

多策略改进蜣螂优化算法无人机三维路径规划

王 雨, 侯恩广*

山东交通学院轨道交通学院, 山东 济南

收稿日期: 2025年11月28日; 录用日期: 2025年12月30日; 发布日期: 2026年1月7日

摘 要

无人机三维路径规划具有高维优化特性, 需要强大的全局搜索能力。为了解决传统DBO算法的过早收敛问题, 提高局部搜索效率, 本研究提出了一种多策略融合算法IDBO。该方法首先利用邻域拓扑机制来增强种群间的信息交流, 保持多样性, 并结合三重自适应权重、振荡增强的社会认知动态平衡机制。其次, 引入PSO分阶段引导滚球机制, 以达到更好的收敛精度。然后通过精英-差分协同觅食机制显著提高全局探索能力。通过CEC2017的29个基准函数验证, IDBO表现出优越的收敛精度和鲁棒性。在UAV3D路径规划实验进一步证明了IDBO生成更平滑路径的能力, 这些发现证实了IDBO通过协调多策略优化解决复杂场景下无人机路径规划挑战的有效性。

关键词

邻域拓扑, 粒子群优化, 精英差分协同, 动态自适应

Multi-Strategy Improved Dung Beetle Optimization Algorithm for UAV 3D Path Planning

Yu Wang, Enguang Hou*

School of Rail Transportation, Shandong Jiaotong University, Jinan Shandong

Received: November 28, 2025; accepted: December 30, 2025; published: January 7, 2026

Abstract

UAV 3D path planning exhibits high-dimensional optimization characteristics, requiring robust global search capabilities. To address the premature convergence problem of traditional Direct Optimization (DBO) algorithms and improve local search efficiency, this study proposes a multi-strategy

*通讯作者。

文章引用: 王雨, 侯恩广. 多策略改进蜣螂优化算法无人机三维路径规划[J]. 人工智能与机器人研究, 2026, 15(1): 123-137. DOI: 10.12677/airr.2026.151013

fusion algorithm, IDBO. This method first utilizes a neighborhood topology mechanism to enhance inter-population communication and maintain diversity, combined with a triple adaptive weighting and oscillatory enhancement social cognitive dynamic balance mechanism. Secondly, a phased guided rolling ball mechanism (PSO) is introduced to achieve better convergence accuracy. Then, an elite-differential cooperative foraging mechanism significantly improves global exploration capabilities. Validated by 29 benchmark functions at CEC2017, IDBO demonstrates superior convergence accuracy and robustness. Further experiments in UAV 3D path planning demonstrate IDBO's ability to generate smoother paths. These findings confirm the effectiveness of IDBO in addressing the challenges of UAV path planning in complex scenarios through coordinated multi-strategy optimization.

Keywords

Neighborhood Topology, Particle Swarm Optimization, Elite Differential Collaboration, Dynamic Adaptation

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

随着科技进步和复杂环境探索需求增加, 无人机在三维地形路径规划成为研究热点。无人机因其灵活性、高效性, 广泛应用于侦察监视、城市物流[1]、电力巡检、火灾救援等领域。面对复杂地形障碍, 如山峰、峡谷、建筑物等, 路径规划算法需精准模拟和适应环境, 确保无人机安全、高效完成任务。

无人机路径规划是实现自主飞行的关键技术, 旨在确定从起点到目标的最佳航迹。根据其核心思想与方法, 现有算法可划分为三大类: 传统路径规划算法、基于采样的路径规划算法以及智能优化算法。传统路径规划算法: 以 A* [2]、Dijkstra [3] 为代表, 基于图搜索或网格搜索, 理论成熟, 在静态、结构化环境中表现稳定, 路径可解释性强。然而其在高维复杂或动态环境中的计算效率与全局寻优能力受限, 易陷入局部最优。

基于采样的路径规划算法: 如 RRT [4] 及其变体(RRT* [5]、Informed RRT* [6])以及 PRM [7], 通过在自由空间中随机采样构建路径, 尤其适用于高维空间与复杂障碍物环境。这类算法具有较强的探索性, 但路径质量的渐近最优性需大量采样保证, 且完备性与实时性仍有提升空间。

智能优化算法: 模拟自然界现象进行迭代寻优。其显著特点是无需精确模型, 对非线性、多约束问题具有强大的求解能力, 并能有效跳出局部最优。如遗传算法[8]、粒子群优化[9]、蚁群算法[10]、蜣螂优化算法[11]等元启发式方法。其中 DBO 算法凭借其独特的生物行为模拟机制和并行架构, 在解决现代复杂优化问题时展现出明显的性能优势, 但是仍需改进。

为此众多学者通过改进智能优化算法来用于无人机的三位路径规划。蒋翱徽等[12]引入 Bernoulli 混沌映射、可变螺旋搜索策略、新型惯性权重和 Levy 飞行策略改进的蜣螂优化算法叶明君等[13]引入拉丁超立方采样初始化策略、平均差分变异策略改进蜣螂优化算法。刘春玲等[14]引入早熟判定机制和位置突变机制改进粒子群算法用于无人机路径规划。闫刚[15]等通过引入粒子变异策略和压缩因子改进的粒子群算法。

本文针对蜣螂优化算法早熟收敛、种群多样性不足及探索 - 开发失衡问题, 本研究提出多策略改进滚球优化算法(IDBO)。算法融合粒子群优化(PSO)机制, 通过三大创新提升性能: 1) 动态邻域拓扑与三

重自适应权重机制：通过动态调节搜索步长与交互范围，有效协调全局探索与局部开发的矛盾，解决种群多样性不足及探索 - 开发失衡问题；2) PSO 分阶段引导机制：引入速度矢量与社会认知，优化粒子运动轨迹，克服搜索盲目性并提升收敛速度与精度；3) 精英 - 差分协同觅食机制：利用变异扰动策略增强局部逃逸能力，避免陷入局部极值，改善早熟收敛难题。

