f_{11}\sim f_{20}$ )上的实验结果

Tab. 3 Comparison results of 7 peer algorithms on hybrid functions ( $f_{11}\sim f_{20}$ ) in CEC2017 test suite

| 函数       | 性能     | EDA-M/D  | UMDAc    | PBILc    | BOFEDA   | GAEDA    | PSOEDA   | AIL-DE/ED |
|----------|--------|----------|----------|----------|----------|----------|----------|-----------|
| $f_{11}$ | Mean   | 4.65E+01 | 6.14E+03 | 8.58E+01 | 2.01E+04 | 3.59E+02 | 8.50E+02 | 9.62E+02  |
|          | Median | 3.78E+01 | 6.37E+03 | 7.64E+01 | 1.99E+04 | 2.52E+02 | 6.59E+02 | 9.02E+02  |
| $f_{12}$ | Mean   | 4.69E+04 | 5.96E+09 | 6.89E+05 | 6.56E+09 | 3.83E+06 | 6.09E+08 | 5.95E+08  |
|          | Median | 2.35E+04 | 6.00E+09 | 3.90E+05 | 6.10E+09 | 3.19E+06 | 3.30E+08 | 5.72E+08  |
| $f_{13}$ | Mean   | 1.47E+04 | 2.41E+09 | 2.13E+04 | 5.81E+09 | 4.43E+04 | 2.24E+08 | 1.18E+08  |
|          | Median | 1.03E+04 | 2.51E+09 | 1.99E+04 | 5.40E+09 | 3.56E+04 | 4.75E+06 | 1.14E+08  |
| $f_{14}$ | Mean   | 1.59E+04 | 4.95E+05 | 6.26E+04 | 6.46E+06 | 7.09E+05 | 3.87E+05 | 1.69E+05  |
|          | Median | 1.03E+04 | 5.27E+05 | 4.90E+04 | 5.21E+06 | 6.48E+05 | 8.46E+04 | 1.48E+05  |
| $f_{15}$ | Mean   | 5.86E+03 | 1.79E+08 | 8.19E+03 | 7.10E+08 | 2.92E+03 | 1.07E+07 | 1.65E+06  |
|          | Median | 3.18E+03 | 1.85E+08 | 3.96E+03 | 5.43E+08 | 2.35E+03 | 7.78E+04 | 1.61E+06  |
| $f_{16}$ | Mean   | 3.91E+02 | 2.93E+03 | 1.50E+02 | 3.70E+03 | 6.65E+02 | 1.77E+03 | 1.99E+03  |
|          | Median | 3.93E+02 | 2.96E+03 | 1.02E+02 | 3.60E+03 | 6.69E+02 | 1.75E+03 | 2.02E+03  |
| $f_{17}$ | Mean   | 1.11E+02 | 1.38E+03 | 7.38E+01 | 2.20E+03 | 2.51E+02 | 9.13E+02 | 8.21E+02  |
|          | Median | 1.05E+02 | 1.41E+03 | 6.26E+01 | 2.00E+03 | 2.56E+02 | 8.97E+02 | 8.40E+02  |
| $f_{18}$ | Mean   | 2.38E+05 | 8.60E+06 | 6.29E+05 | 1.03E+08 | 8.19E+05 | 2.56E+06 | 6.05E+06  |
|          | Median | 1.39E+05 | 7.93E+06 | 3.80E+05 | 9.37E+07 | 6.50E+05 | 6.17E+05 | 5.50E+06  |
| $f_{19}$ | Mean   | 4.70E+03 | 3.07E+08 | 9.94E+03 | 7.65E+08 | 4.50E+03 | 1.72E+07 | 1.42E+06  |
|          | Median | 2.48E+03 | 3.00E+08 | 4.36E+03 | 6.93E+08 | 2.47E+03 | 4.01E+05 | 1.40E+06  |
| $f_{20}$ | Mean   | 1.59E+02 | 7.98E+02 | 7.32E+01 | 1.31E+03 | 3.16E+02 | 7.83E+02 | 6.36E+02  |
|          | Median | 1.72E+02 | 8.07E+02 | 4.05E+01 | 1.34E+03 | 2.77E+02 | 8.07E+02 | 6.25E+02  |

