

基于吸引互斥多策略改进雾凇算法

胡印福

温州大学计算机与人工智能学院, 浙江 温州

收稿日期: 2026年3月13日; 录用日期: 2026年4月12日; 发布日期: 2026年4月23日

摘要

雾凇算法(RIME)是近年提出的一种模拟雾凇生长现象的元启发式优化算法, 具有结构简洁、参数较少等优势。但在处理高维复杂的工程优化问题时, 该算法仍存在种群多样性缺失、搜索后期易陷入局部最优解以及收敛精度受限等不足。因此, 研究如何通过多策略协同改进以提升RIME的全局寻优性能具有重要的理论价值与应用意义。针对上述问题, 本文提出一种基于吸引互斥多策略改进的雾凇算法。首先, 在算法搜索阶段引入吸引互斥机制, 通过模拟粒子间的动态引力与斥力作用来实时调整搜索步长, 使个体在搜索前期具备更广的探索范围, 在后期实现精准开发, 从而有效平衡算法的全局探索与局部搜索能力。其次, 借鉴冠豪猪优化算法(CPO)中的防御机制对算法进行二次改进: 利用气味防御机制的随机扩散特性增强种群的扰动性, 防止算法在迭代后期陷入早熟收敛; 同时, 通过物理攻击机制的强扰动性能, 提升个体在复杂解空间中跳出局部极值的概率。两种策略相互配合, 从动力学调整与多样性维护两个维度优化了原算法的生长演化过程。为验证改进算法的性能, 选取多组基准测试函数进行仿真实验。结果表明, 与原始RIME、CPO及其他典型优化算法相比, 改进算法在寻优精度、收敛速度和统计鲁棒性方面均有显著提升。特别是在多峰复杂函数的处理上, 该算法展现出卓越的跳出局部最优能力。通过Wilcoxon秩和检验进一步证明了改进策略的统计显著性, 验证了所提算法在解决复杂优化任务时的有效性。

关键词

元启发式算法, 雾凇算法, 吸引互斥机制, 冠豪猪算法, 多策略协同

Improved Rime Optimization Algorithm Based on Multi-Strategy of Attraction-Repulsion Mechanism

Yinfu Hu

School of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou Zhejiang

Received: March 13, 2026; accepted: April 12, 2026; published: April 23, 2026

Abstract

The Rime Optimization Algorithm (RIME) is a recently proposed metaheuristic algorithm inspired by the physical growth of rime. While it features a simple structure and few parameters, RIME often suffers from insufficient population diversity, premature convergence, and limited accuracy when dealing with complex high-dimensional optimization problems. Therefore, researching how to improve the global optimization performance of RIME through multi-strategy collaborative improvement has important theoretical value and application significance. This paper proposes an improved rime optimization algorithm (ACRIME) based on a multi-strategy coordination approach. First, an attraction-repulsion mechanism is introduced during the algorithm's search phase. By simulating the dynamic attraction and repulsion between particles, the search step size is adjusted in real time, allowing individuals to have a wider exploration range in the early stages of the search and achieve precise development in the later stages, thus effectively balancing the algorithm's global exploration and local search capabilities. Second, the algorithm is further improved by borrowing the defense mechanism from the Caucasian porcupine optimization algorithm (CPO): the random diffusion characteristics of the odor defense mechanism are used to enhance the perturbation of the population, preventing the algorithm from falling into premature convergence in the later stages of iteration; at the same time, the strong perturbation performance of the physical attack mechanism is used to increase the probability of individuals escaping local optima in complex solution spaces. The two strategies work together to optimize the growth and evolution process of the original algorithm from two dimensions: dynamic adjustment and diversity maintenance. To verify the performance of the improved algorithm, simulation experiments were conducted using multiple benchmark test functions. The results show that compared with the original RIME, CPO, and other typical optimization algorithms, the improved algorithm has significant improvements in optimization accuracy, convergence speed, and statistical robustness. Especially in the handling of multimodal complex functions, the algorithm exhibits excellent ability to escape local optima. The statistical significance of the improved strategy was further demonstrated by the Wilcoxon rank-sum test, which verifies the effectiveness of the proposed algorithm in solving complex optimization tasks.

Keywords

Metaheuristic Algorithms, Rime Optimization Algorithm, Attraction-Repulsion Mechanism, Crested Porcupine Optimizer, Multi-Strategy Coordination

Copyright © 2026 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着科技的快速发展, 工程设计、机器学习等领域出现了大量复杂的优化问题。这些问题往往极其复杂, 传统方法很难在短时间内找到满意的答案。元启发式算法因为不需要复杂的求导计算、适应性强且容易实现, 成为了解决这类难题的利器。根据灵感来源的不同, 这类算法主要分为四类: (1) **进化类算法**: 包括遗传算法(GA) [1]、差分进化(DE) [2]、协方差矩阵自适应进化策略(CMA-ES) [3]、龙格-库塔优化器(RUN) [4]、加权向量平均(INFO) [5]; (2) **群体智能算法**: 包括粒子群优化(PSO) [6]、鲸鱼优化算法(WOA) [7]、果蝇优化算法(FOA) [8]、哈里斯鹰优化(HHO) [9]、黏菌算法(SMA) [10]、鸚鵡优化器(PO) [11]、

水母群算法(SSA) [12]、饥饿游戏搜索(HGS) [13]; (3) **基于人类行为的算法**: 如群体搜索优化(GSO) [14]、人类记忆优化(HMO) [15]; (4) **基于物理现象的算法**: 如雾凇优化算法(RIME) [16]、水循环算法(WCA) [17]。它们通过模拟自然规律, 为解决复杂的寻找最优解任务提供了多样的思路。

