基于耿贝尔采样的差分进化算法

张合[1a],王川[1b],黎建宇[2]

(1.河南师范大学 a.教育学部;b.软件学院,河南 新乡 453007;2.南开大学 人工智能学院,天津 300350)

摘 要:针对差分进化算法在求解复杂优化问题中面临的搜索多样性不足、易陷入局部最优等问题,提出了一种融合耿贝尔采样机制的差分进化算法.该算法引入了两种新型变异策略:一是基于耿贝尔采样学习的变异策略,通过对高质量个体进行耿贝尔采样,提升个体生成的质量;二是基于耿贝尔采样的精英变异策略,结合耿贝尔扰动机制对精英个体进行局部搜索,增强算法的局部开发能力.两种策略协同作用,能够有效提高种群的多样性和搜索精度.在大量测试集函数和大规模定日镜场应用问题上对所提算法与多种主流差分进化算法变体进行了对比.实验结果表明,所提算法在大多数测试问题上表现优越,兼具较好的全局搜索能力与收敛性能,具有较强的稳定性与适应性.该研究为复杂优化问题的智能求解提供了一种有效的新方法.

关键词:差分进化算法;耿贝尔采样;变异策略;全局优化;进化计算

中图分类号:TP278;TP391 **文献标志码**:A **文章编号**:1000-2367(2026)02-0038-08

随着工业领域的智能化发展,众多复杂的优化问题也随之涌现[1].这些问题利用传统的数学方法常常难以得到有效求解或在无法可接受的时间内获得高质量的解决方案.受益于其不依赖于数学模型的性质及算法简易性,受启发于生物智能演化的差分进化算法(differential evolution,DE)[2]已被广泛用于求解众多的复杂优化问题.自提出以来,DE 算法一直受到研究人员的广泛关注[3].在该算法中,不同个体之间的差分向量用于生成新个体,以辅助其能快速找到高质量候选解.与其他经典的进化计算方法相比,DE 算法具有多项优势,包括简单的种群结构、直观的搜索方法以及快速的收敛效率[4].

近年来,研究者提出了许多 DE 变体.例如,文献[5]提出了分数阶差算子,以提高 DE 的探索能力.由于在求解不同问题或在求解同一个问题的不同阶段,可能需要不同的算子和参数,很多学者也开始研究自适应的 DE.文献[6]提出了基于知识引导的自适应 DE,期望提高其通用性.文献[7]提出基于云计算和多种群的自适应 DE,将个体资源从表现不佳的算子种群迁移到表现更好的算子种群中,以增加计算效率.文献[8]提出合并和分解算子,以自适应分配个体和计算资源.文献[9]同样基于多种群,提出了自适应资源分配的分布式 DE(distributed DE with adaptive resources allocation,DDE-ARA),可通过计算资源自适应分配,提高算法的求解效率.此外,在计算效率方面,文献[10]提出了基于矩阵的 DE(matrix-based DE,MDE),通过矩阵算子加快了算法的搜索能力.文献[11]提出低维空间建模的 DE(low-dimensional space modeling-based DE,LSMDE),通过降维提高算法效率.除了单纯的算法改进,很多研究也对众多复杂优化问题进行了 DE 的扩展研究,包括通信网络优化[12]和国防军工应用[13]等.

尽管已有不少 DE 的改进研究,现有 DE 算法在求解复杂问题时仍面临搜索多样性差、容易落入局部最

收稿日期:2025-07-11;**修回日期**:2025-09-10.

基金项目:国家自然科学基金(62406152);中央高校基本科研业务费专项资金资助(078-63251088;078-63253248).

作者简介:张合(1984—),男,河南南阳人,河南师范大学博士研究生,研究方向为进化计算.

通信作者:王川(1976—),男,河南新乡人,河南师范大学副教授,E-mail:wangch@htu.edu.cn;黎建宇,E-mail:jianyuli@nankai.edu.cn.

引用本文:张合,王川,黎建宇.基于耿贝尔采样的差分进化算法[J].河南师范大学学报(自然科学版),2026,54(2):38-45.(Zhang He,Wang Chuan,Li Jianyu.Gumbel sampling-based differential evolution[J].Journal of Henan Normal University(Natural Science Edition),2026,54(2):38-45.DOI:10.16366/j.cnki.1000-2367.2025.07.11.0001.)

优的性能瓶颈.针对这些问题，本文提出基于耿贝尔分布采样的 DE 算法(gumbel sampling-based DE，GS-DE).GSDE 通过借助耿贝尔采样改善变异算子产生个体的多样性和质量，从而提升整个种群的演化效率，帮助算法避开局部最优，更高效地求解到最优解.所提出的算法和方法在多个常用测试函数及大规模定日镜场测试案例上进行了实验，并与现有的前沿 DE 算法进行了对比.相关结果显示了本文所提算法的高效性.

1　差分进化算法

在自然界中，生物的进化遵循“物竞天择，适者生存”的原则.作为进化计算的一个分支，DE 同样模拟了自然界中生物的生存和繁衍过程，不断优化种群并最终得到最优解.在 DE 中，一个个体对应当前需要求解的问题的一个潜在解.DE 算法通过顺序执行以下 3 个步骤并进行迭代，从而寻找最优解.

首先，算法随机地初始化 N 个个体，作为初始种群，并测试每个个体的适应值.初始化过程一般基于以下公式：

$$X_{i,j}=LB_j+\mathrm{rand}(0,1)\cdot(UB_j-LB_j),\tag{1}$$

其中，$X_{i,j}$ 是第 i 个个体的第 j 维度取值，UB_j 和 LB_j 分别是第 j 维度的搜索空间上界和下界取值. 可见，每个初始个体就是一个随机产生的解，适应值对应优化问题的候选解的质量.