表 4 对比算法在 CEC2017 测试集复合函数( $f_{21}\sim f_{30}$ )上的实验结果

Tab. 4 Comparison results of 7 peer algorithms on composite functions ( $f_{21}\sim f_{30}$ ) in CEC2017 test suite

| 函数       | 性能     | EDA-M/D  | UMDAc    | PBILc    | BOFEDA   | GAEDA    | PSOEDA   | AIL-DE/ED |
|----------|--------|----------|----------|----------|----------|----------|----------|-----------|
| $f_{21}$ | Mean   | 2.25E+02 | 5.83E+02 | 2.24E+02 | 6.60E+02 | 2.34E+02 | 4.56E+02 | 4.27E+02  |
|          | Median | 2.25E+02 | 5.83E+02 | 2.23E+02 | 6.61E+02 | 2.34E+02 | 4.48E+02 | 4.27E+02  |
| $f_{22}$ | Mean   | 6.49E+02 | 5.65E+03 | 4.22E+02 | 8.17E+03 | 1.11E+03 | 5.43E+03 | 7.49E+03  |
|          | Median | 1.00E+02 | 5.73E+03 | 1.00E+02 | 8.21E+03 | 1.35E+02 | 7.35E+03 | 7.82E+03  |
| $f_{23}$ | Mean   | 3.77E+02 | 9.37E+02 | 3.73E+02 | 1.31E+03 | 4.03E+02 | 7.29E+02 | 5.90E+02  |
|          | Median | 3.77E+02 | 9.41E+02 | 3.73E+02 | 1.31E+03 | 4.04E+02 | 7.22E+02 | 5.91E+02  |
| $f_{24}$ | Mean   | 4.58E+02 | 1.05E+03 | 4.48E+02 | 1.54E+03 | 4.84E+02 | 7.27E+02 | 6.77E+02  |
|          | Median | 4.52E+02 | 1.06E+03 | 4.46E+02 | 1.54E+03 | 4.84E+02 | 7.09E+02 | 6.77E+02  |
| $f_{25}$ | Mean   | 3.87E+02 | 4.11E+03 | 3.87E+02 | 3.98E+03 | 4.27E+02 | 8.45E+02 | 4.96E+02  |
|          | Median | 3.87E+02 | 4.17E+03 | 3.87E+02 | 3.88E+03 | 4.25E+02 | 8.22E+02 | 4.99E+02  |
| $f_{26}$ | Mean   | 1.33E+03 | 7.20E+03 | 1.25E+03 | 8.59E+03 | 1.70E+03 | 4.74E+03 | 3.15E+03  |
|          | Median | 1.33E+03 | 7.21E+03 | 1.27E+03 | 8.50E+03 | 1.70E+03 | 4.80E+03 | 3.13E+03  |
| $f_{27}$ | Mean   | 5.13E+02 | 1.09E+03 | 5.08E+02 | 5.00E+02 | 4.84E+02 | 5.00E+02 | 5.00E+02  |
|          | Median | 5.14E+02 | 1.09E+03 | 5.07E+02 | 5.00E+02 | 4.85E+02 | 5.00E+02 | 5.00E+02  |
| $f_{28}$ | Mean   | 3.45E+02 | 3.63E+03 | 4.33E+02 | 5.00E+02 | 5.29E+02 | 6.37E+02 | 5.00E+02  |
|          | Median | 3.00E+02 | 3.70E+03 | 4.21E+02 | 5.00E+02 | 5.29E+02 | 5.00E+02 | 5.00E+02  |
| $f_{29}$ | Mean   | 5.15E+02 | 2.63E+03 | 4.86E+02 | 3.58E+03 | 6.01E+02 | 1.94E+03 | 1.59E+03  |
|          | Median | 4.96E+02 | 2.70E+03 | 4.55E+02 | 3.53E+03 | 5.83E+02 | 1.88E+03 | 1.56E+03  |
| $f_{30}$ | Mean   | 3.82E+03 | 2.87E+08 | 9.06E+03 | 7.28E+08 | 2.80E+03 | 1.11E+07 | 1.30E+07  |
|          | Median | 3.39E+03 | 2.73E+08 | 9.18E+03 | 6.54E+08 | 1.90E+03 | 2.85E+06 | 1.38E+07  |