在这些算法中, 雾凇优化算法(RIME)是一种较新的物理类算法。它模拟了雾凇的生长过程, 具有结构简单、初期求解速度快的优点。然而, 正如“没有免费的午餐”定理所言, 没有哪种算法是万能的。RIME 在处理极度复杂的任务时, 也暴露了一些共性问题: 首先, 在搜索后期, 算法的种群多样性会严重下降, 导致其容易困在局部最优的“死胡同”里出不来。其次, 它在“大范围探索”和“小范围精确挖掘”之间的平衡做得不够好, 步伐调整不够灵活。这导致算法要么找得不够广, 要么找得不够准。虽然之前的研究尝试过引入一些常规策略来改进 RIME, 但往往只起到了局部的修补作用, 没有从根本上解决广泛搜索与精准挖掘之间的矛盾。因此, 如何让算法在复杂环境中既能灵活跳出陷阱, 又能精准锁定最优解, 仍然是目前亟待解决的难点。

针对原始 RIME 算法存在的上述缺陷, 本文提出了一种基于吸引互斥多策略改进的雾凇算法(ACRIME)。本文的主要贡献体现在以下几个方面:

(1) 引入吸引互斥机制, 通过模拟粒子间的动态引力与斥力关系来实时调节搜索步长, 以此显著提高原 RIME 算法在搜索初期的全局探索能力, 确保种群能快速覆盖解空间。

(2) 由于原 RIME 中硬凇穿刺阶段过于依赖最优解, 本文创造性地融合了冠豪猪优化算法(CPO)中的气味防御和物理攻击机制, 利用气味扩散的随机性增加扰动, 并通过物理攻击的强局部跳变增强算法的开发精度, 从而有效解决局部停滞问题。

(3) 通过这两种策略的协同作用, 使得 ACRIME 算法能够动态权衡探索与开发的关系, 从本质上提升了算法的全局寻优性能与统计鲁棒性。

本文接下来的章节安排如下: 第二章将详细介绍雾凇优化算法(RIME)及相关机制的理论背景; 第三章将阐述 ACRIME 算法的数学模型、改进策略的具体实现及其算法流程; 第四章将在标准 CEC2017 基准测试函数集上进行仿真实验, 并与多种主流算法进行对比分析; 第五章对全文的研究工作进行总结, 并指出未来的研究方向。

2. 相关理论知识

RIME 算法

雾凇优化算法(RIME)是 Su 等人[16]近期提出的一种新型算法。它的灵感来源于自然界, 模拟了空气中的水蒸气遇冷并在物体表面结成冰晶(即雾凇)的过程。在寻找最优解时, 算法就像雾凇生长一样, 在解空间内不断蔓延来捕捉目标。

RIME 算法主要依靠两个核心机制来运行:

1. **软凇搜索(Soft Rime Search)**: 主要用在搜索前期, 负责大范围地“撒网”, 确保能探索到全局的各个角落。

2. **硬凇穿刺(Hard Rime Puncture)**: 通过加强信息交流来提高搜索的精确度, 并帮助算法跳出“死胡同”(局部最优陷阱)。

在**软凇搜索**阶段, 算法模拟了水汽粒子飘动并最终凝结的过程。简单来说, 水汽粒子一开始四处飘荡, 一旦靠近已有的软凇, 就会被吸附上去。随着软凇越长越大, 它吸附新粒子的能力也越来越强。不过, 受环境限制, 软凇不会无限生长, 而是会慢慢饱和并稳定下来。这代表着算法从前期的广泛探索, 逐渐过渡到了稳定状态。具体的计算方法如公式(1)(2)所示。

$$R_{ij}^{new} = R_{best,j} + Rf \times \left(h \times (B_{\max(i,j)} - B_{\min(i,j)}) + B_{\min(i,j)} \right), r_2 < E \quad (1)$$

$$Rf = r_1 \times \cos \theta \times \beta \quad (2)$$

其中 R_{ij}^{new} 是自由状态粒子移动后的位置, $R_{best,j}$ 表示当前最优粒子的位置信息, r_1 是取值在 $[-1, 1]$ 之间的随机数, β 代表环境因素, 与算法一起通过迭代更新, 以动态地改变和控制算法的收敛性。 h 是介于 $[0, 1]$ 之间的随机值。

$$\theta = \pi \cdot \frac{t}{10 \cdot T} \quad (3)$$

$$\beta = 1 - \left[\frac{w \cdot t}{T} \right] / w \quad (4)$$

$$E = \sqrt{t/T} \quad (5)$$

在公式(3)中, t 和 T 分别代表当前迭代次数和最大迭代次数。在公式(4)中, $[\]$ 表示向最近整数取整的运算; w 为控制步长的系数, 其取值固定为 5; E 则表示处于游离态的粒子被捕捉并发生凝结的概率。

在硬淤穿刺中, 强风影响下的硬淤生长规则要比软淤简单。在强风环境中, 其他因素对颗粒的影响将变得很小, 风向将成为影响颗粒运动的主要因素。此外, 由于生长方向相同, 硬淤很容易出现刺穿交叉现象。此外, 随着雾淤的生长, 穿刺现象会更加频繁。因此, 硬淤穿刺策略可以让代理交换粒子, 提高算法的收敛性。数学模型如公式所示。

$$R_{ij}^{new} = R_{best,j}, r_3 < F^{normr}(S_i) \quad (6)$$

其中, $F^{normr}(S_i)$ 是归一化的适应度值, 代表当前搜索代理出现硬淤穿刺的概率; r_3 是取值在 $[0, 1]$ 之间的随机值。

3. ACRIME 算法

3.1. AROA 算法描述

吸引 - 排斥优化算法(Attraction-Repulsion Optimization Algorithm, AROA)是由 Cymerys 和 Oszust [18] 于 2024 年提出的一种新型元启发式搜索算法。该算法通过模拟自然界中群体成员之间的吸引与排斥现象, 旨在构建一个更为平衡的探索(Exploration)与开发(Exploitation)机制。