2. 改进蜣螂算法

2.1. 环形邻域拓扑与三重自适应权重机制

传统的优化算法通常直接依赖全局最优引导，这在搜索初期易因过早收敛而限制探索范围，在后期则倾向于陷入局部最优。IDBO 通过引入固定邻域大小的拓扑结构，根据粒子间的相对位置和适应度构建“邻居圈”，初期利用固定的较宽邻域增强全局探索多样性，避免被少数极值过早吸引；后期通过邻域信息支持局部精化搜索。周期性更新邻域的机制帮助算法适应不同阶段需求，平衡探索与开发能力，从而提升整体优化性能。

动态邻域拓扑设计为：

第 i 个个体的邻域集合 \mathcal{N}_i 为：

$$\mathcal{N}_i = \{j \mid \max(1, i - \lfloor N_s/2 \rfloor) \leq j \leq \min(N, i + \lceil N_s/2 \rceil)\} \quad (1)$$

其中， $N_s = 10$ 。若 $|\mathcal{N}_i| < N_s$ ，通过循环补齐扩展：

$$\mathcal{N}_i \leftarrow \mathcal{N}_i \cup \{1, \dots, N_s - |\mathcal{N}_i|\} \text{ or } \{N - (N_s - |\mathcal{N}_i|) + 1, \dots, N\} \quad (2)$$

每个 5 代更新邻域，邻域适应度集合为

$$\mathcal{F}_i = \{f(\mathbf{x}_j) \mid j \in \mathcal{N}_i\}, \mathbf{x}_j \in \mathbb{R}^D \quad (3)$$

邻域最优：

$$\mathbf{x}_i^{\text{nBest}} = \arg \min_{j \in \mathcal{N}_i} f(\mathbf{x}_j) \quad (4)$$

三重自适应机制：

速度惯性权重 α 随迭代次数线性递减，确保算法在初期具有较大的探索步长，后期则逐步减小步长以利于局部开发。

$$\alpha = \alpha_0 \cdot (1 - t/T_{\max}) \quad (5)$$

认知学习权重 β 采用指数衰减形式，使粒子在初期更侧重向自身历史最优学习，保持个体多样性，随着迭代深入，认知影响逐渐减弱。

$$\beta = \beta_0 \cdot \exp\left(-\frac{t}{T_{\max}/5}\right) \quad (6)$$

社会学习权重 γ 引入振荡机制(正弦函数调节)，使其在不同迭代阶段动态波动。这种振荡不仅有助于跳出局部最优，还能在算法陷入停滞时提供额外的扰动，激发新的搜索方向，增强全局寻优能力。

$$\gamma = \gamma_0 \left(1 + \sin\left(\frac{2\pi t}{T_{\max}}\right)\right) \quad (7)$$

其中： T_{\max} 为最大迭代次数

2.2. PSO 分阶段引导滚球机制

传统 DBO 算法在滚球阶段暴露三重缺陷：首先，过度依赖单一全局最优位置(gBest_pos)，忽略邻域内高质量个体信息，导致协作搜索能力严重削弱；其次，固定扰动模式限制种群多样性生成；第三，缺乏分阶段平衡机制，造成早期探索不足或后期开发缓慢。为克服这些局限，本研究提出动态邻域拓扑与 PSO 分阶段引导机制：通过动态邻域与 PSO 社会引导的分阶段策略(初期邻域主导、后期全局主导)协同作用，不仅放大种群多样性，还实现探索向开发的平滑过渡，提升整体搜索效率。滚球动力学模型为：

$$k(t) = 0.7 + 0.3 \cdot (t/T_{\max}) \quad (8)$$

$$V_{IGR} = k(t) \cdot (g_{best} - x_i) + (1 - k(t)) \cdot R_{perturb} \quad (9)$$

其中： g_{best} 是截至迭代 t 时的全局最佳位置， x_i 为当前位置 $R_{perturb}$ 是一个随机扰动向量，通常由标准正态分布生成。IDBO 对滚球蜣螂的运动机制进行了深度优化。传统的滚球行为可能缺乏有效的全局引导或局部精调。算法引入了一个动态平衡系数 $k(t)$ ，根据迭代进程调节全局引导方向和随机扰动的比重。在搜索初期， $k(t)$ 值较小，更多地依赖随机扰动来扩大搜索范围；随着迭代进行 $k(t)$ 值逐渐增大，加强向全局最优方向的收敛，从而实现探索与开发的平滑过渡。

$$V_i(t+1) = \alpha \cdot V_{IGR}(t) + \beta \cdot r_p (p_{best} - x_i) + \gamma \cdot r_g (x_i^{nBest} - x_i) \quad (10)$$

其中： $V_i(t)$ 为第 i 个个体在迭代 t 时的速度向量

社会引导的选择：

$$\mathbf{x}_{social} = \begin{cases} \mathbf{x}_{nBest,i}, & t \leq 0.5T_{\max} \\ \mathbf{x}_{gBest}, & t > 0.5T_{\max} \end{cases} \quad (11)$$

随机系数采用混合策略：

$$r_p, r_g = \begin{cases} \min(\max(\mathcal{N}(0,1), -1), 1), & t \leq 0.5T_{\max} \\ U(0,1), & t > 0.5T_{\max} \end{cases}$$

其中 $\mathcal{N}(0,1)$ 为标准正态分布， $U(0,1)$ 为均匀分布。

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + V_i(t+1) \quad (12)$$

2.3. 精英差分引导协同觅食机制

其中，惯性引导滚动(IGR)阶段产生的精英个体(滚球者)构成了种群进化的“引领核心”，而精英-差分觅食策略则作为“探索前沿”，核心负责将“引领核心”的方向性信息与种群的多样性信息相融合，是实现探索与开发动态平衡的关键环节。