### 3.2 求解精度及分析

(1) Unimodal Functions ( $f_1 \sim f_3$ ): 从实验结果可以看出 EDA-M/D 在 3 个单峰函数上均取得了最好的结果, 此外, PBILc 也取得了较好的结果。由于单峰函数中不存在局部最优解, 种群中的精英个体较之其它个体而言, 距离全局最优解更近。因此, EDA-M/D 依赖精英群体所生成的概率模型使种群具有较快的收敛速度。此外, 多精英个体独立并行采样使得当前种群最优个体更容易得到更新。正是 EDA-M/D 在此类函数中表现出最优性能的原因。

(2) Simple Multimodal Functions ( $f_4 \sim f_{10}$ ): EDA-M/D 对 5 个多峰函数( $f_4 \sim f_8$ )求解结果的 *Mean* 及 *Median* 等指标均取得了最优结果, 表现出最稳定的优化性能, EDA-M/D 在此类函数上具有最优的综合性能。在另外 2 个多峰函数( $f_9, f_{10}$ )上, PBILc 则在上述 2 个指标上得到了最优结果。需要指出的是, 尽管 EDA-M/D 在  $f_9$  上取得的 *Mean* 值要明显劣于 PBILc, 但它们均能在多次运行中搜索到全局最优解。这也说明, EDA-M/D 设定的自适应子代生成策略能有效利用种群全局信息和个体信息, 增加了种群多样性, 从而提高了算法跳出局部最优的能力, 增强算法在多峰函数上的优化性能。

(3) Hybrid Functions ( $f_{11} \sim f_{20}$ ): 从实验结果可以看出, EDA-M/D 在  $f_{11}, f_{12}, f_{13}, f_{14}$  和  $f_{18}$  等 5 个混杂函数的 2 个性能指标上均得到最好的结果, PBILc 则在  $f_{16}$  和  $f_{17}$  上取得了最优的综合性能。此外, EDA-M/D 还在其他 3 个混合函数( $f_{15}, f_{19}$  和  $f_{20}$ )上取得了次优的 *Mean* 值, 一定程度上说明了 EDA-M/D 所采用的 NDL 策略在帮助种群跳出局部最优上具有一定效果。综合而言, EDA-M/D 在面对此类混杂函数时的性能较为稳定, 但在个体优化方面还有提升空间。

(4) Composition Functions ( $f_{21} \sim f_{30}$ ): 在 10 个复合函数上, PBILc 和 EDA-M/D 取得了相对更优的结果, 同时, GA-EDA 也表现出良好的性能。这也

说明, 变异操作在一定程度上有助于 EDA 保持种群多样性, 从而提升其在复杂函数优化中的性能。需要指出的是, 复合函数是性质复杂、较难优化的一类问题, 从实验结果可以看出, 没有任何对比算法在此类函数上能搜索到全局最优解。而且从求解结果的绝对值上来看, 算法间的性能差异也不明显。

这也说明我们需要从种群历史经验中获取有益的信息, 构建符合当前适应值景观的概率模型来指导种群的后续搜索, 以提升算法在复杂多峰函数中的综合性能。

### 3.3 t-检验结果

为了判断不同算法在求解性能上是否存在显著性差异, 本文利用 t-检验对算法的求解精度进行了显著性检验, 显著性水平均为  $\alpha=0.05$ 。检验结果如表 5 所示。表中 Better, Same 和 Worse 分别表示 EDA-M/D 比相应的对比算法在 t-检验结果中具有显著更优、相同和显著更劣的函数个数。*B-W* 表示 EDA-M/D 显著性优于对比算法的函数个数与其显著性劣于相应算法的函数个数的差值, 该值越大说明相应算法在 t-检验中体现的性能越差。