AROA 的核心机制在于其独特的吸引 - 排斥算子, 该算子根据候选解邻域内其他个体的质量来决定其移动方向。对于第 i 个个体, 其受到的邻域综合影响向量 \mathbf{n}_i 定义为:

$$\mathbf{n}_i = \frac{1}{k} \sum_{j=1}^k c \cdot (\mathbf{x}_j - \mathbf{x}_i) \cdot I(d_{i,j}, d_{i,\max}) \cdots (f_i, f_j) \quad (7)$$

其中, k 表示动态变化的邻居数量, c 为步长因子, $d_{i,j}$ 是第 i 个代理与第 j 个代理之间的距离, $d_{i,\max}$ 而是种群中最远的距离。 I 函数用于评估第 i 个解的影响力强度, 其计算公式为:

$$D = \begin{bmatrix} d_{1,1} & d_{1,2} & \cdots & d_{1,n} \\ d_{2,1} & d_{2,2} & \cdots & d_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n,1} & d_{n,2} & \cdots & d_{n,n} \end{bmatrix} \quad (8)$$

$$d^2(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{\dim} (x_i^k - x_j^k)^2 \quad (9)$$

$$t_{\max} = \left\lceil \frac{fes_{\max} - n}{2n} \right\rceil \quad (10)$$

$$k = \left\lfloor \left(1 - \frac{t}{t_{\max}} \right) \cdot n \right\rfloor + 1 \quad (11)$$

$$I(d_{i,j}, d_{i,\max}) = 1 - \frac{d_{i,j}}{d_{i,\max}} \quad (12)$$

此外, 方向函数 $s(f_i, f_j)$ 决定了力属性(吸引或排斥)。在最小化问题中, 其定义如下:

$$s(f_i, f_j) = \begin{cases} 1, & f_i > f_j \\ 0, & f_i = f_j \\ -1, & f_i < f_j \end{cases} \quad (13)$$

除了邻域算子外, AROA 还引入了向全局最优解吸引的向量 \mathbf{b}_i , 并受到非线性控制因子 m 的调节, 以确保算法在搜索后期具有更强的收敛性。为了增强局部搜索能力并维持种群多样性, AROA 集成了基于修正布朗运动、三角函数和随机跳转的局部搜索算子 \mathbf{r}_i 。最终, 个体的更新公式描述为:

$$\mathbf{x}_i(t) = \mathbf{x}_i(t-1) + \mathbf{n}_i + \mathbf{b}_i + \mathbf{r}_i \quad (14)$$

最后, AROA 引入了群体激励算子(如涡流形成策略)和基于适应度对比的记忆机制。如果更新后的个体适应度未获得改善, 算法将利用记忆机制恢复其先前的状态, 从而有效防止算法陷入局部次优区域。实验证明, AROA 在处理复杂的 CEC 基准测试函数和现实工程约束优化问题中表现出极强的竞争优势。

3.2. 冠豪猪的气味防御和物理攻击机制

冠豪猪优化算法(Crested Porcupine Optimizer, CPO)是由 Abdel-Basset 等人[19]于 2024 年提出的一种新型元启发式优化算法。该算法的灵感源于冠豪猪在面对不同程度威胁时所采取的四种防御行为: 视觉威慑(Sight)、听觉威慑(Sound)、气味防御(Odor)以及物理攻击(Physical attack)。根据算法设计, 前两种策略(视觉和听觉)主要用于全局探索, 而气味防御和物理攻击则构成了算法的开发阶段(Exploitation phase), 旨在对候选解周围的局部区域进行精细化搜索。

在 CPO 算法的局部开发(Exploitation)阶段, 通过动态权衡气味防御(Scent Defense)与物理攻击(Physical Attack)两种机制, 赋予了算法卓越的局部寻优精度以及摆脱局部极值陷阱的能力。

1) 气味防御策略(Third Defensive Strategy)

该策略模拟了自然界中冠豪猪在捕食者逼近时, 通过释放异味信号以阻碍对方进攻的生物行为。在 CPO 算法的设计逻辑中, 这一生物特性被抽象建模为围绕当前个体解邻域开展的局部搜索行为。通过该机制, 算法能够在搜索空间内的潜在解区域进行精细化探测, 从而有效提升收敛精度。其位置更新公式如下:

$$\bar{\mathbf{x}}_i^{t+1} = (1 - \bar{U}_1) \times \bar{\mathbf{x}}_i^t + \bar{U}_1 \times (\bar{\mathbf{x}}_{r_1}^t + S_i^t \times (\bar{\mathbf{x}}_{r_2}^t - \bar{\mathbf{x}}_{r_3}^t) - \tau_3 \times \bar{\delta} \times \gamma_i \times S_i^t) \quad (15)$$

其中, $\bar{\mathbf{x}}_i^t$ 表示第代的个体位置, \bar{U}_1 是一个模拟捕食者行为状态的二值向量, r_1, r_2, r_3 是种群中的随机索引。核心参数气味扩散因子 S_i^t 定义为:

$$S_i^t = \exp\left(\frac{f(\bar{\mathbf{x}}_i^t)}{\sum_{k=1}^N f(\bar{\mathbf{x}}_k^t) + \epsilon}\right) \quad (16)$$