该策略的数学模型如下所述。在每次迭代 t 中，适应度居中的个体被赋予“觅食者”(Foragers)角色，其位置更新依赖于一个融合了开发性与探索性信息的混合梯度：

觅食粒子(占种群比例 30)通过精英引导与差扰动动的协同作用更新位置。从滚球粒子中随机选择一个精英引导者(X_l)同时，从种群中随机选择两个不同粒子(索引为 $r_1, r_2 \neq i$)，计算差分向量：

$$\mathbf{D} = \mathbf{X}_{r_1} - \mathbf{X}_{r_2} \quad (13)$$

$$\mathbf{X}_i(t+1) = \mathbf{X}_i(t) + \omega_e(t) \cdot \mathbf{R} \cdot (\mathbf{X}_l - \mathbf{X}_i(t)) + \omega_d(t) \cdot \mathbf{R} \cdot \mathbf{D} \quad (14)$$

其中： $\mathbf{R} \in [0,1]$ ：均匀分布随机数

权重系数的自适应策略设计为:

$$\omega_e(t) = 0.3 + 0.5 \cdot (t/T_{\max}) \quad (15)$$

$$\omega_d(t) = 0.7 - 0.5 \cdot (t/T_{\max}) \quad (16)$$

信息流的双向闭环: IGR 阶段产生的精英个体为觅食者提供了高质量的进化方向。反过来, 觅食者通过差分扰动探索的新区域有可能发现更优的解, 从而在后续迭代中进入精英集合, 甚至取代原有的精英个体, 形成了一个“精英引导探索→探索发现更优精英→新精英引导新一轮探索”的正反馈闭环。功能上的互补增强: IGR 机制确保了种群核心的快速收敛性, 而觅食策略则通过引入随机差分扰动, 有效避免了种群因过度追随精英而陷入局部最优。两者相辅相成, 前者“聚焦”, 后者“扩野”, 共同保证了算法同时具备强大的收敛速度和全局逃逸能力。

2.4. 繁殖蜣螂与偷窃蜣螂

本文保留了原始 DBO 算法的繁殖机制与偷窃机制, 其数学模型如下繁殖蜣螂以全局最优位置为基础, 通过高斯扰动构建动态产卵中心:

$$\mathbf{x}_{\text{brood}} = \mathbf{x}_{\text{gBest}} + \sigma \cdot \mathcal{N}(0, \mathbf{I}) \quad (15)$$

其中 $\sigma = 0.1$ 为扰动强度系数, $\mathcal{N}(0, \mathbf{I})$ 为标准高斯分布随机矩阵。产卵中心需满足边界约束:

$$\mathbf{x}_{\text{brood}} = \min(\max(\mathbf{x}_{\text{brood}}, \text{LB}), \text{UB}) \quad (16)$$

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \eta_1(\mathbf{x}_{\text{brood}} - \mathbf{x}_i(t)) + \eta_2(\mathbf{x}_{\text{gBest}} - \mathbf{x}_i(t)) \quad (17)$$

偷窃蜣螂:

$$\mathbf{x}_i(t+1) = \mathbf{x}_{\text{gBest}} + \lambda \cdot \mathcal{N}(0, 1) \cdot |\mathbf{x}_i(t) - \mathbf{x}_{\text{gBest}}| \quad (18)$$

其中 $\lambda = 0.9$ 为掠夺强度系数, 控制更新步长。

在改进的觅食策略增强型粪甲虫优化算法(IDBO)中, 繁殖行为与偷窃行为通过与觅食蜣螂和滚球蜣螂的协同作用, 显著提升了算法的性能。其协同机制主要体现在以下几个方面:

首先, 稳定性保障: 繁殖行为通过在全局最优解附近生成多样化后代, 提供稳定的局部开发能力, 弥补觅食蜣螂和滚球蜣螂在全局探索中的波动性; 偷窃行为则直接利用全局最优信息, 快速锁定优质区域, 与觅食蜣螂的精英-差分探索和滚球蜣螂的全局引导形成互补, 确保算法在探索与开发之间的平衡。

其次, 种群多样性维持: 繁殖行为通过高斯扰动在最优解周围生成多样化解, 为觅食蜣螂的差分扰动提供多样性支持; 偷窃行为通过自适应调整搜索范围(基于与最优解的距离), 进一步增强种群多样性。这种机制与觅食蜣螂的精英引导协同, 防止种群过早收敛, 提升算法对复杂搜索空间的适应性。

此外, 收敛性辅助: 繁殖行为在算法后期加强对优质区域的精细搜索, 与觅食蜣螂的差分扰动共同提升收敛精度; 偷窃行为则通过利用全局最优信息增强局部开发强度, 与动态邻域拓扑形成双重自适应机制。这种多角色协作有效平衡了觅食蜣螂的探索能力与繁殖行为的开发能力。

综上, 繁殖蜣螂与偷窃蜣螂通过与觅食蜣螂和滚球蜣螂的协同作用, 构建了“全局探索-局部开发-多样性维持”的闭环框架, 显著提升了 IDBO 算法的鲁棒性、收敛精度和多样性, 适用于复杂优化问题。流程见表 1。

3. 消融实验

为了评估新提出的 IDBO 算法的有效性, 使用 CEC2017 测试函数集(Dim = 30)运行进行测试。CEC

系列包括各种基本测试函数, 不仅可以作为比较各种优化算法性能的基准, 还可以作为模拟现实世界问题复杂性的工具。

Table 1. Process flow of the improved dung beetle optimization algorithm
表 1. 改进蜣螂优化算法流程