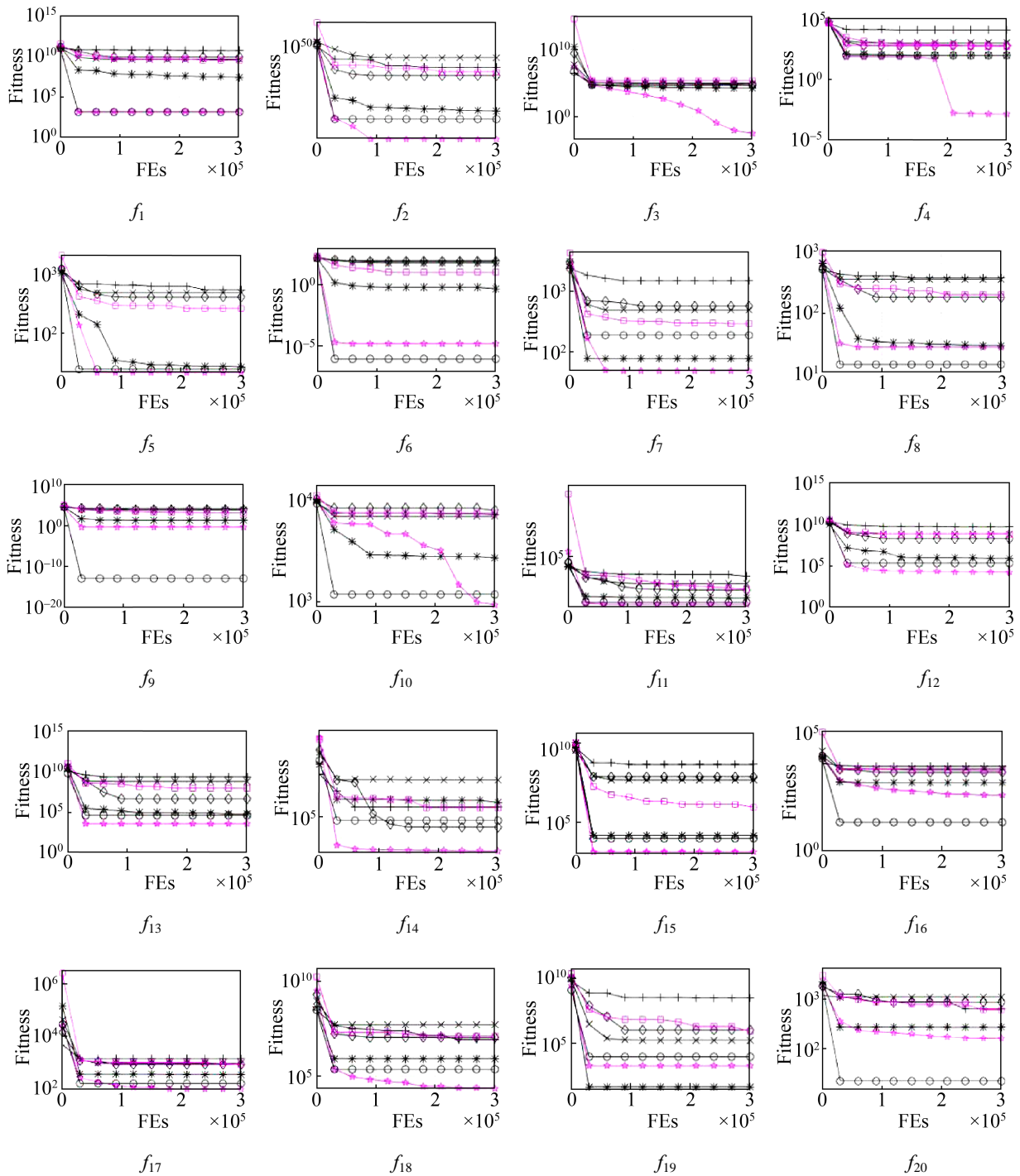
从表 5“*B-W*”指标来看, EDA-M/D 在测试集上具有最优的综合性能, PBILc 则取得了次优的表现。需要指出的是, 尽管 EDA-M/D 在单峰函数( $f_1 \sim f_3$ )、简单多峰函数( $f_4 \sim f_{10}$ )和混杂函数( $f_{11} \sim f_{20}$ )上表现优于 PBILc, 但后者在复合函数( $f_{21} \sim f_{30}$ )上表现出更优的性能。我们认为在 2 种子代生成模式的自适应策略及劣势个体的差分学习上有待进一步改善。