然后，算法通过变异算子产生新的目标向量 $\boldsymbol{V}$.常见的变异算子包括：

1)DE/rand/1

$$V_{i,j}=X_{r1,j}+F\cdot(X_{r2,j}-X_{r3,j}),\tag{2}$$

其中，$V_{i,j}$ 是第 i 个目标向量的第 j 维度值，$X_{r1,j}$，$X_{r2,j}$ 和 $X_{r3,j}$ 是从种群中随机选出的 3 个个体，$r1$，$r2$，$r3$ 是对应的互不相同的随机索引.F 是控制差分向量影响的缩放因子.

2)DE/best/1

$$V_{i,j}=X_{\mathrm{best},j}+F\cdot(X_{r2,j}-X_{r3,j}),\tag{3}$$

其中，$X_{\mathrm{best},j}$ 是当前算法种群中的适应值最好的个体的第 j 维度取值.

在通过变异算子得到目标向量 $\boldsymbol{V}$ 后，算法会基于 X 和 $\boldsymbol{V}$ 执行交叉算子，产生试验向量 $\boldsymbol{U}$，即新的候选个体.交叉算子的公式如下：

$$U_{i,j}=\begin{cases}V_{i,j}，若\ \mathrm{rand}(0,1)\le CR\ 或\ j=j_{\mathrm{rand}},\\ X_{i,j}，否则,\end{cases}\tag{4}$$

其中，$U_{i,j}$ 是第 i 个试验向量的第 j 维度取值，CR 是变异概率，用于控制 $\boldsymbol{U}_i$ 中继承原来 $\boldsymbol{X}_i$ 个体的比例，j_{rand} 是一个随机维度索引，以保证 $\boldsymbol{U}_i$ 中至少有一维的取值来源于 $V_{i,j}$.

通过交叉算子得到的新个体会进行适应值评估，然后执行选择算子，以确定新一代种群中的个体.假设要求解的优化问题是最小化问题，解的适应值越小越好，选择算子可以用以下公式表示

$$\boldsymbol{X}_i=\begin{cases}\boldsymbol{U}_i，若\ f(\boldsymbol{U}_i)\le f(\boldsymbol{X}_i),\\ \boldsymbol{X}_i，否则,\end{cases}\tag{5}$$

其中，$f(\boldsymbol{X}_i)$ 和 $f(\boldsymbol{U}_i)$ 分别是 $\boldsymbol{X}_i$ 和 $\boldsymbol{U}_i$ 的适应值. DE 不断迭代地执行变异、交叉、评估和选择算子，直到达到算法停止条件.当算法达到停止条件，算法输出当前找到的最优解.至此，完整算法结束.

2　基于耿贝尔采样学习的差分进化算法

2.1　基于耿贝尔采样学习的变异策略

GLM 旨在通过耿贝尔分布，将离散的个体分布转换为连续的分布，从而让种群个体更好地选择表现优秀的个体进行学习.GSM 的过程简介如下.首先，每个个体，例如 $\boldsymbol{X}_i$，找出种群中比它适应值更好的个体，并记录于矩阵 $\boldsymbol{I}$.其中，$I_{i,j}$ 是第 i 行第 j 列的取值，其取值定义如下：

$$I_{i,j}=\begin{cases}1，若\ f(\boldsymbol{X}_j)<f(\boldsymbol{X}_i),\\ 0，否则.\end{cases}\tag{6}$$

假设整个种群一共有 N 个个体，则种群第 i 个体采样种群第 j 个体进行学习的对应权重为

$$q_{i,j}=I_{i,j}\cdot\frac{f(\boldsymbol{X}_j)}{\sum_{k=1}^{N}f(X_k)}.\tag{7}$$

为了增加采样多样性,使用耿贝尔分布调整采样权重,调整后的采样权重 $p_{i,j}$ 计算方式如下:

$$p_{i,j}=\lg(q_{i,j})+G_u(0,1),\tag{8}$$

其中,$G_u(0,1)$ 代表从标准的耿贝尔分布中采样的随机数.基于 p 的值,每个个体选择最大权重的其他个体进行学习,选择过程可表示为

$$z(i)=\max_j\{p_{i,j}\}.\tag{9}$$

基于选择得到的第 $z(i)$个体,个体 i 使用如下变异公式向第 $z(i)$个体进行学习得到 $\boldsymbol{V}_i$:

$$V_{i,j}=X_{i,j}+F\cdot(X_{z(i),j}-X_{r1,j}),\tag{10}$$

其中,$r1$ 是随机个体索引.

2.2 基于耿贝尔采样的精英变异策略

为了增加搜索的多样性,避免算法落入局部最优,本文提出 GEM 策略.针对种群的精英解,即当前种群的最优解,使用如下变异公式,基于正态分布生成目标向量.

