

DOI:10.19322/j.cnki.issn.1006-4710.2026.02.008

<https://xuebao.xaut.edu.cn>

引文格式: 张文宇, 高岳林, 郭灿. 基于自适应混合粒子群算法的无人机三维路径规划[J]. 西安理工大学学报, 2026, 42(2): 228-239.

Zhang Wenyu, Gao Yuelin, Guo Can. Three dimensional path planning for unmanned aerial vehicles by adaptive hybrid particle swarm algorithm[J]. Journal of Xi'an University of Technology, 2026, 42(2): 228-239.

基于自适应混合粒子群算法的无人机三维路径规划

张文宇¹, 高岳林², 郭 灿¹

(1. 北方民族大学 计算机科学与工程学院, 宁夏 银川 750021;

2. 北方民族大学 宁夏科学计算与智能信息处理协同创新中心, 宁夏 银川 750021)

摘要: 针对无人机(UAV)三维路径规划问题, 本文提出了一种自适应混合粒子群优化算法, 以解决传统启发式算法易陷入局部最优、搜索能力不佳以及收敛精度不高等问题。首先, 算法通过自适应调整惯性权重来平衡其全局搜索与局部搜索能力, 同时, 引入变异机制以增强搜索的多样性。其次, 在粒子位置和速度更新后, 通过结合人工蜂群算法的雇佣蜂策略, 提升了算法的局部搜索能力。最后, 在算法迭代后期, 利用混沌映射对种群进行随机干扰, 帮助算法加快收敛速度并跳出局部最优。将该自适应混合粒子群算法与其他启发式算法用于无人机三维路径规划仿真实验, 实验结果表明该算法在搜索精度和收敛速度等方面均表现优异, 验证了其有效性和可行性。

关键词: 无人机; 三维路径规划; 粒子群优化算法; 自适应惯性权重; 人工蜂群算法; 混沌映射

中图分类号: V249; TP301.6 **文献标志码:** A **文章编号:** 1006-4710(2026)02-0228-12

Three dimensional path planning for unmanned aerial vehicles by adaptive hybrid particle swarm algorithm

Zhang Wenyu¹, Gao Yuelin², Guo Can¹

(1. School of Computer Science and Engineering, North Minzu University, Yinchuan 750021, China;

2. Ningxia Collaborative Innovation Center for Scientific Computing and Intelligent Information Processing, North Minzu University, Yinchuan 750021, China)

Abstract: The article proposes an adaptive hybrid particle swarm optimization algorithm for the three-dimensional path planning of unmanned aerial vehicles (UAV) to solve the problems by traditional heuristic algorithms prone to getting stuck in local optima, poor search ability, and low convergence accuracy. Firstly, the algorithm balances its global and local search capabilities by adaptively adjusting inertia weights, while introducing mutation mechanisms for enhancing search diversity. Secondly, after updating the particle position and velocity, the hiring bee strategy combined with the artificial bee colony algorithm is used to enhance the algorithm's local search capability. Finally, in the later stage of algorithm iteration, chaotic mapping is used to randomly interfere with the population, helping the algorithm accelerate convergence speed and jump out of local optima. The adaptive hybrid particle swarm algorithm was used in simulation experiments by other heuristic algorithms for 3D path planning of unmanned aerial vehicles. The experimental results showed that the algorithm performed excellently in search accuracy and convergence speed, verifying its effectiveness and feasibility.

收稿日期: 2024-12-07; 网络首发日期: 2025-06-12

网络首发地址: <https://link.cnki.net/urlid/61.1294.n.20250612.1306.006>

基金项目: 宁夏自然科学基金重点资助项目(2022AAC02043)

第一作者: 张文宇, 男, 硕士生, 研究方向为智能优化算法与无人机路径规划。E-mail: 1062049747@qq.com

通信作者: 高岳林, 男, 博士, 教授, 研究方向为智能优化算法、智能计算与大数据处理。E-mail: gaoyuelin@263.net

Key words: UAV; 3D path planning; particle swarm optimization algorithm; adaptive inertia weight; artificial bee colony algorithm; chaotic mapping

近年来,无人机(unmanned aerial vehicle, UAV)因在军事和民用领域展现出广阔的应用前景,受到了众多学者的关注。得益于其成本低、响应灵活,无人机已被应用于多个领域,如电力巡逻检查、搜索和救援、灾害管理、目标搜索等^[1-2]。作为无人机路径系统的核心问题之一,路径规划对于无人机的自主飞行和任务执行至关重要,是提高无人机自主飞行能力、确保飞行安全的关键技术^[3]。路径规划综合考虑了无人机的机动性和环境约束,力图寻找一条从起始点到目标点的无碰撞最优路径,其本质是一类具有复杂约束的系统优化问题^[4],需要依托高效且稳定的算法予以求解。无人机路径规划示意图见图1。

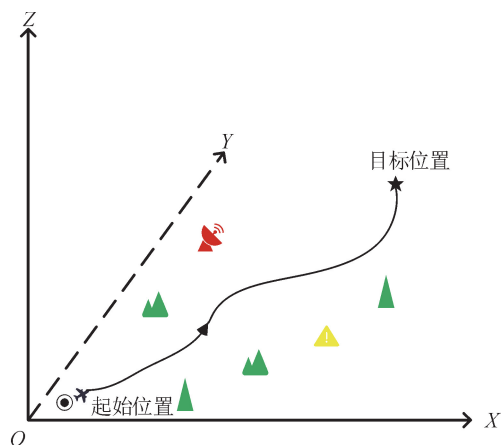


图1 无人机路径规划示意图

Fig.1 Schematic diagram of UAV path planning

国内外学者围绕无人机路径规划展开了深入研究。Qu等^[5]基于简化的灰狼优化器和改进的共生生物搜索提出了一种新的混合算法 HSGWO-MSOS,该算法通过将探索和开发能力有效结合,简化了灰狼算法(grey wolf optimizer, GWO)的相位,提高了收敛速度,保留了种群的探索能力。陈琦等^[6]将传统蜣螂算法(dung beetle optimizer, DBO)与灰狼算法结合,并引入混沌策略和莱维飞行策略,有效增强了算法的探索能力、提升了收敛速度。梅雨琳等^[7]对蜣螂优化算法进行了改进,通过混合混沌序列提升初始种群多样性,引入差分变异算子增强局部搜索能力,并结合改进的正弦算法提升了全局搜索能力。胡鑫宇等^[8]通过优化种群初始化、引入多维移动策略和非线性收敛因子,提高了鲸鱼优化算法的收敛速度和精度。Wang等^[9]通过动态调整进化算子的生成区域、限制新基因的产生范围,提

高了遗传算法的局部搜索能力和整体搜索效率。Tu等^[10]将差分进化算法与海洋捕食者算法相结合,应用于复杂场景下的无人机三维路径规划问题,有效避免了算法陷入局部最优。Chai等^[11]使用了多策略融合差分进化算法,该算法集成了多种群策略、新颖的自适应策略和交互式变异策略,以平衡算法的开发和探索能力。

粒子群算法(particle swarm optimization, PSO)因其特殊的搜索机制、良好的收敛性能及计算简便,被广泛应用于工程优化领域,但该算法同样存在一些缺陷,如容易陷入局部最优,收敛精度低、速度慢,种群多样性差等^[12]。而人工蜂群算法(artificial bee colony, ABC)可通过信息交换避免局部最优,仅需要少量参数即可运行,可应用于不同类型的优化问题^[13]。