### 3.4 收敛速度对比

为了更直观地呈现算法的收敛过程, 本文对上述算法在 CEC2017 测试集上进行了收敛性能的对比, 结果如图 3 所示。

表 5 EDA-M/D 与其它 7 个算法的 t-检验结果  
Tab. 5 T-test results between EDA-M/D and other 7 peer algorithms

| 对比结果   | UMDAc | PBILc | EDA-B/O | GA-EDA | EDA-PSO | AIL-DE/EDA |
|--------|-------|-------|---------|--------|---------|------------|
| Better | 30    | 12    | 28      | 24     | 28      | 29         |
| Same   | 0     | 8     | 1       | 2      | 1       | 0          |
| Worst  | 0     | 10    | 1       | 4      | 1       | 1          |
| B-W    | 30    | 2     | 27      | 20     | 27      | 28         |



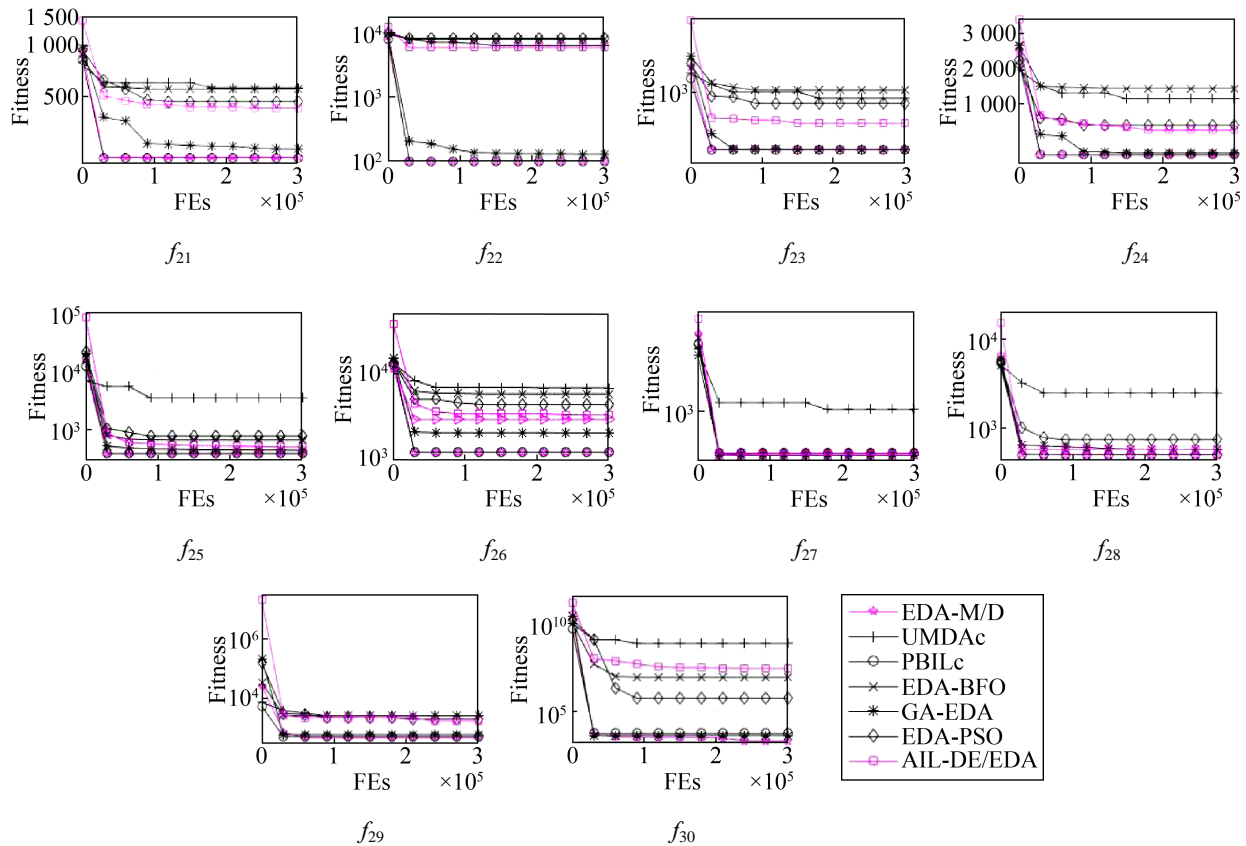


图 3 对比算法在 CEC2017 测试集上的收敛过程对比  
Fig. 3 Comparison results of 7 peer algorithms on CEC2017 test suite