$$V_{i,j}=G_a(\mu_{i,j},\sigma_{i,j}),\tag{11}$$

其中,$G_a(\mu_{i,j},\sigma_{i,j})$ 代表以 $\mu_{i,j}$ 为均值,$\sigma_{i,j}$ 为方差的正态分布.$\mu_{i,j}$ 的计算如下:

$$\mu_{i,j}=\frac{1}{2}(X_{z(i),j}+X_{i,j}),\tag{12}$$

而 $\sigma_{i,j}$ 的计算如下

$$\sigma_{i,j}=\frac{1}{2}(X_{z(i),j}-X_{i,j}).\tag{13}$$

2.3 基于耿贝尔采样的完整差分进化算法流程

GSDE 伪代码如算法 1 所示,并介绍如下.首先,算法执行初始化操作,随机采样一个初始种群,包含 N 个个体,每个个体表示一个潜在解(第 1 行).随后,执行适应值评估操作,评估种群中的每个个体的适应值,用于衡量其在当前问题上的优劣(第 2 行).在主循环中(第 3～14 行),算法不断迭代优化,直到满足终止条件.在每一代中,算法依次对种群中的每一个个体进行操作(第 4 行).对于当前种群中表现最优的个体,采用 GEM(第 6 行,公式),引入一定扰动以提升对局部最优的搜索精度;对于其他个体,则采用 GLM 策略(第 8 行,式(10)),以保持搜索的多样性和全局探索能力.之后,算法对每个变异向量 $\boldsymbol{V}_i$ 和与其对应的父个体 $\boldsymbol{X}_i$ 执行交叉操作,生成试验个体 $\boldsymbol{U}_i$(第 10 行,公式),并评估其适应值(第 11 行).通过选择操作,算法从 U_i 和 $\boldsymbol{X}_i$ 中保留适应度更优的个体进入下一代种群(第 13 行,式(5)),以推动解的不断演化和优化.最终,如果 GSDE 判断满足终止条件,则 GSDE 将输出目前搜索到的最优个体(第 15 行).该算法结合了基于耿贝尔分布采样的两种变异策略与经典差分进化算法,在保持搜索多样性的同时,有效提升了局部收敛性能.

算法 1 基于耿贝尔采样的差分进化算法

```
1:初始化种群;//式(1)
2:对初始个体进行适应值评估;
3:While 算法未达到终止条件 Do
4:    For i=1:N Do
5:        If X_i 是当前最优个体 Then
6:            根据式(11)执行变异操作,得到 V_i;
7:        Else
8:            根据式(10)执行变异操作,得到 V_i;
9:        End If
10:       对 V_i 和 X_i 进行交叉操作生成新个体 U_i;//式(4)
11:       评估新个体的适应值;
12:   End
13:   通过选择操作选择更好的个体形成新种群;//式(5)
14:End
15:将当前最优个体作为结果进行输出,并停止算法运行.
```

3 实验结果

3.1 实验设置

本文使用 12 个广泛使用的可扩展基准问题以测试所提出 GSDE 算法的求解性能.这 12 个基准问题在

表 1 中被标记为 f_1 至 f_{12}，它们具有不同的问题特征，并且可以设置不同的维度以进行大规模优化研究.因此，这 12 个问题有助于更好地观察所提议的 GSDE 在不同情况下的行为.对于实验环境，所有实验均在配备 Intel(R)Core(TM)i5-7400 的 CPU 和 8.00 GB RAM 的计算机集群上进行.

本部分将 GSDE 与 3 种前沿 DE 算法在 12 个广泛使用的可扩展基准问题上进行比较.这 12 个问题的维度设置为 $D=1\ 000$.被比较的算法包括：MDE[10]，DDE-ARA[9] 和 LSMDE[11].对于 GSDE，每次变异和交叉算子执行种，F 的取值采用均值为 0.7，标准差为 0.5 的正态分布随机数，CR 的取值使用均值为 0.5、标准差为 0.5 的正态分布随机数.此外，对于每个算法在每个问题上的运行，每次独立运行中适应度评估的最大次数(即 Max FEs)设置为 5×10^5.每个算法在每个问题上运行 25 次，并使用平均结果进行比较.此外，本文采用 Wilcoxon 秩和检验(显著性水平 $\alpha=0.05$)对 25 次运行中的算法结果进行统计比较，其中表格中采用 3 个符号"＋"、"≈"和"－"分别表示所提 GSDE 算法相较于其他算法具有显著优于、与之相当或显著劣于的性能.

表 1　本文采用的 12 个测试函数

Tab. 1　The 12 test functions used in this paper

测试问题	描述	函数特点	搜索范围
f_1	Sphere function	单峰	$[-100,100]^D$
f_2	Sch's 2.22 function	单峰	$[-5,5]^D$
f_3	Quadric function	单峰	$[-100,100]^D$
f_4	Sch's 2.21 function	单峰	$[-100,100]^D$
f_5	Step function	单峰	$[-100,100]^D$
f_6	Quadric noise function	单峰，噪声	$[-1.28,1.28]^D$
f_7	Rosenbrock function	多峰	$[-10,10]^D$
f_8	Schwefel function	多峰	$[-500,500]^D$
f_9	Rastrigin function	多峰	$[-5.12,5.12]^D$
f_{10}	Noncontinuous Rastrigin function	多峰	$[-5.12,5.12]^D$
f_{11}	Ackley function	多峰	$[-32,32]^D$
f_{12}	Griewank function	多峰	$[-600,600]^D$

3.2　与前沿算法的比较结果

为了测试本文所提出的 GSDE 算法，本文将 GSDE 与现有的 3 种先进 DE 变种 MDE，DDE-ARA 和 LSMDE 进行对比.均值及标准差结果如表 2 所示.