S_i' 的引入使得开发过程能够根据适应度值非线性地调整搜索范围，从而在保证局部开发精度的同时兼顾种群多样性。

2) 物理攻击策略(Fourth Defensive Strategy)

当防御失败且捕食者极度接近时，冠豪猪会利用背部坚硬的刺进行倒退撞击。CPO 将这一过程模拟为一维空间下的完全非弹性碰撞，其数学表达为：

$$\vec{x}_i^{t+1} = \vec{x}_{CP}^t + (\alpha(1-\tau_4) + \tau_4) \times (\delta \times \vec{x}_{CP}^t - \vec{x}_i^t) - \tau_5 \times \delta \times \gamma_i \times \vec{F}_i^t \quad (17)$$

在此公式中， \vec{x}_{CP}^t 代表目前已发现的最佳全局解(即冠豪猪的位置)， \vec{F}_i^t 表示受碰撞定律启发产生的平均作用力：

$$\vec{F}_i^t = \bar{\tau}_6 \times \frac{m_i \times (\vec{v}_i^{t+1} - \vec{v}_i^t)}{\Delta t} \quad (18)$$

物理攻击策略通过强制候选解向全局最优解(\vec{x}_{CP}^t)方向剧烈摆动并进行小范围修正，极大地加速了算法的收敛速度，使其在处理大规模复杂问题时表现出卓越的精确度。

总的来说，CPO 算法的开发阶段通过气味策略的随机局部偏移和物理攻击策略的加速收敛机制，在局部搜索的广度与深度之间达成了良好的平衡，这使得 CPO 在处理高维约束优化问题时具有显著的性能优势。

3.3. ACRIME 算法描述

针对标准雾凇优化算法(RIME)在求解复杂多峰优化问题时存在探索与开发能力不平衡、容易陷入局部最优等问题，本文提出了一种融合多策略改进的雾凇优化算法(ACRIME)。标准 RIME 算法凭借其模拟自然界雾凇形成过程的软凇和硬凇穿刺机制，具有搜索路径新颖、寻优效率较高等优势。然而，面对极其复杂的解空间时，标准 RIME 的搜索机制往往难以兼顾前期的种群多样性与后期的高精度收敛，极易在达到全局最优前出现早熟现象。

为有效平衡算法的全局探索与局部开发能力，本文对 ACRIME 的算法框架进行了重新梳理，明确了吸引 - 互斥优化算法(AROA)、RIME 软凇机制与冠豪猪优化器(CPO)之间的分工协作关系。在 ACRIME 中，这三种算子并非概率性选择其一，而是采取了分阶段串行协作的运行机制。单次迭代的完整流程依次为：首先，在原 RIME 的软凇阶段之前，深度融合了 AROA 的吸引互斥机制。该机制通过评估邻域解的质量来动态调整当前候选解的移动方向(即被优质解吸引或被劣质解排斥)，有效扩大了种群在解空间中的覆盖面；其次，保留原 RIME 的软凇机制作为中期的过渡搜索，利用粒子间的交叉行为实现平滑衔接；最后，针对原算法的硬凇穿刺阶段，本文引入了 CPO 中的防御行为模型进行完全替换。具体采用动态权衡气味扩散与物理攻击的策略，使算法能够在迭代后期更加聚焦于当前最优解附近的精细化搜索。ACRIME 算法的具体流程如下所示：

Algorithm 1 ACRIME 的原理与实现

Require: 种群规模 N ，问题维度 dim ，最大迭代次数 $MaxFEs$ ，其他相关参数

Ensure: 当前最优解 X_{best}

- 1: 初始化种群位置 $X_i(i = 1, 2, \dots, N)$ 并计算适应度值，确定 X_{best} ;
- 2: **while** $it \leq MaxFEs$ **do**
- 3: 更新环境参数 a ;
- 4: **for** $i = 1$ **to** N **do**
- 5: {引入 AROA 机制增强探索}

```

6:   使用吸引互斥机制更新个体  $X_i$  的位置信息;
7: end for
8: for  $i = 1$  to  $N$  do
9:   计算个体的适应度值, 更新当前最优解  $X_{best}$ ;
10: end for
11: 通过公式 (5)更新系数  $E$ ;
12: 对种群适应度进行归一化处理;
13: for  $i = 1$  to  $N$  do
14:   生成随机数  $r_1, r_2$ ;
15:   if  $r_1 < P$  (软湍触发条件) then
16:     使用公式 (1-2)执行软湍搜索更新位置;
17:   end if
18:   if  $r_2 > P$  (局部开发触发条件) then
19:     {引入 CPO 动态防御机制增强开发}
20:     使用动态权衡气味扩散和物理攻击机制更新位置;
21:   end if
22: end for
23: for  $i = 1$  to  $N$  do
24:   边界处理及适应度计算;
25:   通过贪婪机制更新当前最优个体位置  $X_{best}$ ;
26: end for
27:  $it = it + 1$ ;
28: end while
29: return  $X_{best}$ 

```

从理论层面上看, 这种分阶段协作的融合方式高度契合群智能优化的“探索 - 开发平衡(Exploration-Exploitation Trade-off)”理论。在搜索前期, 首要任务是避免局部最优, AROA 的引入利用排斥机制强制个体分散, 最大化了全局探索(Exploration)能力; 随着迭代推进, 软湍机制的随机游走特性适时缩减了搜索步长; 而在搜索后期, 面对原 RIME 硬湍机制穿刺策略较为单一的缺陷, CPO 的防御模型提供了多梯度的邻域局部扰动能力(Exploitation), 大幅提升了局部开发的精度与整体收敛速度。这种“由广入深、先分散后聚集”的阶段分工协作, 有效保证了算法在复杂函数演化过程中的动力学合理性。