实验结果表明 EDA-M/D 和 PBILc 在 3 个单峰函数上的收敛速度较快。其中, EDA-M/D 在进化初期的收敛速度要略逊于 PBILc, 而在进化后期, EDA-M/D 则表现出更优的收敛特性。这是由于在进化初期, 精英个体分布范围较广, 在没有局部最优解的干扰下, EDA-M/D 中多精英采样策略的搜索效率不及 PBILc 的效率。而在进化后期, 由于种群逐步收敛, 其精英个体分布也较为集中, 此时多精英并行采样的优势则显现出来, 有效地提升了 EDA-M/D 的收敛速度。从各算法在  $f_3$  上的收敛曲线来看, EDA-M/D 要显著性地优于其它算法。这也说明 EDA-M/D 中劣势个体基因修复策略通过保留良好“建筑块”, 在一定程度上缓解了  $f_3$  中多变量耦合的难题。在 7 个简单多峰函数上, EDA-M/D 在  $f_4, f_5, f_6, f_7$  和  $f_9$  等 5 个简单多峰函数上的收敛性能要明显优于其它对比算法。此外, EDA-M/D 在  $f_8$  上的性能也仅略差于 PBILc, 在  $f_{10}$  上,

EDA-M/D 在整个进化过程中保持着较平稳的收敛性能, 并在进化末期取得了仅次于 PBILc 的求解精度。这说明 EDA-M/D 较好地平衡了全局搜索于收敛速度之间的矛盾。在其它 20 个复杂函数上(包括混杂函数和复合函数), 尽管各算法在此类函数上的求解精度从绝对值的角度来看差别不大, 但各算法的收敛速度存在一些明显的差异。其中, EDA-M/D 在 14 个函数上取得了最优或近于最优的收敛速度, 同时, PBILc 也很多函数上也取得了较好的性能表现。尽管这 2 个算法在此类复杂函数上的性能相对较优, 但从其收敛曲线来看, 算法在大部分函数的优化后期均未得到明显改善, 即存在“早熟”收敛的现象。

## 4 结论

为了在有效利用群体统计信息的同时, 融入个体的并行搜索机制, 本文提出了一种基于多精英采

样和劣势个体差分搜索的分布估计算法 EDA-M/D。通过与其它多个 EDAs 在 CEC2017 测试集上的对比实验表明, EDA-M/D 在算法稳定性和最优搜索能力方面均表现出明显的优势。同时, 本文还对 EDA-M/D 涉及的几个新引入的参数和策略进行了性能对比与分析。结果表明, 多精英并行采样策略和劣势个体的基因修复策略在复杂多峰函数上有利于保持种群多样性和种群收敛速度。同时, 劣势个体的差分学习也能较好地引导种群跳出局部最优。