表 2　GSDE 与前沿 DE 算法的均值和标准差结果

Tab. 2　The mean and standard deviation results of GSDE and state-of-the-art DEs

问题	GSDE	MDE	DDE-ARA	LSMDE
f_1	2.45E＋04±4.50E＋03	3.22E＋05±4.20E＋04(＋)	6.84E＋04±8.25E＋03(＋)	8.41E＋04±1.19E＋04(＋)
f_2	1.77E＋02±1.11E＋01	7.62E＋02±5.32E＋01(＋)	2.80E＋02±2.15E＋01(＋)	3.78E＋02±2.49E＋01(＋)
f_3	2.34E＋06±7.39E＋05	2.61E＋06±9.06E＋04(≈)	6.12E＋05±7.51E＋04(－)	4.68E＋04±2.36E＋03(－)
f_4	3.71E＋01±2.60E＋00	4.54E＋01±1.39E＋00(＋)	3.91E＋01±3.34E＋00(≈)	2.05E＋01±1.05E＋00(－)
f_5	4.55E＋04±3.70E＋03	3.40E＋05±3.93E＋04(＋)	8.35E＋04±7.77E＋03(＋)	8.10E＋04±1.27E＋04(＋)
f_6	6.41E＋02±7.34E＋02	1.50E＋04±3.56E＋03(＋)	8.63E＋02±3.61E＋02(≈)	1.78E-04±1.80E-04(－)
f_7	8.29E＋04±1.71E＋04	2.73E＋06±4.62E＋05(＋)	2.80E＋05±7.02E＋04(＋)	2.50E＋05±5.31E＋04(＋)
f_8	2.20E＋05±8.21E＋03	3.43E＋05±2.75E＋03(＋)	2.52E＋05±8.04E＋03(＋)	3.62E＋05±1.38E＋04(＋)
f_9	2.62E＋03±2.06E＋02	8.99E＋03±2.12E＋02(＋)	3.42E＋03±1.78E＋02(＋)	6.94E＋03±4.13E＋02(＋)
f_{10}	4.96E＋03±2.56E＋02	8.78E＋03±2.20E＋02(＋)	4.07E＋03±1.94E＋02(－)	6.31E＋03±3.58E＋02(＋)
f_{11}	1.20E＋01±3.33E-01	2.00E＋01±8.35E-05(＋)	1.11E＋01±2.14E-01(－)	1.05E＋01±3.25E-01(－)
f_{12}	2.11E＋02±2.77E＋01	2.87E＋03±3.23E＋02(＋)	6.31E＋02±1.02E＋02(＋)	7.70E＋02±7.05E＋01(＋)
＋/≈/－	NA	11/1/0	7/2/3	8/0/4

从表 2 可以看出,GSDE 在多数测试函数上均优于现有的 3 种主流差分进化算法(MDE,DDE-ARA 和 LSMDE),体现了较强的全局搜索能力与局部收敛性能.统计结果显示,GSDE 在 12 个测试问题中,相较于 MDE 有 11 个问题表现更优,1 个相当,0 个劣于对方;相较于 DDE-ARA 有 7 个问题更优,2 个相当,3 个劣势;相较于 LSMDE 则在 8 个问题上取得更优结果,仅在 4 个问题上略逊一筹.在函数 f_1,f_2,f_5,f_7,f_8 和 f_9 等高维、多峰或具有复杂搜索空间的测试问题上,GSDE 均明显优于其他算法.例如,在函数 f_1上,GSDE 的平均误差为 2.45×10^4,远低于 MDE,DDE-ARA 和 LSMDE 的结果,表明其在处理大规模复杂优化问题时具有更强的稳定性和收敛速度.GSDE 通过引入基于耿贝尔分布采样的全局型变异策略(GLM)和针对精英解的变异策略(GEM),有效增强了算法对搜索方向和局部区域的双重寻优能力,保证了算法在不同类型目标函数上的鲁棒性与稳定性.

综上所述,GSDE 在大多数测试问题中展现出优越的优化性能,在解决具有复杂结构或多峰特性的优化问题时具有明显优势,显示出良好的适应性与推广潜力.

3.3 在 IEEE CEC 2025 竞赛测试集的测试结果

为了全面评估本文提出的 GSDE 算法的综合性能,本节将其与 MDE,DDE-ARA 及 LSMDE 等当前表现优异的多策略差分进化算法,在 IEEE CEC 2025 竞赛测试集[14]的 29 个基准函数上进行了对比.所有算法均在相同实验设置下运行:最大函数评价次数为 3×10^5,独立运行 50 次,记录结果的均值和标准差.实验结果如附录表 S1 所示.

从整体结果看,GSDE 在 CEC2025 测试集上表现出优秀的综合性能.在与 MDE 的对比中,GSDE 在 22 个问题上表现更优,7 个问题上略逊;与 DDE-ARA 相比,GSDE 在 23 个问题上取得更好的结果,仅在 1 个问题上表现不佳;与 LSMDE 相比,GSDE 在 21 个问题上优势明显,7 个问题上稍差.这一结果表明 GSDE 在绝大多数测试问题上均具有优越的收敛精度和稳定性.尤其在函数特征复杂的优化问题(如 F_9,F_{20},F_{22},F_{23},F_{25}等)中,GSDE 显著优于对比算法,说明其耿贝尔采样引导的学习机制能够有效协调全局探索与局部开发,避免早熟收敛,并提升对复杂地形函数的适应能力.因此,在多峰及混合特性函数上,GSDE 凭借其良好的跳出局部极值能力,更好获得更优解.

综上所述,GSDE 算法在 IEEE CEC 2025 测试集上的综合表现优于当前主流 DE 变体,验证了其耿贝尔采样机制在提升收敛性、稳定性与泛化能力方面的有效性,尤其适用于复杂多峰及混合特征的优化问题.

3.4 对所提方法的消融实验分析

为了深入调研本文所提出的 GLM 和 GEM 机制的具体作用,本节将 GSDE 算法与去掉 GLM 或 GEM 的变种算法进行比较.为方便阐述,本节将去掉 GLM 和 GEM 策略的 GSDE 分别命名为 GLDE-w/o-GLM 和 GLDE-w/o-GEM.