Algorithm 1. 改进的觅食策略增强型粪甲虫优化算法(IDBO)				
Require: 种群大小 N , 最大迭代次数 T_{\max} , 维度 D , 搜索空间边界 LB, UB, 目标函数 $fobj$				
Ensure: 全局最优位置 \mathbf{x}_{gBest} , 最优适应值 $fgBest$				
1: 定义最大迭代次数 $T_{\max} = 100$, 维度 D , 种群大小 $N = 50$				
2: 随机初始化种群位置 X , 速度 V_i , 计算个体适应值				
3: 初始化个体最优 $pBest$, 全局最优 $gBest$, 邻域拓扑 $N_s = 10$				
4: for 迭代 $t = 1$ 到 T_{\max} do				
5: 更新自适应权重 α, β, γ				
6: 计算邻域最优位置 \mathbf{x}^{nBest}				
7: 更新滚球甲虫(30%优秀个体)位置: 若 $t < 0.5T_{\max}$, 使用邻域最优和标准正态分布随机系数; 否则使用全局最优和均随机系数融合全局引导、扰动、认知和社会部分				
8: 更新繁殖甲虫(20%较差个体)位置: 基于全局最优中心加扰动				
9: 更新觅食甲虫(剩余个体)位置: 结合精英引导和差分扰动				
10: 更新偷窃甲虫(20%优秀个体)位置: 靠近全局最优加扰动				
11: 更新每个甲虫位置, 约束边界, 计算适应值				
12: 更新个体最优 $pBest$ 和全局最优 $gBest$				
13: 每 5 次迭代更新邻域拓扑				
14: 记录当前最优适应值				
15: 检查迭代次数: 若达到 T_{\max} , 停止并返回 Bestpos; 否则继续步骤 4				
16: end for				

Table 2. CEC2017 30-dimensional test comparison
表 2. CEC2017 30 维度测试对比

ID	Metric	原始 DBO	IGR-DBO	IDBO
CEC2017-F1	Std	1.1430e+09	3.1196e+04	6.4365e+03
	Ave	1.3303e+09	3.1481e+04	6.8694e+03
CEC2017-F3	Std	2.6554e+04	2.0146e+04	9.7926e+03
	Ave	1.3519e+05	6.2972e+04	3.2424e+04
CEC2017-F4	Std	1.2818e+02	2.5778e+01	4.0650e+01
	Ave	7.0351e+02	5.2309e+02	5.1662e+02
CEC2017-F5	Std	4.4939e+01	4.4734e+01	3.4341e+01
	Ave	9.1094e+02	6.9209e+02	6.4976e+02
CEC2017-F6	Std	1.6993e+01	9.5612e+00	7.5768e+00

续表

	Ave	6.8418e+02	6.4847e+02	6.2730e+02
CEC2017-F7	Std	8.3612e+01	1.3768e+02	5.8119e+01
	Ave	1.2257e+03	1.1784e+03	9.5720e+02
CEC2017-F8	Std	4.6554e+01	3.3845e+01	2.1177e+01
	Ave	1.1829e+03	9.4554e+02	9.1547e+02
CEC2017-F9	Std	3.0159e+03	2.2799e+03	7.9128e+02
	Ave	1.6764e+04	5.6575e+03	2.8882e+03
CEC2017-F10	Std	2.5775e+02	3.4248e+02	7.9410e+02
	Ave	9.0110e+03	8.9291e+03	8.2769e+03
CEC2017-F11	Std	3.7550e+02	5.8945e+01	5.4210e+01
	Ave	1.7871e+03	1.2786e+03	1.2467e+03
CEC2017-F12	Std	1.1718e+08	1.9421e+06	2.4542e+06
	Ave	6.2762e+07	2.8542e+06	2.8670e+06
CEC2017-F13	Std	1.5172e+05	1.8576e+04	1.7327e+04
	Ave	1.6729e+05	2.5767e+04	2.0749e+04
CEC2017-F14	Std	1.0221e+06	1.1528e+05	4.5574e+04
	Ave	5.2959e+05	6.9748e+04	3.1588e+04
CEC2017-F15	Std	4.5786e+04	5.6682e+03	1.1329e+04
	Ave	5.3532e+04	6.8214e+03	1.0538e+04
CEC2017-F16	Std	3.3831e+02	4.8712e+02	2.6131e+02
	Ave	4.2311e+03	3.1222e+03	2.5991e+03
CEC2017-F17	Std	2.4286e+02	2.0301e+02	1.5739e+02
	Ave	2.7583e+03	2.4245e+03	2.0592e+03
CEC2017-F18	Std	5.4410e+06	7.6157e+05	3.3488e+05
	Ave	5.0241e+06	7.0396e+05	2.5877e+05
CEC2017-F19	Std	1.4108e+04	5.3936e+03	1.1185e+04
	Ave	1.4016e+04	7.2476e+03	1.0181e+04
CEC2017-F20	Std	1.9549e+02	3.0702e+02	1.7321e+02
	Ave	3.1071e+03	2.9764e+03	2.4549e+03
CEC2017-F21	Std	4.0574e+01	4.6258e+01	2.8114e+01
	Ave	2.6664e+03	2.4691e+03	2.4029e+03
CEC2017-F22	Std	2.2942e+03	3.9954e+03	1.5752e+03
	Ave	9.2004e+03	6.0082e+03	2.7090e+03
CEC2017-F23	Std	7.8050e+01	9.1234e+01	4.3480e+01
	Ave	3.0654e+03	2.9441e+03	2.8167e+03
CEC2017-F24	Std	6.7531e+01	9.7013e+01	6.3271e+01
	Ave	3.2154e+03	3.0903e+03	2.9808e+03
CEC2017-F25	Std	5.7066e+01	2.4221e+01	2.6939e+01
	Ave	3.0239e+03	2.9291e+03	2.9367e+03

续表

CEC2017-F26	Std	1.5330e+03	1.3872e+03	1.1599e+03
	Ave	7.2020e+03	6.6359e+03	5.5002e+03
CEC2017-F27	Std	6.8794e+01	5.7042e+01	3.1372e+01
	Ave	3.3028e+03	3.3138e+03	3.2809e+03
CEC2017-F28	Std	1.5938e+02	2.6177e+01	4.1808e+01
	Ave	3.5225e+03	3.2900e+03	3.2881e+03
CEC2017-F29	Std	4.4917e+02	3.4701e+02	2.3180e+02
	Ave	4.8965e+03	3.4701e+02	4.0812e+03
CEC2017-F30	Std	2.8860e+06	3.8351e+04	2.4104e+04
	Ave	1.4542e+06	6.1220e+04	2.8396e+04