当前,无人机的路径规划仍面临着效率不高、优化过程不稳定以及易陷入局部最优等问题,而这在复杂环境中表现的尤为突出。对此,本文首先以各类约束成本为优化目标,构建面向山区地形和动态环境的仿真平台,并建立相应数学模型;其次,为克服PSO易陷入局部最优的局限,本文通过融合人工蜂群算法中的雇佣蜂搜索阶段、采用自适应惯性权重、引入变异机制,同时在算法后期引入Logistic混沌映射,提出了一种自适应混合粒子群优化算法(adaptive hybrid artificial bee particle swarm optimization, AHAPSO),并将其应用于无人机的路径规划;最后,通过静态与动态仿真实验的对比分析,验证了所提算法的有效性 with 优越性。

1 无人机路径规划建模

1.1 环境建模

通过对无人机飞行环境的抽象化处理,构建了一个三维空间模型。为准确模拟山区地形的特点,采用指数函数来描述飞行环境中较高的天然山体,其数学模型为^[14]:

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

式中: n 为山峰总个数; (x_i, y_i) 为第 i 个山峰的中点坐标; h_i 为地形参数,用于控制高度; $x_{s,i}$ 和 $y_{s,i}$ 分别为第 i 个山峰沿 x 、 y 方向的衰减量、控制坡度。图2为山体地图模型。

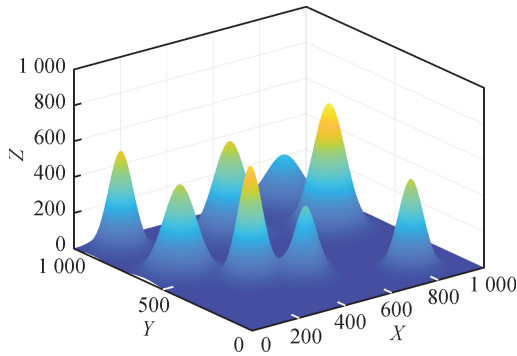


图 2 山体地图模型

Fig. 2 Mountain terrain model

1.2 约束成本建模

1.2.1 路径成本

通过累加路径相邻点之间的欧几里得距离来计算路径成本。假设路径由 (x_1, y_1, z_1) , $(x_2, y_2, z_2), \dots, (x_n, y_n, z_n)$ 这 n 个点组成, 路径长度的具体计算过程为:

1) 计算无人机沿路径的相邻坐标间的差值:

$$\begin{cases} dx_i = X_{i+1} - X_i \\ dy_i = Y_{i+1} - Y_i \\ dz_i = Z_{i+1} - Z_i \end{cases} \quad (2)$$

2) 根据欧几里得距离公式, 路径中相邻两点间的距离可表示为:

$$d(p_i, p_{i+1}) = \sqrt{(X_{i+1} - X_i)^2 + (Y_{i+1} - Y_i)^2 + (Z_{i+1} - Z_i)^2} \quad (3)$$

3) 计算每一段路径的距离并累加, 得到路径总长度:

$$f_1 = \sum_{i=1}^{n-1} d(p_i, p_{i+1}) \quad (4)$$

1.2.2 碰撞成本

无人机在飞行过程中可能会产生碰撞, 可通过检测路径点与地形之间的高度差来判断是否会发生碰撞。碰撞成本的计算公式为:

$$f_2 = \begin{cases} \infty, \Delta d \leq 0 \\ \sum_{i=2}^n \frac{c_{\text{punk}}}{\Delta d}, \Delta d > 0 \end{cases} \quad (5)$$

$$\Delta d = z_c(x_i, y_i) - z_i \quad (6)$$

式中: z_i 为路径点的高度; $z_c(x_i, y_i)$ 为地形高度; Δd 为路径点与地形之间的距离; c_{punk} 为碰撞惩罚系数。

当路径点高度与地形点高度交汇时, 碰撞成本为无穷大, 否则碰撞成本随着与地形间距离的增加而逐渐减少。

1.2.3 飞行高度成本

飞行高度成本用于惩罚路径中飞行高度不符合

安全要求的部分, 以确保路径点始终位于规定的安全高度范围内, 从而避免飞行过低或过高的情况^[15]。飞行高度成本的计算公式为:

$$z_{\min} + \Delta h \leq z_i < z_{\max} \quad (7)$$

$$f_3 = \sum_{i=1}^n \begin{cases} f_3 \times 10, \\ z_i < z_{\min} + \Delta h \text{ 或 } z_i > z_{\max} \\ f_3 + (z_i - z_{\min} - \Delta h) \times h_{\text{punk}}, \\ \text{其他} \end{cases} \quad (8)$$

式中: z_i 为当前路径点的高度; $z_{\min}$ 为无人机在该点的地面高度; $z_{\max}$ 为无人机允许飞行的最大高度; Δh 为无人机与地面保持的安全距离; h_{punk} 为飞行高度的惩罚系数。

1.2.4 路径平滑度成本

评估无人机路径的平滑度成本, 目的是确保路径平滑、没有明显的急转弯或剧烈的高度变化^[16]。本文通过计算路径的转向角(下标 turn)和爬升角(下标 climb)的变化来评估路径平滑度。具体计算公式为:

$$f_4 = \sum_{i=2}^{n-1} (p_{\text{turn}, i} + p_{\text{climb}, i}) \quad (9)$$

$$\begin{cases} p_{\text{turn}, i} = (\max(0, |\theta_{\text{turn}, i}| - \theta_{\max}))^2 \cdot \lambda_{\text{turn}} \\ p_{\text{climb}, i} = (\max(0, |\varphi_{\text{climb}, i}| - \varphi_{\max}))^2 \cdot \lambda_{\text{climb}} \end{cases} \quad (10)$$

$$\theta_{\text{turn}, i} = \arctan\left(\frac{\|\mathbf{p}^1 \times \mathbf{p}^2\|}{\mathbf{p}^1 \cdot \mathbf{p}^2}\right) \quad (11)$$

$$\varphi_{\text{climb}, i} = \arctan\left(\frac{p_z^2 - p_z^1}{\sqrt{(p_x^2 - p_x^1)^2 + (p_y^2 - p_y^1)^2}}\right) \quad (12)$$

式中: θ_{turn} 和 φ_{climb} 为转向角和爬升角; $\theta_{\max}$ 和 $\varphi_{\max}$ 为转向角和爬升角的最大限制角度; λ_{turn} 和 λ_{climb} 为转向角和爬升角的惩罚系数; $\mathbf{p}^1$ 和 $\mathbf{p}^2$ 为第 i 点前后的两个向量。

1.2.5 雷达威胁成本

雷达威胁成本用于评估路径中无人机与雷达威胁之间的接近程度, 其目的是避免无人机进入雷达的探测范围或接近雷达的探测区域, 从而增加路径的安全性。假设有多个雷达, 每个雷达的位置为 $r_{\text{adar}, j} = [x_j, y_j, z_j]$, 雷达的威胁成本为:

$$f_5 = \begin{cases} \infty, d_{i,j} < r_j \\ \sum_{i=1}^n \sum_{j=1}^m \frac{\omega_{\text{radar}}}{d_{i,j} - r_j}, r_j < d_{i,j} \leq r_j + \Delta s \\ 0, d_{i,j} > r_j + \Delta s \end{cases} \quad (13)$$

$$d_{i,j} = \sqrt{(X_i - x_j)^2 + (Y_i - y_j)^2 + (Z_i - z_j)^2} \quad (14)$$

式中: $d_{i,j}$ 为第 i 个路径点和第 j 个雷达之间的欧氏

距离; r_j 为雷达的探测半径; ω_{radar} 为雷达威胁系数; Δ_s 为安全缓冲区距离。

1.3 总成本函数建模

在考虑上述各项成本与路径成本的约束后,总的成本函数为:

$$F_{\text{it}} = \sum_{i=1}^5 b_i f_i \quad (15)$$

式中: b_i ($i = 1, 2, 3, 4, 5$) 为权重系数,且 $\sum_{i=1}^5 b_i = 1$; $f_1 \sim f_5$ 分别为路径成本、碰撞成本、飞行高度成本、路径平滑度成本和雷达威胁成本。