不同 GSDE 算法的比较结果如表 3 所示.从表 3 中可以观察到,去除 GLM 或 GEM 策略后,GSDE 算法的性能在多个测试函数上出现不同程度的波动,进一步验证了这两种策略对算法整体性能的关键作用.为便于分析,将完整 GSDE 算法作为基准版本,并以“+”“≈”“?”分别表示变种算法在某测试函数上相较于基准版本性能更优、相当或更差的情况.由表 3 可知,GLDE-w/o-GLM 在 12 个测试函数中有 7 项性能下降(“+”),3 项持平,2 项有所提升(“-”),说明去除 GLM 机制后,算法整体性能呈明显下滑趋势.特别是在第 1,4,6,7,11 和 12 个函数问题上,该变种版本的最优解均明显劣于原始 GSDE,表明 GLM 在提升搜索效率、避免早熟收敛方面具有积极作用.GLM 通过对当前个体进行耿贝尔随机采样的引导学习,有效提高了 GSDE 对带有复杂搜索空间问题的全局寻优能力.

相较之下,GLDE-w/o-GEM 在 12 个测试函数中,有 5 项表现更差,3 项相当,4 项优于基准算法,整体表现略逊于原始算法但优于 GLDE-w/o-GLM.在第 1,6,10 和 11 个函数问题上,该算法的性能明显下降,其中在复杂函数 f_6上下降幅度尤为显著,说明 GEM 在提升算法收敛精度方面发挥了关键作用.该机制主要作用于当前最优个体,通过耿贝尔分布采样引导的轻扰动,增强了算法对最优解邻域的搜索能力,尤其在搜索后期对精细化解的获取具有重要意义.

进一步结合函数特性分析可知,GLM 采用的耿贝尔随机采样策略对含噪声函数(如 f_6)具有显著的抗干扰能力.耿贝尔采样通过对当前个体施加结构化的随机扰动,在增强全局探索的同时有效平抑了噪声引起

的局部误导，从而在复杂搜索空间中保持良好的寻优稳定性.

相比之下，GLDE-w/o-GEM 在 f_6 函数上表现明显下滑，尤其在 f_{10} 这类非连续函数中性能回落较为显著.这表明 GEM 机制在提高算法收敛精度和局部开发能力方面具有关键作用.GEM 通过对当前最优个体施加基于耿贝尔采样的轻量扰动，增强了对最优解邻域的精细搜索能力，尤其在收敛后期对解的质量提升至关重要.然而，f_{10} 作为非连续函数，其搜索空间存在多处跃变和间断区，仅依赖 GEM 所提供的局部扰动难以实现跨区域全局探索.该类问题需更强力的全局搜索策略以跳出潜在局部最优，这也解释了 GEM 在该函数上的作用局限性.

综上可知，GLM 和 GEM 作为 GSDE 的两个核心子模块，分别从全局探索和局部开发两个维度提升了算法性能.去除 GLM 会严重削弱算法的全局搜索能力，而去除 GEM 则影响算法在收敛阶段对解的精细优化能力.两者协同作用，构成了 GSDE 算法良好优化性能的重要支撑.

表 3　采用不同变异算子的变种 GSDE 算法结果均值及标准差

Tab. 3　The mean and standard deviation results of GSDE variants with different mutations

问题	GSDE	GLDE-w/o-GLM	GLDE-w/o-GEM
f_1	2.45E+04 ± 4.50E+03	5.45E+04 ± 1.18E+04(+)	3.03E+04 ± 2.09E+03(+)
f_2	1.77E+02 ± 1.11E+01	1.85E+02 ± 1.89E+01(≈)	1.66E+02 ± 1.05E+01(−)
f_3	2.34E+06 ± 7.39E+05	2.56E+06 ± 7.29E+05(≈)	1.16E+05 ± 9.50E+03(−)
f_4	3.71E+01 ± 2.60E+00	5.72E+01 ± 4.10E+00(+)	3.74E+01 ± 1.43E+00(≈)
f_5	4.55E+04 ± 3.70E+03	1.15E+05 ± 1.96E+04(+)	4.45E+04 ± 4.54E+03(≈)
f_6	6.41E+02 ± 7.34E+02	2.19E+04 ± 5.67E+03(+)	3.24E+03 ± 2.20E+03(+)
f_7	8.29E+04 ± 1.71E+04	4.89E+05 ± 2.06E+05(+)	6.90E+04 ± 1.36E+04(−)
f_8	2.20E+05 ± 8.21E+03	1.57E+05 ± 4.52E+03(−)	2.57E+05 ± 8.37E+03(+)
f_9	2.62E+03 ± 2.06E+02	2.59E+03 ± 2.90E+02(≈)	2.58E+03 ± 2.23E+02(≈)
f_{10}	4.96E+03 ± 2.56E+02	3.48E+03 ± 2.87E+02(−)	5.36E+03 ± 3.57E+02(+)
f_{11}	1.20E+01 ± 3.33E-01	1.97E+01 ± 1.02E+00(+)	1.86E+01 ± 3.50E+00(+)
f_{12}	2.11E+02 ± 2.77E+01	4.88E+02 ± 8.80E+01(+)	1.88E+02 ± 3.02E+01(−)
+/≈/−	NA	7/3/2	5/3/4

3.5　种群规模参数对算法的性能影响

为了测试不同种群规模参数对 GSDE 的性能影响，本节将 GSDE 与采样不同种群规模的变种算法进行比较.由于初始设置中 GSDE 的种群规模为 $N=50$，本节将 GSDE 与分别采用种群规模为 $N=10$，100 和 150 的变种算法比较.为方便阐述，本节将这些变种算法分别命名为 GSDE($N=10$)，GSDE($N=100$)和 GSDE($N=150$).同时，原始的 GSDE 也命名为 GSDE($N=50$)，以便更直观地进行比较.