需要注意的是, 劣势个体隐含的有益信息的有效利用及其差分学习的具体策略及时机的选择依然是一个难点。本文作者将在后续工作中对该问题进行更深入的研究。

### 参考文献:

- [1] Larrañaga P, Lozano JA. Estimation of distribution algorithms[J]. *Genetic Algorithms & Evolutionary Computation (S1063-6560)*, 2001, 64(5): 454-468.
- [2] 周树德, 孙增圻. 分布估计算法综述[J]. *自动化学报*, 2007, 33(2): 113-124.  
Zhou Shude, Sun Zengqi. A survey estimation of distribution algorithms[J]. *Acta Automatica Sinica*, 2007, 33(2): 113-124.
- [3] 任志刚, 梁永胜, 张爱民, 等. 基于一般二阶混合矩的高斯分布估计算法[J]. *自动化学报*, 2018, 44(4): 635-645.  
Ren Zhigang, Liang Yongsheng, Zhang Aimin, et al. A gaussian estimation of distribution algorithm using general second-order mixed moment[J]. *Acta Automatica Sinica*, 2018, 44(4): 635-645.
- [4] Zhou A M, Sun J Y, Zhang Q F. An estimation of distribution algorithm with cheap and expensive local search methods[J]. *IEEE Transactions on Evolutionary Computation (S1089-778X)*, 2015, 19(6): 807-822.
- [5] Wang J, Tang K, Lozano JA, et al. Estimation of the distribution algorithm with a stochastic local search for uncertain capacitated arc routing problems[J]. *IEEE Transactions on Evolutionary Computation (S1089-778X)*, 2016, 20(1): 96-109.
- [6] 陈伟能, 杨强. 基于概率分布的多峰演化算法[J]. *计算机研究与发展*, 2017, 54(6): 1185-1197.  
Chen Weineng, Yang Qiang. Probability distribution based evolutionary computation algorithms for multimodal optimization[J]. *Journal of Computer Research and Development*, 2017, 54(6): 1185-1197.
- [7] Baluja S. Population-Based Incremental Learning. A method for integrating genetic search based function optimization and competitive learning[R]. Technical Report CMU-CS-94-163, Pittsburgh, PA: Carnegie Mellon University, 1994.
- [8] Sebag M, Ducoulombier A. Extending population-based incremental learning to continuous domains by learning and simulation of Gaussian networks[C]. *Proc. of the 5th Conference on Parallel Problem Solving from Nature*. Amsterdam, Netherlands: Springer, 1998: 418-427.
- [9] Mühlenbein H, Paass G. From recombination of genes to the estimation of distributions[C]. *Proc. of the 4th Parallel Problem Solving from Nature*. Berlin, Germany: Springer, 1996: 178-187.
- [10] Mühlenbein H. The equation for response to selection and its use for prediction[J]. *Evolutionary Computation (S1063-6560)*, 2014, 5(3): 303-346.
- [11] Larranaga P, Etxeberria R, Lozano JA, Pena JM. Optimization in continuous domains by learning and simulation of Gaussian networks[C]. *Proc. of the 2000 Genetic and Evolutionary Computation Conference Workshop Program*. Nevada, USA: ACM, 2000: 201-204.
- [12] Harik G R, Lobo F G, Goldberg D E. The compact genetic algorithm[J]. *IEEE Transactions on Evolutionary Computation (S1089-778X)*, 1999, 3(4): 287-297.
- [13] Krejca M S, Witt C. Lower bounds on the run time of the univariate marginal distribution algorithm on onemax[C]. *Proc. of Acm/sigevo Conference on Foundations of Genetic Algorithms*. Copenhagen, Denmark: ACM, 2017: 65-79.
- [14] Baluja S, Davies S. Using optimal dependency-trees for combinatorial optimization: learning the structure of the search space[C]. *Proc. of International Conference on Machine Learning*. San Francisco, USA: ACM, 1997: 30-38.
- [15] Henry E C, Rodrigo C B, Marcio P B. Estimation of distribution algorithms for decision-tree induction[C]. *Proc. of IEEE Congress on Evolutionary Computation*. San Sebastian, Spain: IEEE, 2017: 2022-2029.
- [16] 邱玲, 高尚, 曹存根. 改进的正态分布的分布估计算法[J]. *计算机科学*, 2015, 42(8): 32-35.  
Qiu Ling, Gao Shang, Cao Cungen. Improved estimation of distribution algorithms based on normal distribution[J]. *Computer Science*, 2015, 42(8): 32-35.



- [17] 刘小龙, 李荣钧, 杨萍. 基于高斯分布估计的细菌觅食优化算法[J]. 控制与决策, 2011, 26(8): 1233-1238.
- Liu Xiaolong, Li Rongjun, Yang Ping. Bacterial foraging optimization algorithm based on estimation of distribution[J]. Control and Decision, 2011, 26(8): 1233-1238.
- [18] El-Abd M. Preventing premature convergence in a PSO and EDA hybrid[C]. Proc. of IEEE Congress on Evolutionary Computation. Trondheim, Norway: IEEE, 2009: 3060-3066.
- [19] Shao W, Shang L, Ma L, et al. Hybrid Differential Evolution/Estimation of Distribution Algorithm Based on Adaptive Incremental Learning[J]. Journal of Computational Information Systems (S0887-4417), 2014, 10(12): 5355-5364.
- [20] Peña J M, Robles V, Larrañaga P, et al. GA-EDA: Hybrid evolutionary algorithm using Genetic and estimation of distribution algorithms[C]. Proc. Of the 17th International Conference on Innovations in Applied Artificial Intelligence. Ottawa, Canada: ACM, 2004: 361-371.
- [21] Awad N H, Ali M Z, Liang J J, et al. Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective real-parameter numerical optimization[R]. Nanyang Technological University, Singapore, Technical Report, 2016.