将总的成本函数作为无人机路径规划的适应度函数。

2 自适应混合粒子群算法

2.1 自适应混合粒子群算法

2.1.1 自适应惯性权重

粒子群算法^[17]由 Eberhart 和 Kennedy 于 1995 年提出,通过模拟鸟群觅食过程中的迁徙和群聚行为,形成了一种基于群体智能的随机搜索算法。该算法的核心思想是从包含若干个粒子的随机种群出发,通过预设的适应度函数来评估每个粒子的适应度,最终将粒子引向最优解区域。算法首先初始化一组粒子,每个粒子在搜索空间中都有一个随机生成的位置和速度,使用适应度函数对每个粒子的位置进行评估,然后更新每个粒子的速度和位置^[18]:

$$v_k(t+1) = \omega \times v_k(t) + c_1 \times r_1 \times (p_{\text{best},k} - x_k(t)) + c_2 \times r_2 \times (g_{\text{best}} - x_k(t)) \quad (16)$$

$$x_k(t+1) = x_k(t) + v_k(t+1) \quad (17)$$

式中: $x_k(t)$ 为第 k 个粒子在第 t 次迭代后的位置; $v_k(t)$ 为第 k 个粒子在第 t 次迭代中的速度; $p_{\text{best},k}$ 为第 k 个粒子的个体最优位置; g_{best} 为当前最优位置,代表整个种群的最优解; c_1 和 c_2 为个体学习因子和社会学习因子; ω 为惯性权重; r_1 和 r_2 为 $[0, 1]$ 之间的随机数。

PSO 算法采用惯性权重来控制粒子对当前速度的依赖程度,通过自适应调整惯性权重,可使算法在不同阶段具有不同的特性,以平衡全局搜索与局部搜索^[19-20]。传统 PSO 多采用线性递减的惯性权重,但该方式偏于刚性,难以兼顾早期探索与后期收敛。因此,本文引入一种非线性自适应惯性权重策略,通过余弦函数动态调整惯性权重,替代传统的线性递减方式。该策略能够更平滑地调控粒子的搜索范围,在算法早期保持种群多样性以增强全局探索能力,在后期逐渐降低权重以加快收敛速度,从而平

衡算法的全局搜索与局部搜索能力。其表达式为:

$$\omega = \frac{\omega_1 + \omega_2}{2} + \left(1 + \cos\left(\frac{i_{\text{ter}}}{N} \cdot \pi\right)\right) \cdot \frac{\omega_1 - \omega_2}{4} \quad (18)$$

式中: ω_1 和 ω_2 为初始设置的最大和最小惯性权重因子; i_{ter} 为当前迭代次数; N 为总迭代次数。

惯性权重的动态变化如图 3 所示。

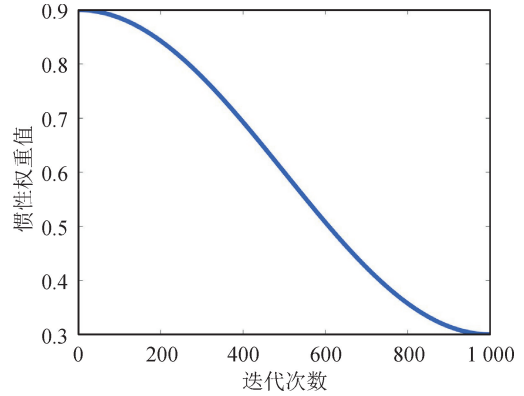


图 3 惯性权重的动态变化图

Fig. 3 Dynamic variation diagram of inertia weight

2.1.2 混合人工蜂群算法

人工蜂群算法是由 Karaboga^[21]于 2005 年提出的一种基于蜜蜂觅食行为的群体智能优化算法,它模拟了蜜蜂在寻找食物源和优化食物源选择过程中的行为,通过共享和交换蜂群信息,发现问题的最优解。该算法将蜂群分为 3 种类型:雇佣蜂、跟随蜂和侦查蜂。在搜索过程中,雇佣蜂优先利用已发现的食物源,并将信息传递给跟随蜂;跟随蜂依据共享信息选择食物源,并在其附近区域探索新的采集位置;食物源被耗尽后,对应的雇佣蜂转变为侦查蜂,随机寻找新的食物源以替代原有的已耗尽食物源。最优解最终由三种类型的蜜蜂协同搜索得出^[22]。本文通过融合人工蜂群算法的雇佣蜂阶段,随机选择其他粒子进行局部搜索更新,利用随机扰动机制代替传统的侦查行为,使算法跳出局部最优。其数学表达式为:

$$v_{ij} = x_{ij} + \varphi_{ij} \cdot (x_{ij} - x_{kj}) \quad (19)$$

$$k \neq i, j \in \{1, 2, \dots, D\}$$

式中: x_{ij} 为当前粒子(蜜源); x_{kj} 为随机选择的目标粒子; φ_{ij} 为扰动因子,模拟蜜蜂在周围的局部搜索行为; v_{ij} 为粒子更新后的位置; D 为粒子位置向量的维数。

本文在 PSO 算法的基础上,引入了人工蜂群算法的邻域搜索机制。在每轮粒子的位置更新后,通过适度扰动对个体进行局部调整,使算法在保持全局搜索能力的同时具备一定的局部开发能力,从而在路径优化过程中表现出更好的适应性和稳定性。

2.1.3 变异机制

在混合优化算法中,变异机制是一种提高搜索空间多样性的方法。当搜索过程陷入停滞时,适当的随机变异可打破粒子的收敛趋势,拓展搜索空间,提升算法性能^[23]。变异机制通常依据一定的变异概率触发,在每次迭代中,随机选择部分粒子进行变异操作。机制实现是对每个粒子 i ,在其路径点位置中,随机选择一个路径点进行变异,其数学表达式为:

$$v_i^{\text{mut}} = r(0,1) \cdot (v_{\max} - v_{\min}) + v_{\min} \quad (20)$$

式中: v_i^{mut} 为变异后粒子的速度; $v_{\max}$ 和 $v_{\min}$ 为粒子的最大和最小速度; $r(0,1)$ 为 $[0,1]$ 区间的随机数。

在算法运行过程中,通过以较小的概率对粒子的速度方向进行调整,有效打破了个体间的趋同状态,缓解了搜索过程中的停滞现象。这一机制作为对标准粒子群算法搜索方式的补充,增强了算法在局部区域的搜索活性,是提高解的质量和避免算法早熟收敛的重要手段。