当种群规模设为较小值 $N=10$ 时，GSDE($N=10$)在大多数测试问题上性能明显下降，共有 10 个测试问题的表现劣于基准算法 GSDE($N=50$)，仅在 1 个测试问题上表现相当，1 个测试问题上优于 GSDE($N=50$).这表明较小的种群规模难以维持种群多样性，算法有容易陷入局部最优的风险，导致其全局搜索能力下降.

相比之下，GSDE($N=100$)与 GSDE($N=150$)的整体性能表现更接近 GSDE($N=50$).GSDE($N=100$)在 4 个测试问题上取得更优结果，其中包括在高维复杂问题 f_5，f_6 和 f_{11} 上取得明显改进，显示出较大的种群规模有助于提升解的精度与搜索稳定性；但在 8 个测试问题中性能有所下降，反映出种群规模扩大虽然可以增强全局探索，但也可能在资源受限时降低搜索效率.

GSDE($N=150$)在 3 个测试问题上优于 GSDE($N=50$)，在 1 个测试问题表现相当，在 8 个测试问题中略逊一筹.总体而言，其性能提升幅度与 GSDE($N=100$)相当，表现出一定的鲁棒性.

综上，适当增大种群规模有助于提升 GSDE 的优化性能，尤其在问题复杂或搜索空间维度较高的情况下更为明显.然而，种群规模过大将增加每代算法的计算负担，降低单位评估次数下的收敛速度.因此，在实际应用中应根据问题的复杂度和资源限制对种群规模进行合理设置，以高效平衡搜索精度与计算效率.此外，基于本节的算法比较，GSDE($N=50$)具有整体最优的性能，因此种群规模推荐设置为 50.

3.6 实际工程案例测试

为了验证算法解决实际工程问题的能力,本节将 GSDE 应用于大规模定日镜场布局优化问题[15].该问题旨在通过优化定日镜的部署位置及参数,最大化镜场的整体能量转化效率,属于典型的高维、计算密集型优化任务.实验采用圆形拓扑排列的定日镜场,分别考虑 25 圈和 30 圈两种部署规模,共计形成 6 个测试案例(T_1 至 T_6).其中,T_1,T_2,T_3 为 25 圈布局,分别需部署 1 000、2 000、3 000 个定日镜;T_4,T_5,T_6 为 30 圈布局,分别需部署 1 000、2 000、3 000 个定日镜.所有算法在每个案例上的最大函数评价次数均为 2×10^4 次,独立运行 25 次,结果以均值±标准差记录,如表 4 所示.

从优化结果来看,GSDE 在绝大多数案例中均取得了最佳性能,显示出其处理复杂实际优化问题的良好适用性和稳定性.具体而言,GSDE 在 T_1,T_3,T_4,T_6 案例上均显著优于对比算法;在 T_2 案例上与 DDE-ARA 性能相当,但仍优于 MDE 和 LSMDE.与其他算法相比,GSDE 在 6 个案例中取得 4 项最优、1 项相当、1 项稍逊的整体表现,显著优于 MDE 和 LSMDE,与 DDE-ARA 相比也具备明显优势.

GSDE 的优异性能可归因于其耿贝尔采样机制在处理高维、复杂优化问题时的优势:一方面,GLM 有助于在广阔的布局空间中高效探索潜在的高性能区域,避免早熟收敛;另一方面,GEM 能够在优化后期对镜场参数进行精细调整,从而有效提升最终解的质量.

结果表明,GSDE 算法能够有效应对实际能源优化场景中的挑战,为大规模定日镜场及其他类似复杂工程优化问题提供了可靠的解决方案.

表 4 基于大规模定日镜场优化实例的测试结果

Tab. 4 The test results on large-scale heliostat field optimization instances

测试案例	GSDE	MDE	DDE-ARA	LSMDE
T_1	8.01E-01±8.34E-03	7.64E-01±1.09E-02(+)	7.62E-01±9.22E-03(+)	7.82E-01±9.27E-03(+)
T_2	8.02E-01±6.64E-03	7.67E-01±9.74E-03(+)	8.06E-01±9.27E-03(≈)	7.69E-01±8.35E-03(+)
T_3	8.02E-01±7.76E-03	7.84E-01±7.68E-03(+)	7.87E-01±7.69E-03(+)	7.85E-01±6.01E-03(+)
T_4	8.13E-01±7.79E-03	7.90E-01±7.18E-03(+)	7.83E-01±6.56E-03(+)	8.12E-01±6.90E-03(≈)
T_5	8.15E-01±6.57E-03	7.92E-01±6.65E-03(+)	8.28E-01±6.88E-03(−)	7.95E-01±7.08E-03(+)
T_6	8.14E-01±5.16E-03	7.58E-01±8.03E-03(+)	8.01E-01±9.24E-03(+)	7.85E-01±7.35E-03(+)
+/≈/−	NA	6/0/0	4/1/1	5/1/0

4 结 论

本文针对差分进化算法在处理复杂优化问题中面临的搜索多样性不足、易陷入局部最优等痛点问题,提出了一种融合耿贝尔采样机制的差分进化算法,即 GSDE.该算法通过引入两种基于耿贝尔采样的关键变异策略,有效提升了种群个体的多样性与搜索精度,从而增强了算法在复杂优化环境下的适应能力.