4. 实验结果与分析

为了系统验证吸引互斥机制以及冠豪猪防御策略(包括气味防御与物理攻击机制)对原始 RIME 算法性能改进的有效性, 本节基于 IEEE CEC 2017 基准测试函数集开展了一系列深度仿真实验。实验方案主要由以下两个核心部分组成: 首先, 通过消融实验评估各改进策略对算法性能提升的独立贡献与协同效应; 最后, 通过与多种优秀算法的对比实验, 全面验证 ACRIME 在处理复杂多峰优化任务时的竞争优势、收敛精度及统计鲁棒性。

4.1. 实验设置

4.1.1. 基准测试集

本实验采用包含 29 个测试函数的 IEEE CEC2017 标准测试集(已剔除不稳定函数 F_2), 以全面评估算

法在单峰收敛、多峰探索、处理变量强相关性以及在复杂环境下的寻优与平衡能力。

4.1.2. 参数设置与实验条件

为确保公平，所有对比实验均在同一软硬件环境(Intel Core i7, MATLAB 2024b)下进行，统一设置种群规模为 30，空间维度为 30 维，最大评价次数为 300,000 次，且每个函数独立运行 30 次以消除随机误差。

4.1.3. 统计分析方法

实验结果以 30 次独立运行的均值和标准差来衡量，并辅以 5%显著性水平的 Wilcoxon 符号秩检验 [20]与 Friedman 检验[21]，通过严格的非参数统计方法验证算法性能提升的显著性与整体稳健性。

4.2. 消融实验

本节通过消融实验验证 AROA 和 CPO 机制的有效性，共设置四种算法变体(见表 1)。

Table 1. Mechanism portfolio of ablation study variants

表 1. 消融实验各变体的机制组合方式

Brown 机制	ACRIME	ARIME1	CRIME	RIME
吸引互斥机制(AROA)	√	√		
冠豪猪防御机制(CPO)	√		√	

为严谨评估各机制的贡献，采用 Wilcoxon 符号秩检验(WSRT)进行统计分析(见表 2)，其中“+/-/-”分别代表 ACRIME 优于、等于或劣于对比算法的测试函数个数。

Table 2. Average rank and statistical results of different algorithms

表 2. 不同算法在 CEC2017 上的平均排名与统计结果

Brown 算法	排名(RANK)	+/-/-	平均排名(AVG)
ACRIME	1	~	1.2414
ARIME	4	26/2/1	3.2414
CRIME	2	21/7/1	2.7241
RIME	2	26/3/0	2.7241

根据统计结果，ACRIME 的平均排名最小(1.2414)且综合排名第一。在具体对比中，ACRIME 在绝大多数函数上显著优于原始 RIME (26 胜 3 平 0 负)以及单一机制变体。值得注意的是，仅使用单一机制有时甚至会导致性能下降，这强有力地证明了 AROA 与 CPO 机制之间存在显著的协同作用。只有将两者有机结合，才能最大限度激发算法潜能，实现最优的寻优精度。

4.3. 优秀算法对比实验

为了进一步验证 ACRIME 算法在处理复杂优化任务时的优越性能，本文将其与十种具有代表性的先

进元启发式算法进行了深度对比研究, 包括: 复合学习策略差分进化算法(DECLS) [22]、线性种群减容自适应差分进化算法(LSHADE) [23]、协同演化差分进化算法(CCDE) [24]、综合学习粒子群优化算法(CLPSO) [25]、改进型 Nelder-Mead 鲸鱼优化算法(ISNMWOA) [26]、加权雾凇优化算法(WHRIME) [27]、竞争学习蚁群算法(CLACO) [28]、加权差分进化算法(WDE) [29]、改进型 ϵ 约束自适应差分进化算法(LSHADE_cnEpSi) [30]以及引入 Levy 飞行的鲸鱼优化算法(LWOA) [31]。

表 3 汇总了 ACRIME 与 10 种对比算法的 Wilcoxon 秩和检验统计量和 Friedman 排名。

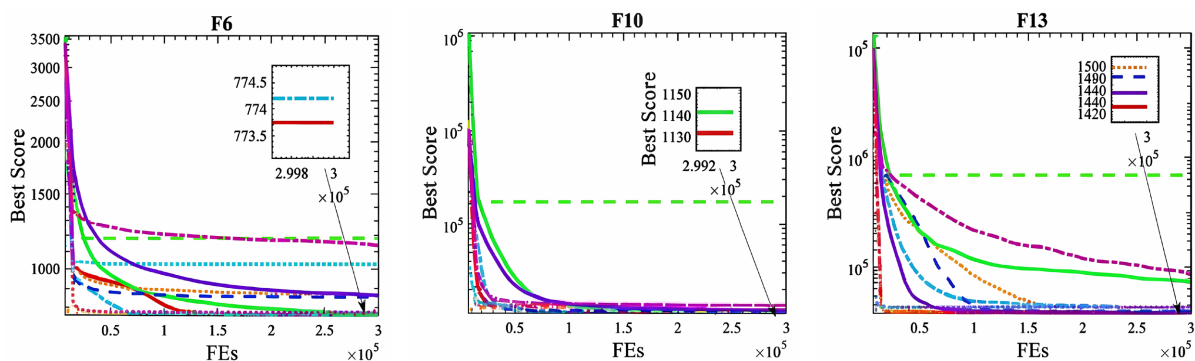
Table 3. Wilcoxon rank-sum test and Friedman ranking of ACRIME versus comparison algorithms
表 3. ACRIME 与对比算法的 Wilcoxon 秩和检验及 Friedman 排名