2.1.4 Logistic 混沌映射

混沌映射是一种常见的算法优化机制,具有随机性和遍历性,可增强搜索过程的多样性和扩大覆盖范围,从而提升算法的全局探索能力和整体优化效果^[24]。通过引入混沌动力学的特性,使粒子的搜索路径具有更高的不可预测性和复杂性。常用的混沌映射包括 Logistic 映射、Tent 映射和 Sine 映射等,其中 Logistic 映射因其结构简单且混沌特性典型而被广泛应用。本文在算法中引入了 Logistic 混沌映射,并在迭代次数达到最大迭代次数的 75% 时启动该机制,具体数学表达式为^[25]:

$$c_v = m \times r(0,1) \times (1 - r(0,1)) \quad (21)$$

式中: c_v 为 Logistic 映射生成的混沌值; m 为 Logistic 映射的控制参数。

对于每个粒子的三维坐标 (x, y, z) ,在各坐标维度的位置分量上分别引入混沌值 c_v ,并根据该值调整粒子的位置。

$$p_n = p_s + c_v \times (b_{\max} - b_{\min}) \quad (22)$$

式中: p_n 为映射后的位置; p_s 为映射前的位置; $b_{\max}$ 和 $b_{\min}$ 为位置边界,用于确保新的粒子位置不会超出预设范围。

AHAPSO 算法在迭代后期引入 Logistic 混沌映射,借助其遍历性和随机性,引导粒子跳出当前区域,扩大搜索范围,提高了收敛末期的搜索覆盖度。

2.1.5 算法融合机制理论分析

为提升路径规划问题中算法的搜索能力与鲁棒性,本文构建了一种融合 PSO 与 ABC 机制的混合优化框架。PSO 和 ABC 分别具有良好的全局搜索

能力和局部搜索能力,将二者结合有助于提升算法在复杂环境下的表现,但在实际融合过程中,仍面临策略冲突与参数耦合等理论挑战。

主要体现为:PSO 通过速度与位置的更新实现连续搜索,而 ABC 则采用基于扰动的跳跃式更新方式,若同时作用于粒子位置,可能导致轨迹不稳定。为此,本文采用分阶段的更新策略,即先执行基于自适应惯性权重的 PSO 操作,在获得初步更新结果后,再引入扰动范围受控的 ABC 算法,进一步优化粒子位置。具体而言,PSO 中的惯性权重按照余弦函数进行非线性递减,使搜索过程由全局逐步过渡到局部,再将 ABC 中的扰动因子控制在合理范围内,并对边界进行限制,避免粒子越界或更新剧烈。该融合策略在实验中表现出较好的稳定性和收敛性能,相比单一优化算法,能够获得更优的路径规划结果。AHAPSO 算法的流程如图 4 所示。

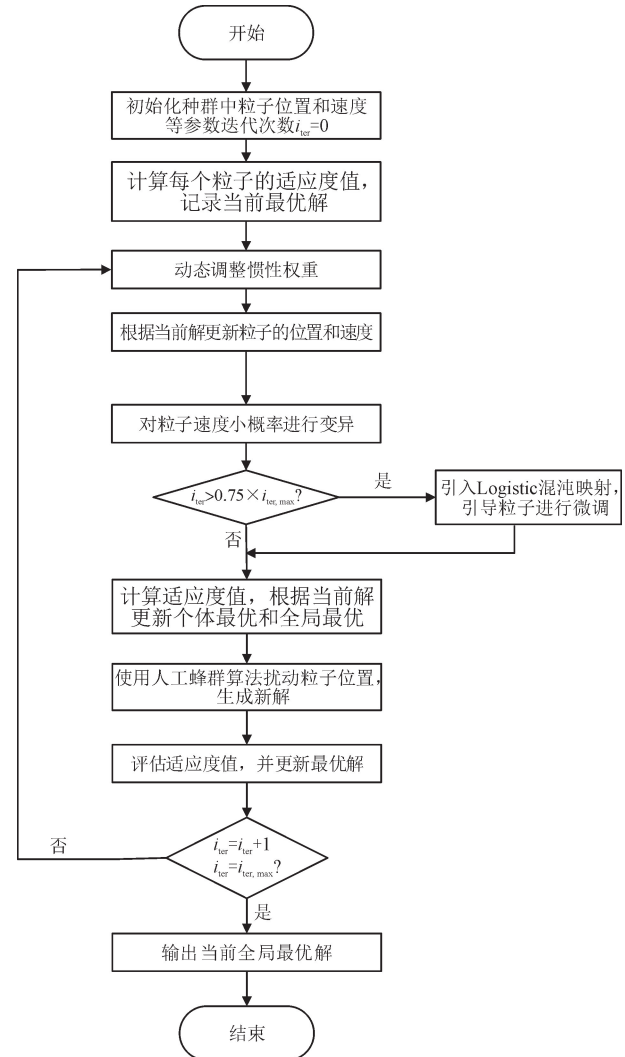


图 4 AHAPSO 算法流程图

Fig. 4 Flowchart of the AHAPSO algorithm

2.2 算法复杂度分析

本文所提 AHAPSO 算法融合了粒子群优化、人工蜂群算法以及混沌映射策略与变异机制,其整体时间复杂度与粒子数量 M 、路径点数 P 、最大迭代次数 N 以及适应度函数复杂度 f 相关。

2.2.1 粒子群算法部分

该部分的时间复杂度主要来源于粒子的位置更新、速度更新及适应度计算。

1) 对于每个粒子,更新粒子的位置和速度需要花费 $O(P)$ 时间, M 个粒子更新的时间复杂度为 $O(P \times M)$ 。

2) 每次需计算 M 个粒子的适应度,若单次适应度函数的时间复杂度为 $O(f)$,则该部分为 $O(M \times f)$ 。

3) 需判断每个粒子是否更新个体最优,并更新全局最优,该操作的时间复杂度为 $O(1)$ 。

因此,粒子群算法总的时间复杂度为 $O(N \times (P \times M + M \times f))$ 。

2.2.2 变异机制部分

变异机制以一定的概率随机改变粒子的速度或位置。由于变异只发生在一个路径点上,且每次变异操作仅对一个粒子中的一个维度产生影响,因此变异的时间复杂度为 $O(1)$ 。在 M 个粒子中进行变异操作时,总的时间复杂度为 $O(M)$ 。该复杂度为常数级,对整体时间复杂度影响较小。

2.2.3 混合人工蜂群部分

在此阶段,需要对每个粒子执行位置扰动、边界限制与适应度更新。对于每个粒子,位置扰动和边界限制需要 $O(P)$ 时间,适应度计算需要 $O(f)$ 时间。共有 M 个粒子,因此总的时间复杂度为 $O(M \times (P + f))$ 。

2.2.4 Logistic 混沌映射部分

在迭代过程中,当迭代达到一定次数时,采用混沌映射调整粒子的位置,混沌映射的时间复杂度为 $O(1)$, M 个粒子的总体复杂度为 $O(M)$ 。

综上,算法总的时间复杂度为以上策略的组合 $O(N \times M \times (P + f))$ 。通过优化策略的设计,在保证算法搜索能力的同时,控制了时间开销,确保算法具备良好的可扩展性和实际应用效率。

2.3 参数敏感性分析

2.3.1 惯性权重参数

为进一步评估惯性权重对算法优化性能的影响,本文采用基于经验选择的典型惯性权重参数组合进行敏感性分析。在保持其他参数不变的前提下,选取 7 组不同的惯性权重上下限组合进行对比,

如表 1 所示。

表 1 不同惯性权重组合的参数设置
Tab.1 Parameter settings for different inertia weight combinations

组别	最大惯性权重 因子 ω_1	最小惯性权重 因子 ω_2
第 1 组	1.0	0.1
第 2 组	1.0	0.2
第 3 组	0.9	0.1
第 4 组	0.9	0.2
第 5 组	0.9	0.3
第 6 组	0.8	0.3
第 7 组	0.7	0.1

图 5 展示了不同惯性权重组合下,算法迭代 300 次的适应度收敛曲线。

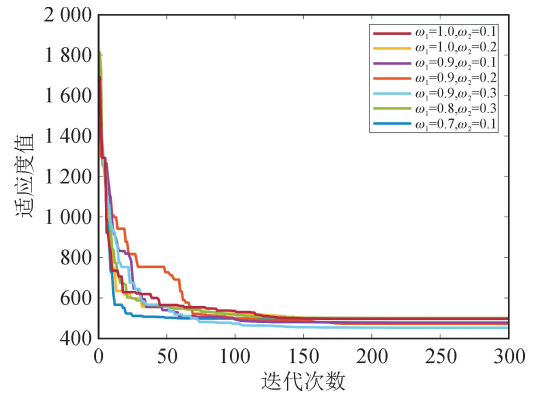


图 5 不同惯性权重组合下算法的收敛曲线

Fig.5 Convergence curves of the algorithm in different inertia weight combinations