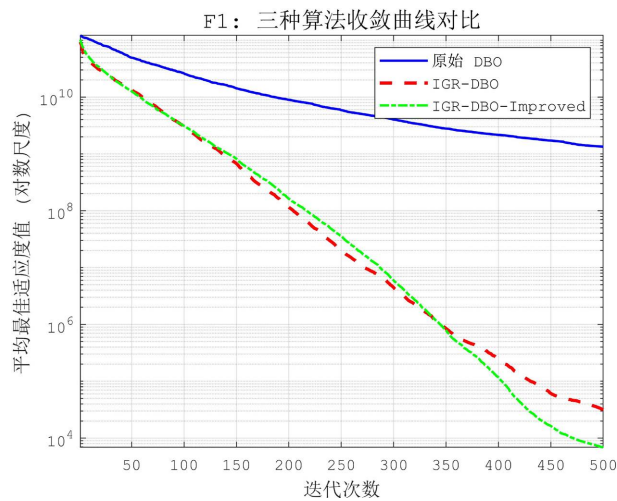


Figure 1. F1 test comparison
图 1. F1 测试对比

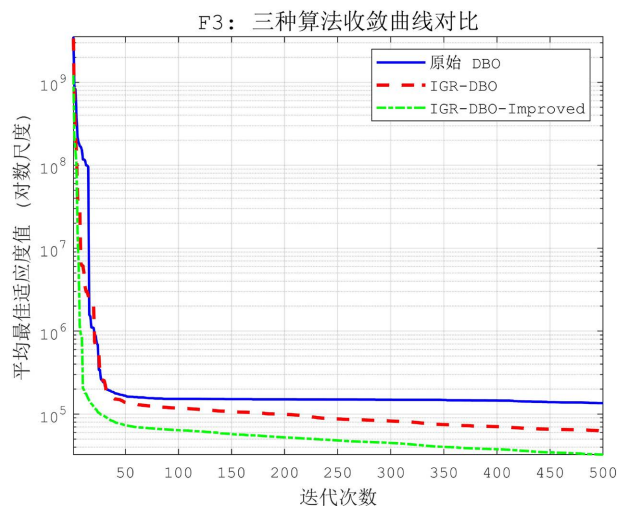


Figure 2. F3 test comparison
图 2. F3 测试对比

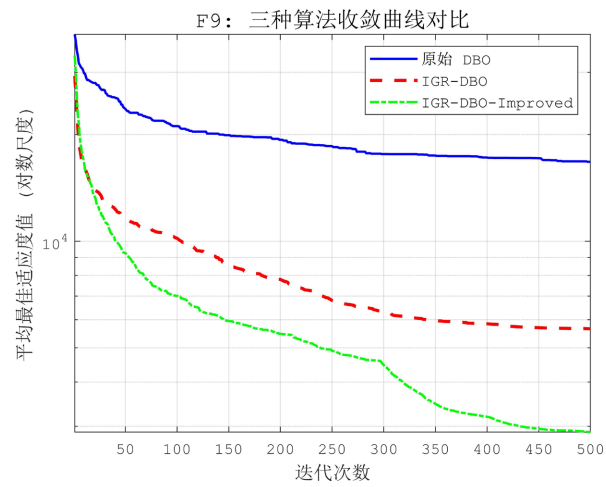


Figure 3. F9 test comparison
图 3. F9 测试对比

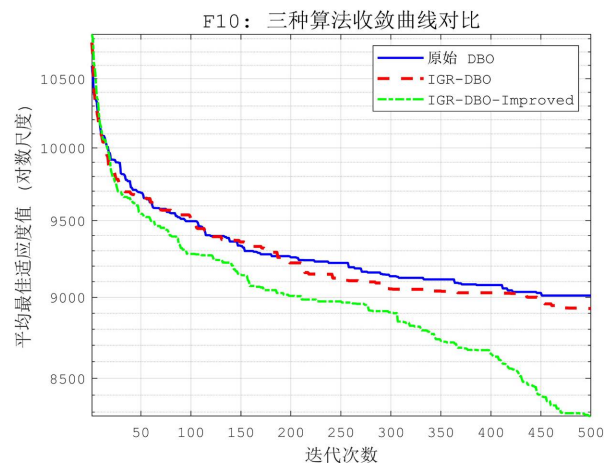


Figure 4. F10 test comparison
图 4. F10 测试对比

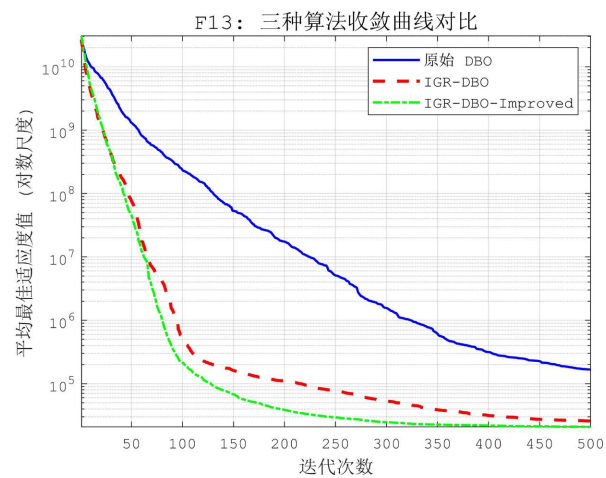


Figure 5. F13 test comparison
图 5. F13 测试对比

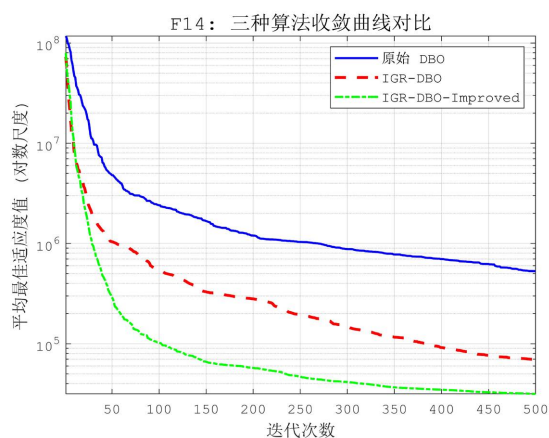


Figure 6. F14 test comparison

图 6. F14 测试对比

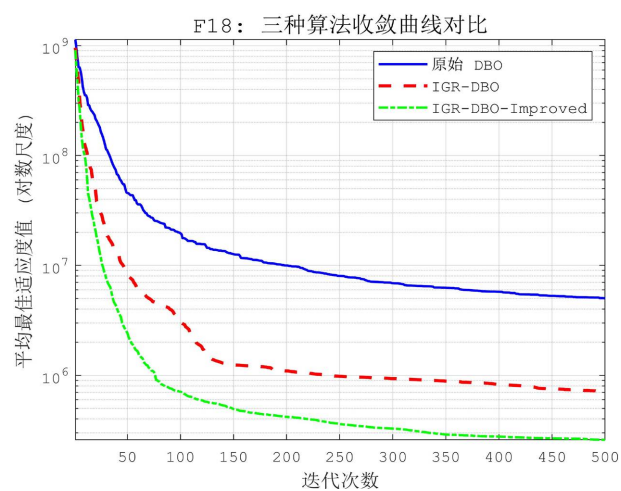


Figure 7. F18 test comparison

图 7. F18 测试对比

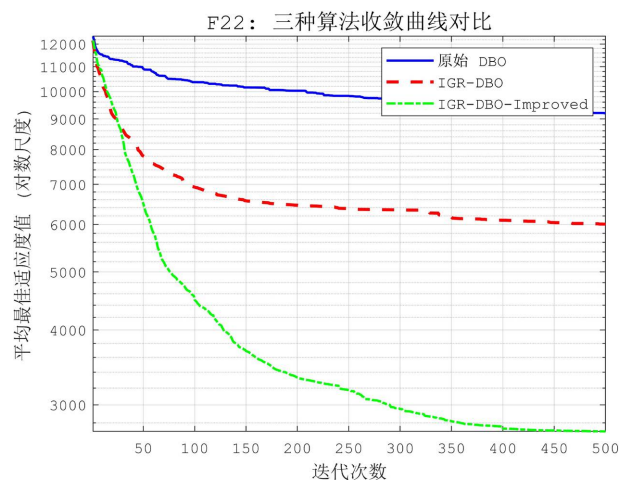


Figure 8. F22 test comparison

图 8. F22 测试对比

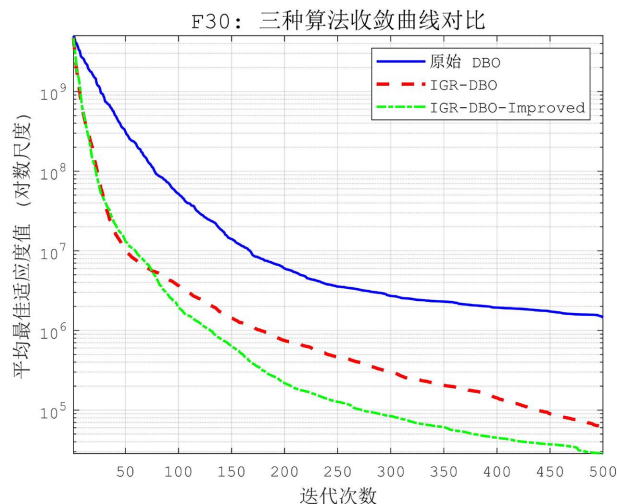


Figure 9. F30 test comparison

图 9. F30 测试对比

从表 2 数据和图 1~9 的可视化图来看, IDBO 在大多数 CEC-2017 基准函数上表现出最优或接近最优的性能, 其平均值和标准差均显著低于原始 DBO 和 IGR-DBO。