Brown 指标	ACRIME	DECLS	LSHADE	CCDE	CLPSO	ISNMWOA	WHRIME	CLACO	WDE	LSHADE_cnEpSi	LWOA
+/-/-	~	18/0/9	17/0/10	29/0/0	19/0/5	21/0/5	17/0/6	23/0/1	24/0/4	16/0/11	29/0/0
Mean Rank	2.9655	4.7586	4.4483	10.2069	4.6207	8.0345	4.6207	6.1379	5.5862	4.8966	8.7931
总排名	1	5	2	11	6	9	3	8	7	4	10

根据表 3 的统计结果显示, ACRIME 的 Friedman 值为 2.9655, 在全部 11 种对比算法中位居首位, 相较于位列第二的 LSHADE (4.4483)显著领先约 1.4828。

在 Wilcoxon 符号秩检验中, ACRIME 展现出了强大的寻优竞争力与统计显著性。具体而言: 1) 针对 CCDE 和 LWOA 算法, ACRIME 在全部 29 个测试函数上均取得了压倒性的全胜记录(29 胜 0 负); 2) 在与 CLACO 和 WDE 的对比中, 分别取得了 23 胜 1 负与 24 胜 4 负的显著优势。即使面对性能稳健的主流算法, ACRIME 依然表现出强劲的跨越式领先: 在与 ISNMWOA、CLPSO 及 DECLS 的竞争中, 分别获得了 21 胜、19 胜及 18 胜。值得注意的是, 针对实力较强的 LSHADE 系列变体, ACRIME 依然能够保持领先优势, 分别在与 LSHADE 和 LSHADE_cnEpSi 的对比中取得了 17 胜与 16 胜的战绩。此外, 面对同属于 RIME 改进框架的 WHRIME, ACRIME 凭借 17 胜 6 负的成绩进一步验证了本文多策略融合改进方案的优越性。

图 1 展示了 ACRIME 在收敛曲线上与优秀的算法对比情况, 可以看出 ACRIME 可以在多个多峰函数上比其他优秀算法具有更显著的性能。



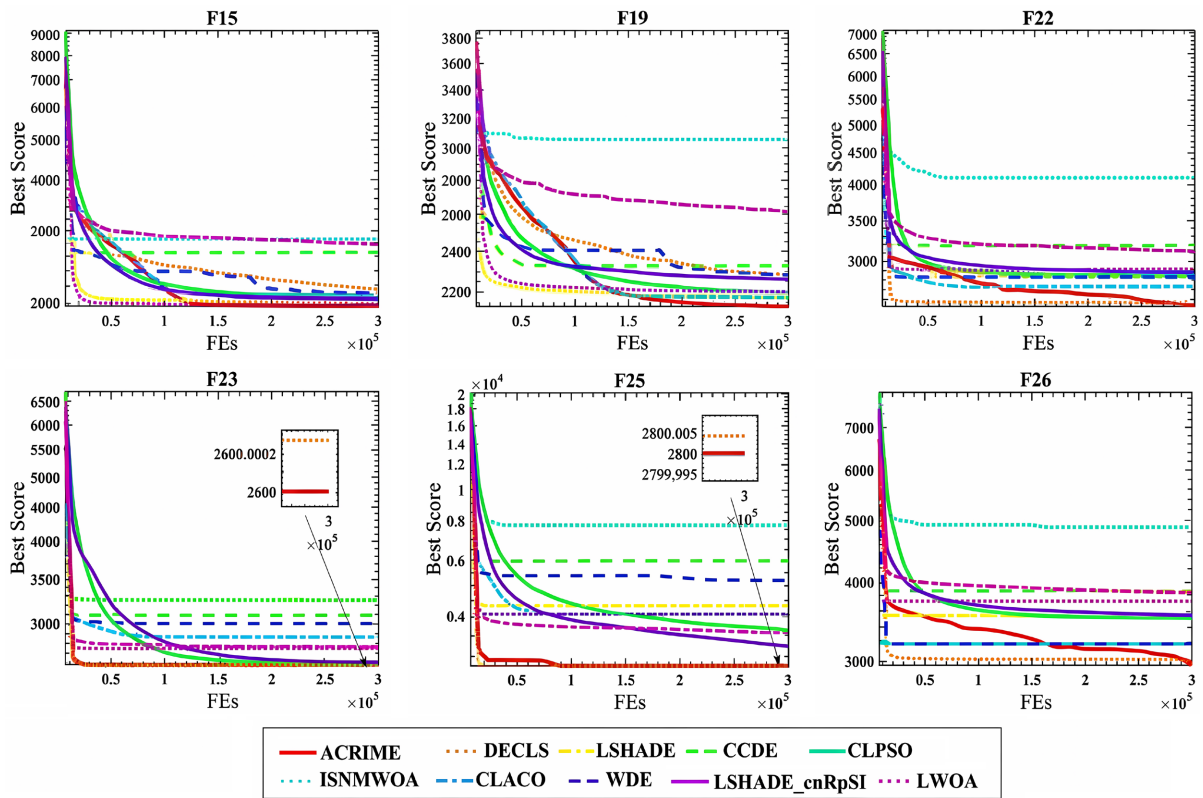


Figure 1. Convergence curves of ACRIME and comparison algorithms on CEC 2017
 图 1. ACRIME 与对比算法在 CEC 2017 上的收敛曲线

4.4. 实验小结

本章通过消融实验、基准函数测试两个方面，全面评估了 ACRIME 算法的性能。

首先，消融实验表明，结合吸引互斥机制(AROA)与冠豪猪防御机制(CPO)能产生很好的协同效果，证明了这两种改进策略对提升算法性能是必不可少的。

最后，在 IEEE CEC2017 测试集上与其他主流算法的对比实验和统计分析进一步证明，ACRIME 算法在计算精度、寻找最优解的速度和运行稳定性上都表现出色。