从图 5 可以看出,所有组合在初期均表现出快速下降的趋势,表明算法具有良好的初始收敛能力。比较而言,参数组合 $\omega_1 = 0.9$ 、 $\omega_2 = 0.3$ 的最终适应度表现优异,且稳定性较强,说明该组合能够在全局搜索与局部开发之间取得较好的平衡。

综上,本文选择 $\omega_1 = 0.9$ 、 $\omega_2 = 0.3$ 的惯性权重参数组合,来实现算法在寻优速度与解的质量上的均衡。

2.3.2 变异概率参数

为分析变异概率(m_p)对算法性能的影响,本文在其他参数保持不变的条件下,选取不同的变异概率值进行敏感性实验。具体测试范围为 $m_p = \{0.1, 0.2, 0.3, 0.4, 0.5\}$ 。图 6 展示了不同变异概率下,算法迭代 300 次的适应度收敛曲线。

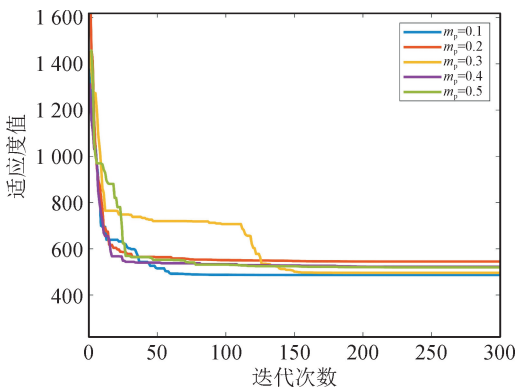


图 6 不同变异概率下算法的收敛曲线
Fig. 6 Convergence curves for the algorithm with different mutation probabilities

从图 6 可以看出,所有设置在初期均表现出较快的收敛趋势,但在中后期其表现有所差异。其中, $m_p=0.3$ 时,算法表现出明显的收敛减缓甚至停滞现象,尽管其最终适应度与 $m_p=0.1$ 时接近,但收敛过程不够平稳,稳定性较差。说明过高的变异概率可能破坏已有较优解的连续性,影响其局部开发能力。相比之下, $m_p=0.1$ 时,算法在整个迭代过程中表现出更快的收敛速度和更高的稳定性,且整体波动较小,更有利于算法在局部开发阶段持续优化,提升路径质量。

因此,本文将 $m_p=0.1$ 作为默认变异概率。

2.3.3 Logistic 混沌映射触发条件

本文引入的 Logistic 混沌映射并非在整个优化过程中持续激活,而是在迭代进度达到设定阈值后才被触发,以避免过早介入而破坏已有较优解的结构。为分析该触发条件对算法性能的影响,本文设置了三组典型的触发阈值:0.25、0.50、0.75,即分别在总迭代次数的 25%、50%、75% 时启用混沌映射机制,实验结果如图 7 所示。

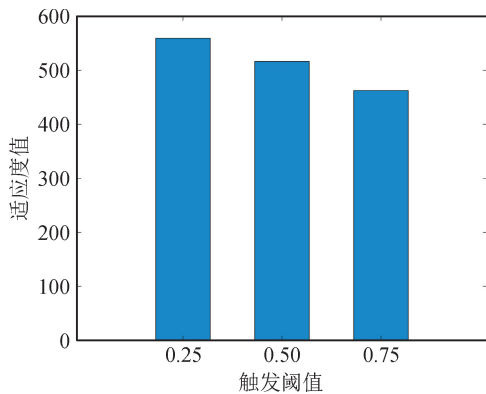


图 7 不同映射触发时机对算法性能的影响
Fig. 7 Impact of different chaotic mapping trigger timings on algorithm performance

从图 7 可以看出,随着触发时间的推迟,算法的优化效果逐渐提升。当触发阈值为 75% 时,最终适应度值最低,说明算法可以在前期稳定收敛的基础上,利用混沌映射有效跳出局部最优,进一步提升解的质量。相比之下,若在早期就触发混沌映射,虽增强了搜索的多样性,但可能会干扰已有收敛趋势,导致最终优化效果下降。

因此,本文将最大迭代次数的 75% 作为混沌映射的触发条件。

2.3.4 适应度函数权重系数

为分析适应度函数中各项权重系数对算法性能的影响,依据对路径成本、碰撞成本、飞行高度成本等的侧重,设置了多组具有代表性的权重组合,其权重系数如表 2 所示,算法的适应度收敛曲线如图 8 所示。

表 2 适应度函数的权重系数

Tab. 2 Weight coefficients in the fitness function

权重	路径成本	碰撞成本	飞行高度成本	平滑度成本	雷达威胁成本
b_1	0.1	0.3	0.3	0.2	0.1
b_2	0.2	0.2	0.2	0.2	0.2
b_3	0.3	0.3	0.1	0.2	0.1
b_4	0.1	0.4	0.2	0.2	0.1
b_5	0.3	0.1	0.3	0.2	0.1

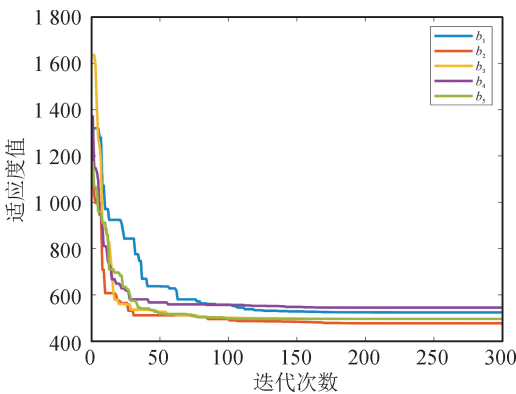


图 8 不同适应度函数权重下算法的收敛曲线
Fig. 8 Convergence curves of the algorithm with different fitness function weights

由图 8 可知,不同权重系数下算法的收敛速度和最终适应度值总体差异较小,均能在较短迭代次数内达到稳定状态,表明算法在适应度函数构成方面具有较强的鲁棒性。尽管个别系数早期的下降速度略有差异,但整体收敛趋势一致,最终结果相近。

因此,适应度函数中各项成本因子的权重设置对算法整体稳定性的影响较小,算法具备一定的鲁棒性。在实际应用中,可根据具体任务需求灵活调整权重参数,同样能保持良好的优化性能。