例如, 在 CEC2017-F1 中, IDBO 的平均值为 $6.486\text{e}+03$, 标准差为 $6.436\text{e}+03$, 而原始 DBO 的平均值高达 $1.330\text{e}+09$, 标准差为 $1.143\text{e}+02$ 。这种差距表明 IDBO 在收敛精度和稳定性上具有显著优势。原始 DBO 作为基准算法, 在大多数函数上的表现相对较差, 尤其是在高维或复杂函数中。例如, CEC2017-F1 和 CEC2017-F12 的平均值分别为 $1.330\text{e}+09$ 和 $6.276\text{e}+07$, 标准差分别为 $1.143\text{e}+09$ 和 $1.172\text{e}+08$, 显示出较大的波动性和较低的收敛精度。这可能与原始 DBO 缺乏有效的全局搜索机制和种群多样性维护策略有关, 导致其容易陷入局部最优, 尤其在面对高维优化问题时表现不佳。

数据表明, IDBO 和 IGR-DBO 的改进策略对原始 DBO 的性能提升起到了关键作用。IDBO 的综合优化策略使其在收敛速度和精度上全面超越其他变体, 而 IGR-DBO 的局部改进策略则在特定函数上表现良好。这与前文提到的增强策略(如全局探索和种群更新)相符, 表明这些策略的有效性。然而, 原始 DBO 的局限性也提示, 单一算法框架在面对多样化优化问题时可能需要更多元化的改进。

4. 路径规划仿真

本节旨在探讨 IDBO 算法在实际场景中的应用潜力, 特别聚焦于其在无人机三维轨迹规划中的表现。为验证其效能, 我们对复杂山地地形下的无人机飞行路径进行了仿真实验。结果清晰展示了 IDBO 算法在处理高难度路径规划任务中的卓越能力, 特别是在复杂环境下的导航表现, 充分证明了其在该领域的实用性和可靠性。

4.1. 适应度函数

为了展示我们提出的算法的实际应用, 我们使用 Matlab 平台进行了三维无人机路径规划的仿真测试的场景。

路径规划的目标是在满足所有约束条件的前提下, 找到一条从起点到终点的最优无人机飞行路径。该路径需同时优化多个性能指标。为此, 我们设计了一个综合的适应度函数 $\Gamma(t)$, 用以评估任何候选路径的质量。该函数将路径长度、飞行平稳性和安全性约束统一在一个惩罚函数框架内。

