

◆生产场景◆

一种巡检无人机航迹自适应规划算法*

石成钰¹, 王天骄¹, 张莹莹¹, 顾悦昕¹, 冯笑^{1,2**}

(1. 国网电力空间技术有限公司, 北京 102200; 2. 北京邮电大学网络空间安全学院, 北京 100876)

摘要:针对传统方法求解复杂环境中巡检无人机航迹规划容易出现精度差、收敛于局部最优航迹的不足, 提出一种自适应高斯云与 α 稳定分布变异白鲸优化器(G- α -BWO)的巡检无人机航迹规划算法。该算法引入分段线性混沌映射(Piecewise Linear Chaotic Map, PWLCM)-准对立学习初始化策略提升初始航迹解的多样性和质量, 利用自适应高斯云策略提升算法对全局最优航迹的搜索能力, 利用自适应 α 稳定分布变异策略使算法脱离局部最优航迹, 丰富搜索空间并提高收敛速度。首先构建巡检无人机航迹规划模型, 设计考虑航程大小、高度、转角及障碍物威胁的权重代价函数, 将三维航迹规划问题转换为约束条件多目标优化问题, 然后通过G- α -BWO迭代搜索巡检无人机最优航迹。结果表明, 改进算法搜索航迹可安全避障, 且代价更小, 有助于提高巡检任务效率和安全性。

关键词:巡检无人机; 航迹规划; 高斯云; α 稳定分布; 白鲸优化器

中图分类号: TP393 文献标识码: A 文章编号: 1005-9164(2024)05-0954-12

DOI: 10.13656/j.cnki.gxkx.20241127.012

电网线路分布广泛, 环境复杂恶劣, 巡检工作量大, 依靠人工巡检费时费力^[1]。随着人工智能及图像处理技术的快速发展, 无人机技术因为成本低、效率高、易于操控且可适应于复杂环境, 已经广泛应用于电力线路故障的巡检。如何有效规划电力巡检无人机的航迹, 成为影响电力巡检能力的关键问题^[2]。

航迹规划旨在确保基本电力巡检任务的要求下, 搜索最优飞行路径, 提高任务执行效率, 降低任务执行成本。目前, 航迹规划的主要方法包括迪杰斯特拉

算法、A星(A*)算法^[3]、人工势场算法^[4]和随机树算法^[5]等。但迪杰斯特拉算法本身空间和时间复杂度较高, A星算法容易生成较多的航迹冗余解, 且时间复杂度在规划场景规模递增时增长过快, 人工势场算法在面临多威胁源的复杂场景时容易生成局部最优而过快收敛, 随机树算法生成的航迹平滑性不足。由此可见, 在搜索效率与最优精度的并行优化上传统算法都存在一定的不足。而智能优化算法因在解决非线性、非凸多峰的复杂航迹规划问题上具有独特的

收稿日期: 2024-04-08

修回日期: 2024-05-17

* 国家自然科学基金项目(62001055)资助。

【第一作者简介】

石成钰(1988—), 女, 硕士, 助理工程师, 主要从事智能交通研究。

【**通信作者简介】

冯笑(1989—), 女, 博士, 高级工程师, 主要从事人工智能与信息安全研究, E-mail: fengxiao@bupt.edu.cn。

【引用本文】

石成钰, 王天骄, 张莹莹, 等. 一种巡检无人机航迹自适应规划算法[J]. 广西科学, 2024, 31(5): 954-965.

SHI C Y, WANG T J, ZHANG Y Y, et al. A Adaptive Planning Algorithm for the Trajectory of Patrol Unmanned Aerial Vehicle [J]. Guangxi Sciences, 2024, 31(5): 954-965.