为了验证所提出方法的有效性,本文在大量具有代表性的 1 000 维测试函数和应用实例上,分别与多种现有主流 DE 变体进行了对比实验.实验结果表明,GSDE 在大多数测试函数上取得了更优或竞争性的性能,体现了良好的稳定性与鲁棒性,尤其在高维多峰复杂问题中表现出显著优势.

未来的研究将进一步探索 GSDE 在多目标优化、大规模优化和动态优化环境下的扩展能力,并考虑与其他代理模型、协同进化机制的融合,以提升其在实际工程问题中的实用性与通用性.

附录见电子版(DOI:10.16366/j.cnki.1000-2367.2025.07.11.0001).

参 考 文 献

[1] ZHAN Z H,SHI L,TAN K C,et al.A survey on evolutionary computation for complex continuous optimization[J].Artificial Intelligence Review,2022,55(1):59-110.

[2] PANT M,ZAHEER H,GARCIA-HERNANDEZ L,et al.Differential Evolution:a review of more than two decades of research[J].Engineering Applications of Artificial Intelligence,2020,90:103479.

[3] REYES-DAVILA E,HARO E H,CASAS-ORDAZ A,et al.Differential evolution:a survey on their operators and variants[J].Archives of Computational Methods in Engineering,2025,32(1):83-112.

[4] 江璞玉,刘均,周奇,等.大规模黑箱优化问题元启发式求解方法研究进展[J].中国舰船研究,2021,16(4):1-18.
JIANG P Y,LIU J,ZHOU Q,et al.Advances in meta-heuristic methods for large-scale black-box optimization problems[J].Chinese Journal of Ship Research,2021,16(4):1-18.

[5] WANG K Y,GAO S C,ZHOU M C,et al.Fractional order differential evolution[J].IEEE Transactions on Evolutionary Computation,2025,29(3):822-835.

[6] 闫李,马佳慧,柴旭朝,等.基于知识引导的自适应动态多模态差分进化算法[J].控制与决策,2023,38(11):3048-3056.
YAN L,MA J H,CHAI X Z,et al.Adaptive dynamic multimodal differential evolution algorithm based on knowledge guidance[J].Control and Decision,2023,38(11):3048-3056.

[7] ZHAN Z H,LIU X F,ZHANG H X,et al.Cloudde:a heterogeneous differential evolution algorithm and its distributed cloud version[J].IEEE Transactions on Parallel and Distributed Systems,2017,28(3):704-716.

[8] GE Y F,YU W J,LIN Y,et al.Distributed differential evolution based on adaptive mergence and split for large-scale optimization[J].IEEE Transactions on Cybernetics,2018,48(7):2166-2180.

[9] LI J Y,DU K J,ZHAN Z H,et al.Distributed differential evolution with adaptive resource allocation[J].IEEE Transactions on Cybernetics,2023,53(5):2791-2804.

[10] ZHAN Z H,ZHANG J,LIN Y,et al.Matrix-based evolutionary computation[J].IEEE Transactions on Emerging Topics in Computational Intelligence,2022,6(2):315-328.

[11] FONSECA T H L,NASSAR S M,DE OLIVEIRA A C M,et al.Low-dimensional space modeling-based differential evolution for large-scale global optimization problems[J].IEEE Transactions on Evolutionary Computation,2023,27(5):1529-1543.

[12] 陈思佳,王庆泉,周丽.基于改进差分进化法的超密集可见光通信网络资源分配研究[J].激光杂志,2025,46(2):185-192.
CHEN S J,WANG Q Q,ZHOU L.Research on resource allocation in ultra dense visible light communication networks based on improved differential evolution method[J].Laser Journal,2025,46(2):185-192.

[13] 郭晶晶,王建华,于沫尧,等.基于改进 DE-SQP 算法的运载火箭轨迹优化方法研究[J].战术导弹技术,2025(1):94-103.
GUO J J,WANG J H,YU M Y,et al.Research on the optimization method of launch vehicle trajectory based on improved DE-SQP algorithm[J].Tactical Missile Technology,2025(1):94-103.

[14] QIAO K,BAN X,CHEN P,et al.Performance comparison of CEC 2025 competition entries on numerical optimization considering accuracy and speed[EB/OL].[2025-07-02].https://github.com/P-N-Suganthan/2025-CEC.

[15] DUAN D T,LI J Y,SUN B,et al.Large-scale heliostat field optimization for solar power tower system using matrix-based differential evolution[J].IEEE Transactions on Artificial Intelligence,2025,6(9):2422-2436.

Gumbel sampling-based differential evolution

Zhang He[1a], Wang Chuan[1b], Li Jianyu[2]

(1. a. Faculty of Education; b. College of Software, Henan Normal University, Xinxiang 453007, China;
2. College of Artificial Intelligence, Nankai University, Tianjin 300350, China)

Abstract: To address the issues of insufficient search diversity and premature convergence in differential evolution when solving complex optimization problems, this paper proposes a novel Gumbel sampling-based differential evolution. The proposed algorithm incorporates two innovative mutation strategies. The first is a Gumbel learning-based mutation strategy, which enhances the quality of generated individuals by applying Gumbel sampling to high-quality solutions. The second is a Gumbel sampling-based elite mutation strategy, which conducts local search around elite individuals using the Gumbel distribution to strengthen local exploitation. The combination of these two strategies effectively improves both population diversity and search precision. Extensive experiments were conducted on various benchmark functions and real-world applications, comparing the proposed algorithm with several state-of-the-art DE variants. Experimental results show that the proposed algorithm outperforms the compared algorithms on most test problems, exhibiting superior global search ability, convergence speed, and robustness. This study provides an effective new approach for solving complex optimization problems with intelligent evolutionary techniques.