3 仿真实验

3.1 无人机三维路径仿真实验

在仿真实验中设置两种不同规模的地图,其山峰数量分别为 9 个和 12 个,相关仿真参数如表 3 所示。

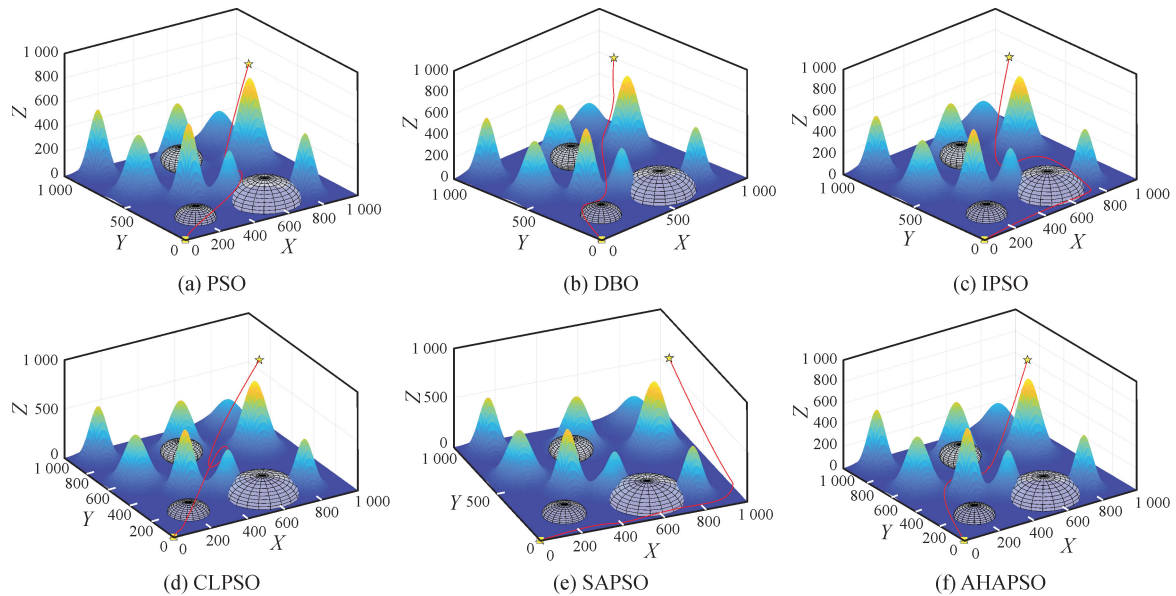
针对无人机三维路径规划问题,本文采用 6 种优化算法进行求解:PSO 算法、DBO 算法、引入信息素增强机制与随机扰动策略的改进型粒子群优化算法(IPSO)、通过综合学习策略提升种群多样性的协同粒子群优化算法(CLPSO)、模拟退火粒子群优化算法(SAPSO)及 AHAPSO 算法。

在不同环境模型下,经过 300 次迭代后,各算法得到的最优无人机路径如图 9 和图 10 所示。图 11、图 12 为地图一、地图二情形下各算法的收敛曲线。表 4 和表 5 为 6 种算法在迭代 300 次后的指标统计。

表 3 仿真参数

Tab. 3 Simulation parameters

参数	取值
最大迭代次数	300
种群规模	50
路径点数量	4
最大惯性权重因子 ω_1	0.9
最小惯性权重因子 ω_2	0.3
个体学习因子 c_1	1.5
社会学习因子 c_2	1.5
变异概率 m_p	0.1
适应度函数权重系数	0.1, 0.3, 0.3, 0.2, 0.1
地图范围	1 000×1 000×1 000
起始点	(1,1,1)
地图一目标点	(1 000,900,600)
地图一山峰数量/个	9
地图一雷达数量/个	3
地图二目标点	(900,900,900)
地图二山峰数量/个	12
地图二雷达数量/个	4



注:黄色方块图标为路径起点位置,黄色五角星图标为路径终点位置;蓝色圆锥部分为山体模型,白色网状球体部分为雷达威胁模型,下同。

图 9 地图一中各算法的路径对比

Fig. 9 Comparison of various algorithm paths in Map 1

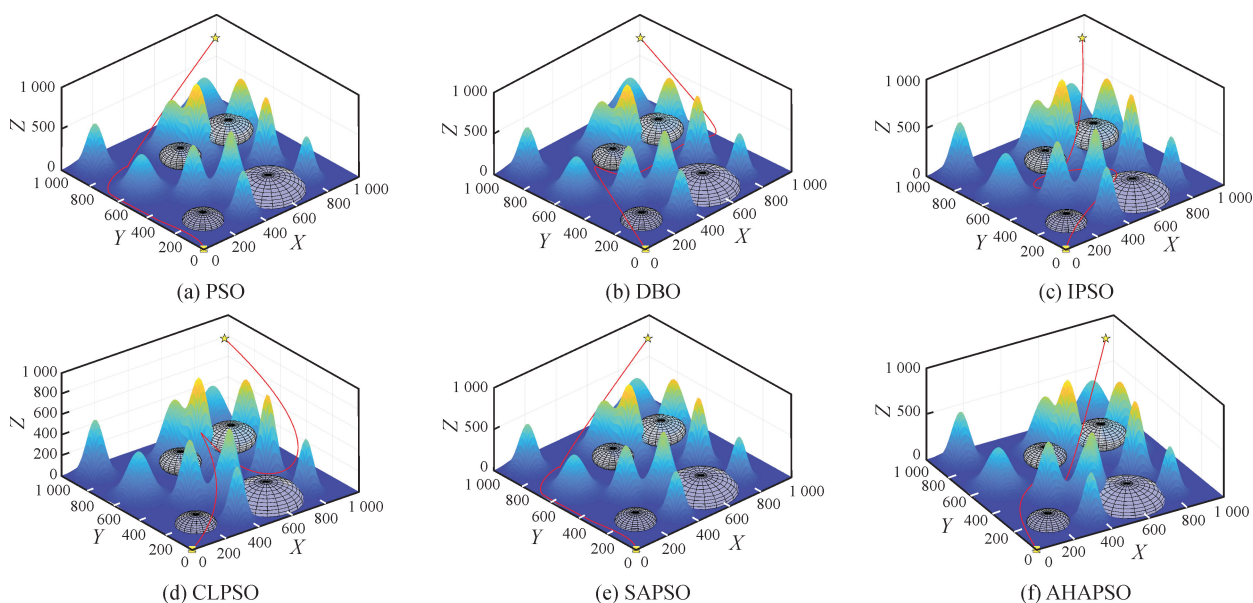


图 10 地图二中各算法的路径对比

Fig. 10 Comparison of various algorithm paths in Map 2

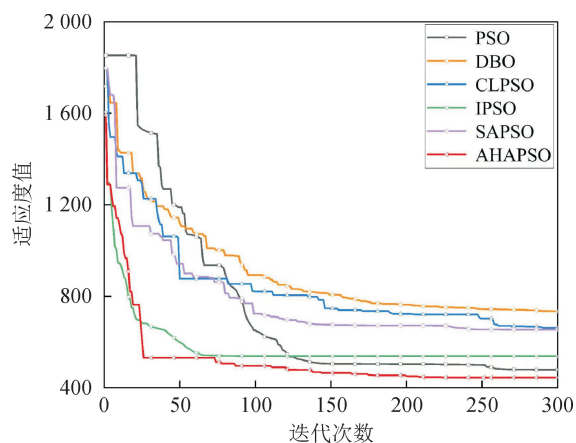


图 11 地图一各算法的收敛曲线

Fig. 11 Convergence curves of various algorithms on Map 1

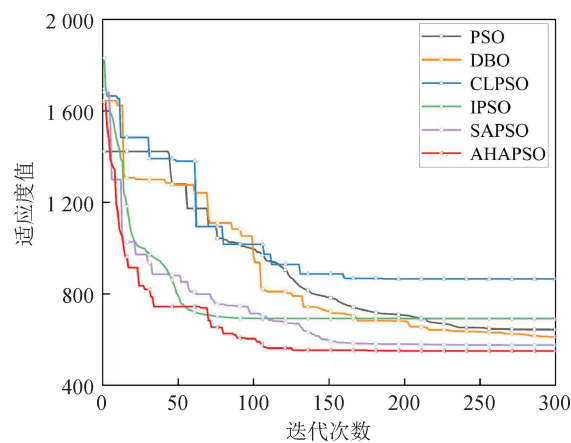


图 12 地图二各算法的收敛曲线

Fig. 12 Convergence curves of various algorithms on Map 2

表 4 地图一各算法的指标统计

Tab. 4 Statistical indicators of various algorithms on Map 1

指标	PSO	DBO	CLPSO	IPSO	SAPSO	AHAPSO
最低适应度	1 853.3	1 718.4	1 796.9	1 604.5	1 798.1	1 591.7
最优适应度	478.92	734.08	662.12	537.95	654.01	445.01
平均适应度	755.57	909.07	843.91	581.87	787.62	519.19
总耗时/s	1 584	1 450	1 496	1 638	1 600	1 312
平均单次迭代耗时/s	5.28	4.83	4.98	5.46	5.33	4.37