优势被广泛应用。田周泰等^[6]提出了一种自适应步长的人工鱼群优化的航迹规划算法,该算法可通过自适应权重值平衡搜索精度与收敛速度;郭志明等^[7]设计了一种改进的蝗虫算法,通过引入逻辑蒂斯函数有效解决了协同航迹搜索效率差的问题;王康等^[8]设计了一种基于改进沙猫群算法的三维航迹规划算法,通过 Levy 飞行机制提高了航迹的搜索精度;丁敏等^[9]提出了一种基于改进蝴蝶算法的航迹规划算法,通过引入自适应惯性权重和动态概率余弦选择机制有效提高了航迹的全局搜索能力。除此之外,鲸鱼算法^[10]、蚁群算法^[11]、鸡群算法^[12]、海鸥算法^[13]及粒子群算法^[14]也在无人机航迹规划领域得到了应用与验证。基于“*No free lunch*”定理考虑,在有限的计算资源和时间约束下,并没有一种算法可以解决所有优化问题。解决实际的航迹规划问题还需优化算法性能,改进算法固有缺陷,从而提升算法的求解性能。

白鲸优化器(Beluga Whale Optimizer, BWO)是2022年提出的一种新型智能算法^[15],模拟了白鲸种群的游泳、捕食和鲸鱼坠落等行为模式。该算法模型简单,搜索精度较高,已经在工艺规划^[16]、车间作业调度^[17]、RFID网络规划^[18]等领域得到了有效应用与验证。但标准 BWO 依然存在收敛精度低、易得到局部最优的不足,尤其在优化巡检无人机航迹规划时无法有效处理全局搜索精度差和局部极值点跳离的问题。本文将提出一种自适应高斯云与 α 稳定分布(以下简称为 α 分布)变异 BWO ($G-\alpha$ -BWO)算法,用于解决电力巡检无人机航迹规划中全局搜索精度差和局部极值点跳离的问题,通过分段线性混沌映射(Piecewise Linear Chaotic Map, PWLCM)-准对立学习初始化策略提升初始航迹的多样性和质量,引入高斯云策略提高算法对全局最优航迹的搜索能力,利用自适应 α 分布变异策略使算法能够跳离局部最优航迹,开拓搜索空间,并通过不同复杂度的场景地图验证算法搜索最优航迹的性能。

1 环境建模

利用公式(1)复合函数模拟巡检无人机航迹规划地貌特征:

$$z_1(x, y) = \sin(y + a) + b \sin x + c \cos(d \sqrt{x^2 + y^2}) + e \cos y + f \sin(g \sqrt{x^2 + y^2}), \quad (1)$$

式中, (x, y) 为投影在二维平面内的 x, y 轴坐标值, $z_1(x, y)$ 为对应二维坐标 (x, y) 的高程值, $a, b, c, d,$

e, f, g 均为特征系数,控制数字地形的地貌起伏特征。

基于地貌特征,再叠加山体模型模拟障碍物,山体模型数学函数式为

$$z_2(x, y) = \sum_{i=1}^n h_i \exp\left[-\left(\frac{x - x_{oi}}{a_i}\right)^2 - \left(\frac{y - y_{oi}}{b_i}\right)^2\right], \quad (2)$$

式中, $z_2(x, y)$ 对应二维坐标 (x, y) 的高程值; h_i 为地形系数,控制山体 i 的高度; (x_{oi}, y_{oi}) 为山体 i 的中心坐标; a_i, b_i 分别为山体 i 在 x, y 轴上的衰减量,控制山体坡度。联立公式(1)和公式(2)可得完整地貌模型,定义为

$$Z(x, y) = \max[z_1(x, y), z_2(x, y)]. \quad (3)$$

图1展示了含有4个山体的基本地貌图。

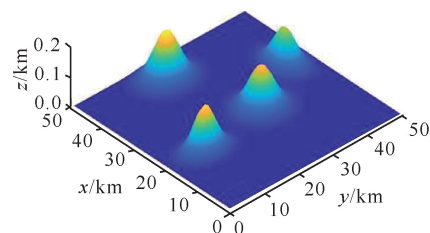


图1 基本地貌图建模

Fig. 1 Basic geomorphological map modeling

2 航迹规划模型

2.1 总航程代价及约束

针对电力巡检任务,航迹规划的首要目标是搜索抵达目标点的最短航迹,以节省无人机的能量损耗。假设 $P_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j})$ 为一个航迹点,代表航迹方案 i 中航迹点 j 的三维坐标。令航迹点数量为 n , 起点坐标为 $P_{i,0}$, 终点坐标为 $P_{i,n+1}$, 则航迹解可表示 $X_i = \{P_{i,0}, P_{i,1}, P_{i,2}, \dots, P_{i,n}, P_{i,n+1}\}$ 。令 $|P_{i,j}, P_{i,j+1}|$ 为相邻航迹点 $P_{i,j}$ 与 $P_{i,j+1}$ 间的欧氏距离,则航迹解 X_i 的总航程可定义为距离代价函数 $C_{\text{dis}}(X_i)$:

$$C_{\text{dis}}(X_i) = \sum_{j=0}^n |P_{i,j}, P_{i,j+1}|, \quad (4)$$

$$|P_{i,j}, P_{i,j+1}| = \sqrt{(x_{i,j+1} - x_{i,j})^2 + (y_{i,j+1} - y_{i,j})^2 + (z_{i,j+1} - z_{i,j})^2}. \quad (5)$$

同时,由于巡检无人机是电池供电,能源受限,因此执行巡检任务需要受到最大航程约束,即满足约束条件:

$$C_{\text{dis}}(X_i) \leq L_{\text{max}}, \quad (6)$$

式中, L_{max} 为无人机单程的最大航程约束。

2.2 高度代价及约束

电力巡检无人机应尽可能保持平稳的高度, 以提升稳定性和任务成功率。令 $h_{i,j}$ 为航迹点 $P_{i,j}$ 至基准地形的高度。将 $P_{i,j}$ 点的高度代价定义为

$$H_{i,j} = \gamma_h \cdot \frac{h_{i,j} - H_{\min}}{H_{\max} - H_{\min}}, \quad (7)$$

式中, γ_h 为飞行高度代价因子, H_{\min} 、 H_{\max} 分别对应高度的下界和上界。同时飞行高度 h 应满足以下约束条件:

$$H_{\min} \leq h \leq H_{\max}, \quad (8)$$

则航迹 X_i 的飞行高度代价函数 $C_{\text{hei}}(X_i)$ 可定义为

$$C_{\text{hei}}(X_i) = \sum_{j=1}^{n-1} H_{i,j}. \quad (9)$$

2.3 转角代价及约束

无人机为保证飞行稳定性及巡检任务的可靠性, 应尽可能减少航线大角度偏移, 即在航向不出现背离的前提下确保更小的飞行转角。飞行转角包括水平转角和爬升角, 模型如图 2 所示, $P_{i,j}$ 、 $P_{i,j+1}$ 、 $P_{i,j+2}$ 为 3 个连续航迹点, $|P_{i,j}, P_{i,j+1}|$ 、 $|P_{i,j+1}, P_{i,j+2}|$ 分别为 $P_{i,j}$ 、 $P_{i,j+1}$ 和 $P_{i,j+1}$ 、 $P_{i,j+2}$ 间的欧氏距离, $|P'_{i,j}, P'_{i,j+1}|$ 、 $|P'_{i,j+1}, P'_{i,j+2}|$ 分别为 $|P_{i,j}, P_{i,j+1}|$ 、 $|P_{i,j+1}, P_{i,j+2}|$ 在二维平面的投影。

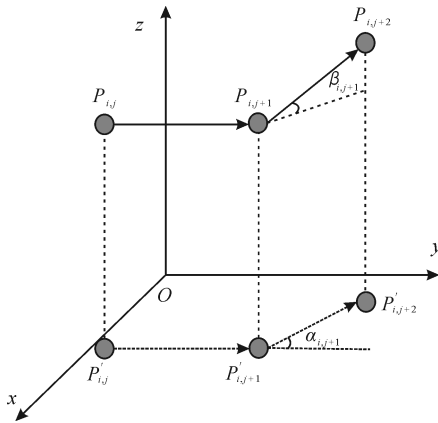


图 2 飞行转角约束

Fig. 2 Flight angle constraint

令 k 为轴正方向单位矢量, $\alpha_{i,j+1}$ 为航迹点 $P_{i,j}$ 、 $P_{i,j+1}$ 、 $P_{i,j+2}$ 之间从航迹段 $|P_{i,j}, P_{i,j+1}|$ 转向 $|P_{i,j+1}, P_{i,j+2}|$ 的水平转角, $\beta_{i,j+1}$ 为爬升角, $z_{i,j}$ 、 $z_{i,j+1}$ 为点 $P_{i,j}$ 、 $P_{i,j+1}$ 相对于海平面的高度, 则:

$$|P'_{i,j}, P'_{i,j+1}| = k \times (|P_{i,j}, P_{i,j+1}| \times k), \quad (10)$$

$$\alpha_{i,j+1} =$$

$$\arctan \frac{|P'_{i,j}, P'_{i,j+1}| - |P'_{i,j+1}, P'_{i,j+2}|}{|P'_{i,j}, P'_{i,j+1}| \times |P'_{i,j+1}, P'_{i,j+2}|}, \quad (11)$$

$$\beta_{i,j+1} = \arctan \frac{z_{i,j+1} - z_{i,j}}{|P'_{i,j}, P'_{i,j+1}|}. \quad (12)$$