候选路由一系列三维航路点 $\mathbf{P}_i = (x_i, y_i, z_i)$ 定义。为了确保路径的连续性与平滑性, 我们采用三次

样条插值(Cubic Spline Interpolation)对控制点进行插值, 生成一条连续可微的飞行轨迹 $\Gamma(t)$:

$$\Gamma(t) = (x(t), y(t), z(t)), t \in [0, 1] \quad (19)$$

其中, t 为归一化的路径参数。插值后, 我们对路径应用移动平均滤波器(Moving Average Filter)以进一步抑制高频抖动, 确保其符合无人机的动力学特性。

适应度函数 F 旨在最小化以下三个关键目标的加权和: 路径长度, 高度变化, 路径平滑度。

a) 路径长度:

最小化总飞行距离是提高效率的核心。路径长度由对轨迹进行线性积分求得:

$$J_{path} = \sum_{k=1}^{N-1} |\Gamma(t_{k+1}) - \Gamma(t_k)| = \sum_{k=1}^{N-1} \sqrt{\Delta x_k^2 + \Delta y_k^2 + \Delta z_k^2} \quad (20)$$

b) 高度变化

为促进无人机平稳飞行并减少能耗, 我们惩罚相对于平均高度的起伏。该目标通过路径点高度的标准差来实现:

$$\bar{z} = \frac{1}{N-2} \sum_{k=2}^{N-1} z_k, J_{height} = \sqrt{\frac{1}{N-2} \sum_{k=2}^{N-1} (z_k - \bar{z})^2} \quad (21)$$

c) 路径平滑度

剧烈的方向变化会导致无人机减速和能耗增加。我们采用曲率的平方和作为平滑度度量, 通过计算轨迹二阶差分(离散加速度)的模长来实现:

$$J_{smooth} = \sum_{k=2}^{N-1} (\|\Gamma''(t_k)\|)^2 \approx \sum_{k=2}^{N-1} (\Delta^2 x_k^2 + \Delta^2 y_k^2 + \Delta^2 z_k^2) \quad (22)$$

其中, $\Delta^2 x_k = x_{k-1} - 2x_k + x_{k+1}$ (对 y 和 z 同理) 该值越小, 表明路径的弯曲程度越低, 飞行越平滑。

d) 约束处理: 障碍物避障

路径必须与所有障碍物保持安全距离。对于第 j 个位于 (x_j, y_j) 且半径为 r_j 的圆柱形障碍物, 我们定义其 $violation v_j$ 为:

$$d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, v_{i,j} = \max(0, r_j - d_{i,j}) \quad (23)$$

该值在路径点 i 侵入障碍物 j 时为正, 否则为零。所有路径点对所有障碍物的总违反量 Φ 计算如下:

$$\Phi = \frac{1}{N} \sum_{j=1}^M \sum_{i=1}^N v_{i,j} \quad (24)$$

其中, M 是障碍物的数量, N 是路径点的数量。通过惩罚函数法, 我们将该约束违反量纳入目标函数。

e) 完整的适应度函数

最终的适应度函数 F 是上述三个目标的加权和, 加上约束违反的惩罚项:

$$F = \min \{ \omega_1 J_{path} + \omega_2 J_{height} + \omega_3 J_{smooth} + \rho \cdot \Phi \} \quad (25)$$

其中: $\omega_1, \omega_2, \omega_3$ 是权重系数, 在本研究中分别设定为 0.4, 0.2, 和 0.4, 以强调路径长度和平滑度。 ρ 是一个巨大的惩罚因子(通常为 10^3 至 10^6 量级), 以确保优化算法优先找到无碰撞的可行解。

4.2. 无人机三维路径规划仿真

为了展示我们提出的算法的实际应用, 我们使用 Matlab 平台进行了三维无人机路径规划的仿真测试的场景。本节通过高斯函数构建三维山峰地图, 用于模拟无人机路径规划中的复杂地形环境。公式中, 地形高度 $z(x, y)$ 由多个高斯函数叠加生成, 每个山峰由中心坐标、峰高和坡度参数定义, 以实现真实

地形的模拟建模:

$$z(x, y) = \sum_{i=1}^n h_i \exp \left[- \left(\frac{x - x_i}{x_{si}} \right)^2 - \left(\frac{y - y_i}{y_{si}} \right)^2 \right] \quad (26)$$

其中, n 表示山峰总个数, (x_i, y_i) 代表第 i 个山峰的中心坐标; h_i 为地形参数, 控制高度; x_{si} 和 y_{si} 分别是第 i 个山峰沿 x 轴和 y 轴方向的衰减量、控制坡度