总而言之，ACRIME 算法不仅在理论机制上具有优势，在解决各种复杂的优化问题时也非常实用且高效。

5. 结论

本文针对原始雾凇优化算法(RIME)在处理高维复杂优化问题时存在的种群多样性不足、收敛精度受限以及易陷入局部最优等问题，提出了一种融合吸引互斥等多策略改进的雾凇优化算法(ACRIME)。

ACRIME 算法通过两种核心机制的协同作用，实现了算法搜索性能的质性提升：首先，在软凇搜索阶段之前，创新性地引入了吸引 - 互斥优化算法(AROA)的动力学机制。通过动态评估个体邻域解的质量并实时调整搜索个体的移动矢量，该机制极大地拓宽了算法在搜索前期的全局探索空间，为发现潜在在最优区域奠定了基础。其次，在局部寻优阶段，本文利用冠豪猪优化器(CPO)中能够动态权衡气味扩散与物理攻击的防御机制，成功替换了原算法中的硬凇穿刺机制。这一改进显著强化了算法在迭代后期的精细化开发能力，不仅提升了最优解的收敛精度，更大幅增强了算法在处理复杂多峰函数时的收敛速度。

在性能验证方面, 本文基于国际标准的 IEEE CEC2017 基准测试集开展了全方位的仿真实验。实验结果表明, 在与 LSHADE、WHRIME、DECLS 等十种先进的主流优化算法的综合对比中, ACRIME 在寻优精度、运行稳健性以及规避局部陷阱能力等方面均展现出了显著的优越性。此外, Wilcoxon 秩和检验与 Friedman 检验的统计结果进一步证实了所提改进策略的有效性与统计学显著性。

综上所述, ACRIME 通过多策略的深度融合, 成功构建了全局探索与局部开发之间的动态平衡体系。未来的研究工作将进一步探索该算法在工程约束优化、多目标任务以及大规模调度问题中的实际应用价值。

参考文献

- [1] Holland, J.H. (1973) Genetic Algorithms and the Optimal Allocation of Trials. *SIAM Journal on Computing*, **2**, 88-105. <https://doi.org/10.1137/0202009>
- [2] Storn, R. (1995) Differential Evolution—A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. Technical Report, International Computer Science Institute, 11.
- [3] Hansen, N. and Ostermeier, A. (2001) Completely Derandomized Self-Adaptation in Evolution Strategies. *Evolutionary Computation*, **9**, 159-195. <https://doi.org/10.1162/106365601750190398>
- [4] Ahmadianfar, I., Heidari, A.A., Gandomi, A.H., Chu, X. and Chen, H. (2021) RUN beyond the Metaphor: An Efficient Optimization Algorithm Based on Runge Kutta Method. *Expert Systems with Applications*, **181**, Article ID: 115079. <https://doi.org/10.1016/j.eswa.2021.115079>
- [5] Ahmadianfar, I., Heidari, A.A., Noshadian, S., Chen, H. and Gandomi, A.H. (2022) INFO: An Efficient Optimization Algorithm Based on Weighted Mean of Vectors. *Expert Systems with Applications*, **195**, Article ID: 116516. <https://doi.org/10.1016/j.eswa.2022.116516>
- [6] Kennedy, J. and Eberhart, R. (1995) Particle Swarm Optimization. *Proceedings of ICNN'95—International Conference on Neural Networks*, Volume 4, 1942-1948. <https://doi.org/10.1109/icnn.1995.488968>
- [7] Mirjalili, S. and Lewis, A. (2016) The Whale Optimization Algorithm. *Advances in Engineering Software*, **95**, 51-67. <https://doi.org/10.1016/j.advengsoft.2016.01.008>
- [8] Pan, W. (2012) A New Fruit Fly Optimization Algorithm: Taking the Financial Distress Model as an Example. *Knowledge-Based Systems*, **26**, 69-74. <https://doi.org/10.1016/j.knosys.2011.07.001>
- [9] Heidari, A.A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M. and Chen, H. (2019) Harris Hawks Optimization: Algorithm and Applications. *Future Generation Computer Systems*, **97**, 849-872. <https://doi.org/10.1016/j.future.2019.02.028>
- [10] Li, S., Chen, H., Wang, M., Heidari, A.A. and Mirjalili, S. (2020) Slime Mould Algorithm: A New Method for Stochastic Optimization. *Future Generation Computer Systems*, **111**, 300-323. <https://doi.org/10.1016/j.future.2020.03.055>
- [11] Lian, J., Hui, G., Ma, L., Zhu, T., Wu, X., Heidari, A.A., et al. (2024) Parrot Optimizer: Algorithm and Applications to Medical Problems. *Computers in Biology and Medicine*, **172**, Article ID: 108064. <https://doi.org/10.1016/j.compbiomed.2024.108064>
- [12] Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H. and Mirjalili, S.M. (2017) Salp Swarm Algorithm: A Bio-Inspired Optimizer for Engineering Design Problems. *Advances in Engineering Software*, **114**, 163-191. <https://doi.org/10.1016/j.advengsoft.2017.07.002>
- [13] Yang, Y., Chen, H., Heidari, A.A. and Gandomi, A.H. (2021) Hunger Games Search: Visions, Conception, Implementation, Deep Analysis, Perspectives, and towards Performance Shifts. *Expert Systems with Applications*, **177**, Article ID: 114864. <https://doi.org/10.1016/j.eswa.2021.114864>
- [14] He, S., Wu, Q.H. and Saunders, J.R. (2006) A Novel Group Search Optimizer Inspired by Animal Behavioural Ecology. *2006 IEEE International Conference on Evolutionary Computation*, Vancouver, 16-21 July 2006, 1272-1278.
- [15] Zhu, D., Wang, S., Zhou, C., Yan, S. and Xue, J. (2024) Human Memory Optimization Algorithm: A Memory-Inspired Optimizer for Global Optimization Problems. *Expert Systems with Applications*, **237**, Article ID: 121597. <https://doi.org/10.1016/j.eswa.2023.121597>
- [16] Su, H., Zhao, D., Heidari, A.A., Liu, L., Zhang, X., Mafarja, M., et al. (2023) RIME: A Physics-Based Optimization. *Neurocomputing*, **532**, 183-214. <https://doi.org/10.1016/j.neucom.2023.02.010>
- [17] Eskandar, H., Sadollah, A., Bahreininejad, A. and Hamdi, M. (2012) Water Cycle Algorithm—A Novel Metaheuristic Optimization Method for Solving Constrained Engineering Optimization Problems. *Computers & Structures*, **110**, 151-166. <https://doi.org/10.1016/j.compstruc.2012.07.010>