表 5 地图二各算法的指标统计

Tab. 5 Statistical indicators of various algorithms on Map 2

指标	PSO	DBO	CLPSO	IPSO	SAPSO	AHAPSO
最低适应度	1 422.7	1 645.3	1 699.1	1 830.8	1 682.1	1 637.1
最优适应度	643.79	611.97	865.08	691.47	575.80	550.57
平均适应度	904.97	881.82	1 023.33	767.53	706.69	639.49
总耗时/s	1 575	1 564	1 477	1 461	1 669	1 390
平均单次迭代耗时/s	5.25	5.21	4.92	4.87	5.56	4.63

实验结果表明,6种算法均能规划出从起始点到目标点的路径,其中AHAPSO算法的解决方案最优。在地图一中,各算法的路径规划存在明显差异。PSO路径抬升明显,贴近障碍边缘,说明其在全局搜索和避障方面存在不足;DBO虽能避开障碍,但路径变化较大,过渡不够平滑,搜索方式偏于激进;CLPSO和IPSO在路径紧凑性上有所改进,但仍存在不必要的高度波动或冗余段,反映出其搜索策略在精度或跳跃能力上存在不足;SAPSO路径较为平滑,但绕行距离较长,显示其搜索过程偏向保守,影响了路径效率。相比之下,AHAPSO利用自适应权重调节和蜂群搜索策略,使路径在穿越障碍区域时更合理,避障过程更自然。从图11和表4可以看出,AHAPSO在地图一中收敛速度较快,最终结果稳定,最优适应度、平均适应度和总耗时均优于其他算法,表现出较强的搜索能力和较高的路径质量。

在地图二中,受障碍密集程度和空间通行性影响,各算法在路径规划中存在明显差异。PSO路径较为直接,但避障不充分,存在穿越风险区域的情况;DBO在避障过程中路径波动较大,整体不够平滑;CLPSO在绕障方面有一定能力,但路径起伏较大,连续性较差;IPSO整体路径较为合理,避障效果明显,但部分路径仍存在不必要的高度变化;SAPSO更偏向保守,路径基本沿边界绕行,避障成功但路径偏长、效率不高。相对而言,AHAPSO能够在复杂环境中规划出较短且平稳的路径,能较好地避开多个障碍区域且路径衔接自然,整体表现更优。在地图二中,结合图12和表5的结果可知,AHAPSO算法的整体收敛趋势更为稳定,特别在70代之后,其适应度的下降速度逐渐优于其他算法,最终达到较低的收敛值。与传统PSO、CLPSO等算法相比,AHAPSO在收敛速度和路径质量方面均表现出一定优势,说明其在复杂环境下依然具备良好的稳定性和综合优化能力。

综上,AHAPSO在路径质量、收敛速度及计算效率方面均表现出明显优势,适用于复杂环境下的无人机路径规划。

进一步的时间复杂度分析与仿真实验结果表明,在相同的硬件与测试环境下,AHAPSO算法具有更低的平均迭代耗时和总运行时间。因此,无论环境复杂度如何变化,该算法均能够生成更平滑、更适于无人机飞行的路径,从而大幅提升无人机的路径规划效率。

3.2 基于多目标的动态环境仿真实验

为验证算法在复杂动态环境下的鲁棒性与适应性,在三维仿真实验中引入动态移动障碍物和雷达威胁模型,并增加了无人机数量。

针对该实验,本文构建了一个三维空间范围为 $1\,000\times 1\,000\times 1\,000$ 的仿真环境,并在该空间中随机生成了一定数量的静态障碍物。为增强实验的动态特性,引入了若干移动障碍,以模拟真实环境中不断变化的障碍分布情况。此外,设置了多个雷达干扰球体,用于模拟通信遮蔽和探测威胁区域,以增加路径规划的约束条件和复杂性。仿真场景中包含3架具有独立起点和目标点的无人机,以检验AHAPSO算法在多目标动态环境下的适应能力与鲁棒性。

表6列出了路径规划实验中的相关环境参数,表7展示了多无人机在动态环境下的路径规划结果。图13为各轮次多无人机的路径规划示意图,其中动态障碍物在每一轮次均会发生位置变化。

表6 相关环境参数

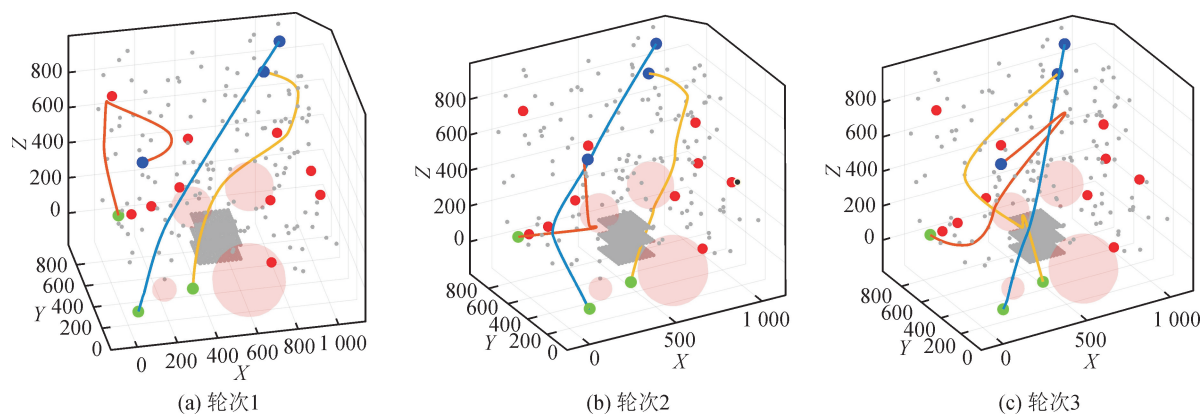
Tab.6 Relevant environmental parameters

参数	取值
静态障碍物/个	150
动态障碍物/个	10
UAV数量/个	3
雷达数量/个	4
轮次	3
UAV1起始点	(10,10,10)
UAV1目标点	(900,900,900)
UAV2起始点	(100,900,10)
UAV2目标点	(50,100,800)
UAV3起始点	(300,100,50)
UAV3目标点	(800,800,800)

表7 多无人机动态环境下路径规划结果

Tab.7 Path planning results in dynamic environment for multiple UAVs

无人机编号	轮次	适应度值	成功率/%
UAV1	1	0.098E+05	100
	2	0.101E+05	100
	3	0.093E+05	100
UAV2	1	1.466E+05	100
	2	1.120E+05	100
	3	1.540E+05	100
UAV3	1	1.276E+05	100
	2	1.638E+05	100
	3	1.298E+05	100



注:灰色小球为静态障碍物,红色小球为动态障碍物;粉红色透明球体表示雷达威胁区域,灰色密集区域代表城市建筑群;绿色小球为 UAV 起始点,蓝色小球为 UAV 目标点。

图 13 多无人机动态环境下路径规划示意图

Fig. 13 Path planning visualization in dynamic environments for multiple UAVs

实验结果如表 7 所示,所有 UAV 在三轮动态测试中均完成了路径规划,成功率为 100%,每架 UAV 的适应度值在各轮测试中也表现出较好的稳定性,说明该算法在高干扰、高密度障碍环境下仍能保持良好的可行性。图 13 展示的三维轨迹表明,各 UAV 路径均成功避开了动态障碍和雷达区域,并能在密集的静态障碍环境中灵活穿行,未出现路径重叠或陷入死区的情况,体现了算法良好的避障性能与多目标协调能力。