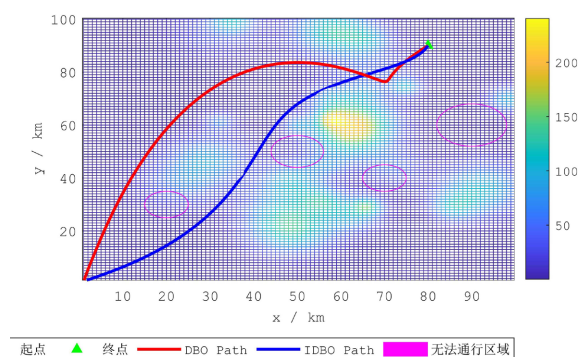


Figure 10. Comparison of 3D simulation paths

图 10. 三维仿真路径对比

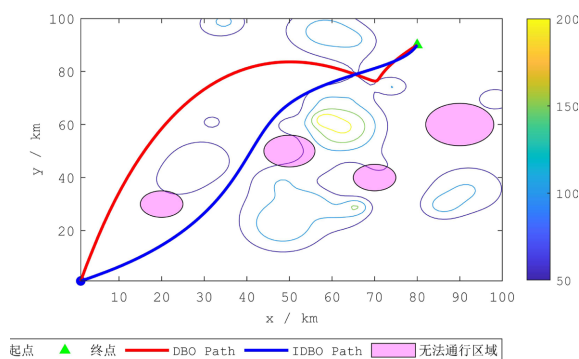


Figure 11. Top-down view comparison

图 11. 俯视图对比

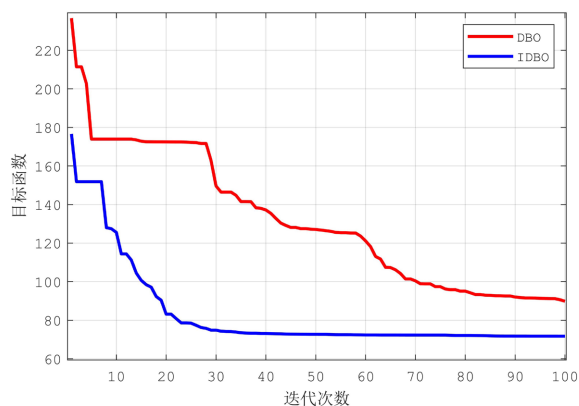


Figure 12. Top-down comparison of 2D grids

图 12. 二维网格俯视对比

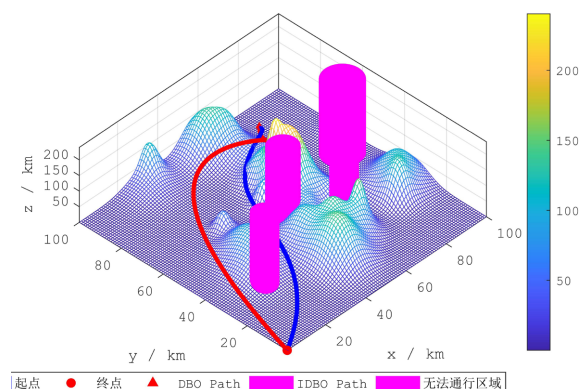


Figure 13. Comparison of fitness iteration curves
图 13. 适应度迭代曲线对比

5. 结论与分析

由图 10~13 可以看出 IDBO 算法在路径规划和优化问题中显示出了显著的优越性。通过对比 IDBO 和 DBO 算法在相同环境下的路径规划图, 我们可以观察到 IDBO 算法能够更有效地绕过障碍物, 找到更直接的路径。这不仅减少了路径长度, 还可能提高了到达目的地的效率。此外, 从性能对比图中可以看出, IDBO 算法在较少的迭代次数内就能达到更低的目标函数值, 并且在整个迭代过程中表现出更好的稳定性。这表明 IDBO 算法在收敛速度和稳定性方面都优于 DBO 算法, 使其成为解决优化问题时的一个更优选择。

综上所述, IDBO 算法在路径规划和优化问题中表现出了更高的效率和稳定性, 这使得它在实际应用中具有更大的潜力和价值。

基金项目

德州市科技型中小企业创新能力提升工程项目(2025DZTSGC013)。

参考文献

- [1] 祝文杰, 李维, 王子炎. 改进 A*算法的无人机城市低空物流路径规划[J]. 计算机工程与应用, 2025, 61(19): 302-310.
- [2] 曹红艳, 周宝, 李希彬, 等. 基于 A 星算法的分层多粒度无人机三维路径规划算法[J]. 湘潭大学学报(自然科学版), 2025, 47(3): 65-76.
- [3] Alshammrei, S., Boubaker, S. and Kolsi, L. (2022) Improved Dijkstra Algorithm for Mobile Robot Path Planning and Obstacle Avoidance. *Computers, Materials & Continua*, **72**, 5939-5954. <https://doi.org/10.32604/cmc.2022.028165>
- [4] 李金贺, 封岸松, 范晓亮, 等. 基于改进 RRT 算法的机器人路径规划[J]. 自动化技术与应用, 2025, 44(11): 7-10.
- [5] 杨彦新, 姜香菊. 优化改进 RRT*和 APF 的无人机航迹规划算法[J]. 电光与控制, 2025, 32(10): 34-40.
- [6] 张森, 庞岩, 周福亮. 基于改进 Informed-RRT*算法的无人机三维路径规划[J/OL]. 系统工程与电子技术: 1-9. <https://link.cnki.net/urlid/11.2422.TN.20251106.1956.056>, 2025-11-26.
- [7] Li, W., Wang, L., Zou, A., Cai, J., He, H. and Tan, T. (2022) Path Planning for UAV Based on Improved PRM. *Energies*, **15**, Article 7267. <https://doi.org/10.3390/en15197267>
- [8] 代进进, 戴治洪, 李季颖, 等. 基于改进遗传算法的多无人机协同搜索方法[J/OL]. 指挥控制与仿真: 1-7. <https://link.cnki.net/urlid/32.1759.tj.20251113.1131.002>, 2025-11-26.
- [9] 胡增明, 徐元龙, 李晓强, 等. 基于改进 PSO 算法的无人机三维路径规划应用研究[J]. 铁道勘察, 2025, 51(5): 7-11+17.
- [10] 刘抒桥. 基于改进蚁群算法的低成本多无人机路径规划[D]: [硕士学位论文]. 西安: 西安石油大学, 2025.

-
- [11] Xue, J. and Shen, B. (2022) Dung Beetle Optimizer: A New Meta-Heuristic Algorithm for Global Optimization. *The Journal of Supercomputing*, **79**, 7305-7336. <https://doi.org/10.1007/s11227-022-04959-6>
 - [12] 蒋翱徽, 刘文红. 基于改进蜣螂优化算法的无人机三维路径规划[J]. 电子测量技术, 2024, 47(13): 128-135.
 - [13] 叶明君, 王姝鉴. 基于改进蜣螂优化算法的无人机路径规划[J]. 计算机科学, 2025, 52(S1): 341-346.
 - [14] 刘春玲, 冯锦龙, 田玉琪, 等. 基于改进粒子群算法的无人机航迹规划[J]. 计算机仿真, 2023, 40(10): 38-43.
 - [15] 闫刚, 夏孝程, 李佳明. 基于改进粒子群算法的无人机复杂三维路径规划[J]. 自动化应用, 2025, 66(12): 38-42.