-
- [18] Cymerys, K. and Oszust, M. (2024) Attraction-Repulsion Optimization Algorithm for Global Optimization Problems. *Swarm and Evolutionary Computation*, **84**, Article ID: 101459. <https://doi.org/10.1016/j.swevo.2023.101459>
- [19] Abdel-Basset, M., Mohamed, R. and Abouhawwash, M. (2024) Crested Porcupine Optimizer: A New Nature-Inspired Metaheuristic. *Knowledge-Based Systems*, **284**, Article ID: 111257. <https://doi.org/10.1016/j.knosys.2023.111257>
- [20] Derrac, J., García, S., Molina, D. and Herrera, F. (2011) A Practical Tutorial on the Use of Nonparametric Statistical Tests as a Methodology for Comparing Evolutionary and Swarm Intelligence Algorithms. *Swarm and Evolutionary Computation*, **1**, 3-18. <https://doi.org/10.1016/j.swevo.2011.02.002>
- [21] Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L. and Herrera, F. (2011) KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework. *Journal of Multiple-Valued Logic and Soft Computing*, **17**, 255-287.
- [22] Jia, D., Zheng, G. and Khurram Khan, M. (2011) An Effective Memetic Differential Evolution Algorithm Based on Chaotic Local Search. *Information Sciences*, **181**, 3175-3187. <https://doi.org/10.1016/j.ins.2011.03.018>
- [23] Biswas, P.P., Suganthan, P.N. and Amarapura, G.A.J. (2017) Optimal Placement of Wind Turbines in a Windfarm Using L-SHADE Algorithm. 2017 *IEEE Congress on Evolutionary Computation (CEC)*, Donostia, 5-8 June 2017, 83-88. <https://doi.org/10.1109/cec.2017.7969299>
- [24] Li, Y., Feng, J. and Hu, J. (2016) Covariance and Crossover Matrix Guided Differential Evolution for Global Numerical Optimization. *SpringerPlus*, **5**, Article No. 1176. <https://doi.org/10.1186/s40064-016-2838-5>
- [25] Liang, J.J., Qin, A.K., Suganthan, P.N. and Baskar, S. (2006) Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions. *IEEE Transactions on Evolutionary Computation*, **10**, 281-295. <https://doi.org/10.1109/tevc.2005.857610>
- [26] Peng, L., He, C., Heidari, A.A., Zhang, Q., Chen, H., Liang, G., et al. (2022) Information Sharing Search Boosted Whale Optimizer with Nelder-Mead Simplex for Parameter Estimation of Photovoltaic Models. *Energy Conversion and Management*, **270**, Article ID: 116246. <https://doi.org/10.1016/j.enconman.2022.116246>
- [27] Xing, J., Heidari, A.A., Chen, H. and Zhao, H. (2024) WHRIME: A Weight-Based Recursive Hierarchical RIME Optimizer for Breast Cancer Histopathology Image Segmentation. *Displays*, **82**, Article ID: 102648. <https://doi.org/10.1016/j.displa.2024.102648>
- [28] Liu, L., Zhao, D., Yu, F., Heidari, A.A., Li, C., Ouyang, J., et al. (2021) Ant Colony Optimization with Cauchy and Greedy Levy Mutations for Multilevel COVID 19 X-Ray Image Segmentation. *Computers in Biology and Medicine*, **136**, Article ID: 104609. <https://doi.org/10.1016/j.compbiomed.2021.104609>
- [29] Civicioglu, P., Besdok, E., Gunen, M.A. and Atasever, U.H. (2018) Weighted Differential Evolution Algorithm for Numerical Function Optimization: A Comparative Study with Cuckoo Search, Artificial Bee Colony, Adaptive Differential Evolution, and Backtracking Search Optimization Algorithms. *Neural Computing and Applications*, **32**, 3923-3937. <https://doi.org/10.1007/s00521-018-3822-5>
- [30] Awad, N.H., Ali, M.Z. and Suganthan, P.N. (2017) Ensemble Sinusoidal Differential Covariance Matrix Adaptation with Euclidean Neighborhood for Solving CEC2017 Benchmark Problems. 2017 *IEEE Congress on Evolutionary Computation (CEC)*, Donostia, 5-8 June 2017, 372-379. <https://doi.org/10.1109/cec.2017.7969336>
- [31] Ling, Y., Zhou, Y. and Luo, Q. (2017) Lévy Flight Trajectory-Based Whale Optimization Algorithm for Global Optimization. *IEEE Access*, **5**, 6168-6186. <https://doi.org/10.1109/access.2017.2695498>