综上所述,引入动态威胁和多 UAV 干扰场景后,算法仍能保持高成功率和稳定的适应度水平,验证了其在复杂场景下的鲁棒性与扩展性。

4 结 语

本文针对无人机全局路径规划问题,提出了一种优化后的自适应混合粒子群算法 AHAPSO。该算法融合了人工蜂群策略,以提升粒子群算法的搜索性能;针对优化过程中搜索空间的非线性特征,设计了自适应惯性权重调整机制,实现了全局探索与局部开发之间的动态平衡;此外,算法还引入了变异机制,并在迭代后期结合 Logistic 混沌映射,有效提升了搜索多样性,缓解了传统粒子群算法在后期易陷入收敛停滞的问题。

AHAPSO 算法在多种山地地形和动态环境模拟中均表现出良好的适应性和稳定性,体现了其实用性、高效性与综合优势。鉴于目前的实验是基于理想仿真环境,尚未部署真实飞行平台,后续将基于 AirSim 或 Gazebo 系统,结合传感器误差模拟,开展更具现实约束的验证工作。

参考文献:

- [1] Wang Y, Xu G, Long T, et al. Receding path planning for UAV swarms using priority-based artificial potential field [C]//2019 Chinese Automation Congress (CAC). IEEE, 2019: 3234-3239.
- [2] 况瑞杰,李祥,刘宇,等. 基于 LEM-YOLO 的轻量化多尺度无人机林火烟雾检测[J]. 激光与光电子学进展, 2025, 62(14): 425-436.
Kuang Ruijie, Li Xiang, Liu Yu, et al. LEM-YOLO-based lightweight multi-scale detection of forest fire smoke in UAV imagery[J]. Laser & Optoelectronics Progress, 2025, 62(14): 425-436.
- [3] Zhang R, Li M, Chen X, et al. UAV path planning method based on modified beluga whale optimization [C]//2023 IEEE International Conference on Unmanned Systems (ICUS). IEEE, 2023: 792-797.
- [4] Hou X, Liu F, Wang R, et al. A UAV dynamic path planning algorithm [C]//2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC). IEEE, 2020: 127-131.
- [5] Qu C, Gai W, Zhang J, et al. A novel hybrid grey wolf optimizer algorithm for unmanned aerial vehicle (UAV) path planning[J]. Knowledge-Based Systems, 2020, 194: 105530.
- [6] 陈琦,王亚杰,孙云飞,等. 基于蜣螂和灰狼复合算法的无人机三维路径规划[J]. 电光与控制, 2025, 32(6): 8-15.
Chen Qi, Wang Yajie, Sun Yunfei, et al. Three-dimensional path planning for UAVs based on DBO-GWO compound algorithm[J]. Electronics Optics & Control, 2025, 32(6): 8-15.
- [7] 梅雨琳,曲良东,饶爽. 多策略改进蜣螂优化算法的无人机航迹规划[J]. 电子测量技术, 2025, 48(11):

- 67-77.
- Mei Yulin, Qu Liangdong, Rao Shuang. Multi-strategy improved dung beetle optimization algorithm for UAV path planning[J]. *Electronic Measurement Technology*, 2025, 48(11): 67-77.
- [8] 胡鑫宇,向慧明. 基于多策略改进鲸鱼算法的无人机三维路径规划[J]. *现代电子技术*, 2025, 48(9): 79-85.
- Hu Xinyu, Xiang Huiming. Unmanned aerial vehicle 3D path planning based on multi-strategy improved whale optimization algorithm[J]. *Modern Electronics Technique*, 2025, 48(9): 79-85.
- [9] Wang X, Meng X. UAV online path planning based on improved genetic algorithm[C]//2019 Chinese Control Conference (CCC). IEEE, 2019: 4101-4106.
- [10] Tu B, Wang F, Han X. 3D path planning for UAV based on a hybrid algorithm of marine predators algorithm with quasi-oppositional learning and differential evolution[J]. *Egyptian Informatics Journal*, 2024, 28: 100556.
- [11] Chai X, Zheng Z, Xiao J, et al. Multi-strategy fusion differential evolution algorithm for UAV path planning in complex environment[J]. *Aerospace Science and Technology*, 2022, 121: 107287.
- [12] 石博博,朱嘉晟,杨天慈,等. 改进粒子群权重的水下机器人路径规划[J]. *舰船科学技术*, 2025, 47(5): 159-165.
- Shi Bobo, Zhu Jiasheng, Yang Tianci, et al. Path planning of underwater vehicle based on improved particle swarm weight[J]. *Ship Science and Technology*, 2025, 47(5): 159-165.
- [13] Wang S, Chen H, Li L. Car path planning with improved artificial bee colony algorithm[C]//2024 IEEE International Conference on Industrial Technology (IC-IT). IEEE, 2024: 1-5.
- [14] Wu X, Bai W, Xie Y, et al. A hybrid algorithm of particle swarm optimization, metropolis criterion and RTS smoother for path planning of UAVs[J]. *Applied Soft Computing*, 2018, 73: 735-747.
- [15] Yu Z, Si Z, Li X, et al. A novel hybrid particle swarm optimization algorithm for path planning of UAVs[J]. *IEEE Internet of Things Journal*, 2022, 9(22): 22547-22558.
- [16] Zheng Z, Chen X. A game based path planning method for dual UAVs in complex environments[C]//2019 Chinese Control and Decision Conference (CCDC). IEEE, 2019: 5431-5437.
- [17] Eberhart R, Kennedy J. Particle swarm optimization [C]// *Proceedings of the IEEE International Conference on Neural Networks*. 1995, 4: 1942-1948.
- [18] Raj A, Punia P, Kumar P. A novel hybrid pelican-particle swarm optimization algorithm (HPPSO) for global optimization problem[J]. *International Journal of System Assurance Engineering and Management*, 2024, 15: 3878-3893.
- [19] Xin J, Chen G, Hai Y. A particle swarm optimizer with multi-stage linearly-decreasing inertia weight [C]//2009 International Joint Conference on Computational Sciences and Optimization. IEEE, 2009, 1: 505-508.
- [20] Yang C, Gao W, Liu N, et al. Low-discrepancy sequence initialized particle swarm optimization algorithm with high-order nonlinear time-varying inertia weight [J]. *Applied Soft Computing*, 2015, 29: 386-394.
- [21] Karaboga D. An idea based on honey bee swarm for numerical optimization[R]. Kayseri: Technical Report TR06, Erciyes University, 2005.
- [22] Li S, Li R, Li J, et al. An improved artificial bee colony algorithm based on k-means clustering[C]//2024 5th International Conference on Information Science, Parallel and Distributed Systems (ISPDS). IEEE, 2024: 470-479.
- [23] Ji R, Chen P, Liu C, et al. Diameter measurement of fuel rods based on particle swarm optimization with all-parameter adaptive mutation mechanism[J]. *IEEE Access*, 2020, 8: 113624-113635.
- [24] Wang B, Zhang Z, Siarry P, et al. A nonlinear African vulture optimization algorithm combining Henon chaotic mapping theory and reverse learning competition strategy[J]. *Expert Systems with Application*, 2024, 236: 121413.
- [25] Herbadji D, Herbadji A, Kahia H, et al. An enhanced logistic chaotic map based tweakable speech encryption algorithm[J]. *Integration*, 2024, 97: 102192.

(责任编辑 周 蓓)