Keywords: differential evolution; gumbel sampling; mutation strategy; global optimization; evolutionary computation

[责任编校 陈留院 杨浦]

附　录

表 S1　GSDE 与前沿 DE 算法在 IEEE CEC 2025 竞赛测试集的实验结果

Tab. S1　Experimental results of GSDE and advanced DEs on IEEE CEC 2025 benchmark

问题	GSDE	MDE	DDE-ARA	LSMDE
F_1	0.00E+00±0.00E+00	3.41E-07±2.78E-07(+)	0.00E+00±0.00E+00(≈)	3.28E-07±2.65E-07(+)
F_2	8.91E-01±2.73E+00	1.68E+03±8.96E+02(+)	1.02E+00±3.13E+00(+)	1.65E+03±8.94E+02(+)
F_3	2.48E+01±3.34E+01	6.01E+01±1.82E+01(+)	2.91E+01±3.22E+01(+)	6.12E+01±2.15E+01(+)
F_4	4.77E+01±2.31E+01	1.86E+02±1.02E+01(+)	5.11E+01±2.16E+01(+)	1.70E+02±9.48E+00(+)
F_5	5.45E-07±1.92E-07	3.10E-05±1.19E-05(+)	5.56E-07±1.87E-07(≈)	2.92E-05±1.35E-05(+)
F_6	8.42E+01±1.94E+01	2.12E+02±1.19E+01(+)	9.08E+01±2.35E+01(+)	1.98E+02±1.36E+01(+)
F_7	4.33E+01±1.93E+01	1.88E+02±1.17E+01(+)	4.51E+01±1.94E+01(≈)	1.95E+02±1.38E+01(+)
F_8	8.88E-01±1.84E+00	0.00E+00±0.00E+00(−)	9.57E-01±2.09E+00(+)	0.00E+00±0.00E+00(−)
F_9	3.31E+03±6.23E+02	7.19E+03±2.61E+02(+)	3.68E+03±5.49E+02(+)	7.36E+03±2.21E+02(+)
F_{10}	1.83E+01±1.64E+01	6.10E+01±1.05E+01(+)	1.92E+01±1.39E+01(+)	5.93E+01±1.23E+01(+)
F_{11}	1.12E+04±6.58E+03	8.17E+03±5.47E+03(−)	1.22E+04±7.40E+03(+)	7.62E+03±5.43E+03(−)
F_{12}	1.09E+02±4.44E+01	9.02E+01±1.16E+01(−)	1.20E+02±4.62E+01(+)	8.38E+01±1.18E+01(−)
F_{13}	2.52E+01±1.53E+01	6.67E+01±6.09E+00(+)	2.82E+01±1.43E+01(+)	6.39E+01±6.53E+00(+)
F_{14}	1.34E+01±6.32E+00	4.37E+01±4.04E+00(+)	1.40E+01±6.70E+00(≈)	4.52E+01±4.22E+00(+)
F_{15}	6.29E+02±1.42E+02	1.37E+03±2.22E+02(+)	6.63E+02±1.72E+02(−)	1.41E+03±2.52E+02(+)
F_{16}	1.40E+02±7.63E+01	1.01E+02±5.38E+01(+)	1.48E+02±6.49E+01(+)	9.42E+01±5.07E+01(+)
F_{17}	2.30E+02±2.82E+02	4.25E+01±3.61E+00(−)	2.60E+02±3.03E+02(+)	4.26E+01±3.17E+00(−)
F_{18}	1.18E+01±4.47E+00	2.88E+01±2.96E+00(+)	1.33E+01±4.44E+00(+)	2.62E+01±2.74E+00(+)
F_{19}	1.66E+02±9.95E+01	6.39E+01±2.23E+01(−)	1.81E+02±9.01E+01(+)	6.57E+01±2.01E+01(−)
F_{20}	2.33E+02±1.72E+01	3.79E+02±9.25E+00(+)	2.40E+02±1.96E+01(+)	3.48E+02±1.05E+01(+)
F_{21}	8.73E+01±1.36E-01	1.00E+02±8.32E-10(+)	1.00E+02±1.66E-01(+)	9.52E+01±8.69E-10(+)
F_{22}	3.89E+02±2.51E+01	5.29E+02±1.23E+01(+)	4.41E+02±2.09E+01(+)	5.25E+02±1.18E+01(+)
F_{23}	4.84E+02±2.80E+01	5.97E+02±1.44E+01(+)	5.08E+02±2.70E+01(+)	6.19E+02±1.19E+01(+)
F_{24}	3.65E+02±1.36E+00	3.87E+02±4.72E-01(+)	3.86E+02±1.48E+00(+)	3.72E+02±5.08E-01(≈)
F_{25}	1.42E+03±5.12E+02	2.57E+03±1.31E+02(+)	1.59E+03±4.45E+02(+)	2.33E+03±1.19E+02(+)
F_{26}	5.16E+02±8.55E+00	4.79E+02±1.23E+01(−)	5.24E+02±7.58E+00(≈)	4.37E+02±1.15E+01(−)
F_{27}	3.04E+02±4.55E+01	3.13E+02±3.52E+01(+)	3.30E+02±4.83E+01(+)	3.25E+02±3.53E+01(+)
F_{28}	6.03E+02±6.68E+01	9.09E+02±1.46E+02(+)	6.33E+02±8.14E+01(+)	9.08E+02±1.66E+02(+)
F_{29}	4.01E+03±3.05E+03	2.47E+03±1.77E+02(−)	4.69E+03±2.76E+03(+)	2.58E+03±1.64E+02(−)
+/≈/−	NA	22/0/7	23/5/1	21/1/7