令 γ_{le} 、 γ_{re} 分别为水平转角和爬升角超出约束限制的代价因子。航迹 X_i 的飞行转角代价函数 $C_{\text{ang}}(X_i)$ 定义为

$$C_{\text{ang}}(X_i) = \gamma_{\text{le}} \sum_{j=1}^{n-2} \alpha_{i,j+1} + \gamma_{\text{re}} \sum_{j=1}^{n-1} |\beta_{i,j} - \beta_{i,j-1}|. \quad (13)$$

式中, $\beta_{i,j}$ 和 $\beta_{i,j-1}$ 为相邻的两个爬升角。

同时, 为保持垂直方向和水平方向的飞行稳定性, 设置无人机受最大转弯角和最大爬升角的限制。令最大转弯角为 α_{\max} , 则相邻航迹点 $P_{i,j}$ 、 $P_{i,j+1}$ 之间的转弯角约束条件为

$$\frac{|y_{i,j+1} - y_{i,j}|}{|x_{i,j+1} - x_{i,j}|} \leq \tan \alpha_{\max}. \quad (14)$$

令最大爬升角为 β_{\max} , 则相邻航迹点 $P_{i,j}$ 、 $P_{i,j+1}$ 之间的爬升角约束条件为

$$\frac{|z_{i,j+1} - z_{i,j}|}{|P'_{i,j+1} - P'_{i,j}|} \leq \tan \beta_{\max}, \quad (15)$$

式中, 分子为航段高度差, 分母为航段的水平投影间距。

2.4 障碍物威胁代价

无人机执行电力巡检任务过程中, 可能遭遇高塔和其他电磁辐射干扰, 本文将此类障碍物威胁形式化为圆柱体模型。令 C_k 为圆柱体 k 中心坐标, 即威胁源中心, R_k 为障碍物威胁半径, D 为障碍物外围碰撞威胁区域。障碍物威胁在平面内的投影如图 3 所示。

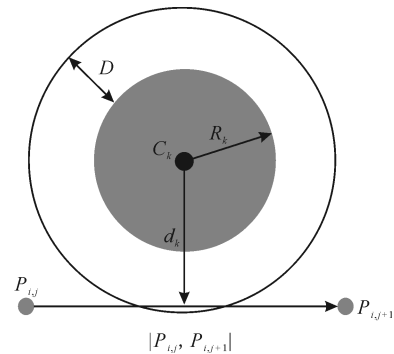


图 3 障碍物威胁区域

Fig. 3 Obstacle threat area

可见, 障碍物威胁成本与相邻航迹点 $P_{i,j}$ 、 $P_{i,j+1}$ 的欧氏距离 $|P_{i,j}, P_{i,j+1}|$ 至圆柱体 k 中心坐标距离 d_k 成反比。令 K 为威胁区域的集合, $k = \{1, 2, \dots, K\}$ 。则航迹解 X_i 的威胁代价函数 $C_{\text{thr}}(X_i)$ 可定义为

$$C_{\text{thr}}(X_i) = \sum_{j=0}^n \sum_{k=1}^K \Upsilon_k(|P_{i,j}, P_{i,j+1}|), \quad (16)$$

$$\Upsilon_k(|P_{i,j}, P_{i,j+1}|) = \begin{cases} 0, & d_k > D + R_k \\ \gamma_c \times [(D + R_k) - d_k], & R_k < d_k < D + R_k \\ \infty, & d_k < R_k \end{cases}, \quad (17)$$

式中, γ_c 为障碍物威胁代价因子, Υ_k 对应障碍物 k 对相应航段的威胁代价。

图 4 为添加障碍物后的地貌图。

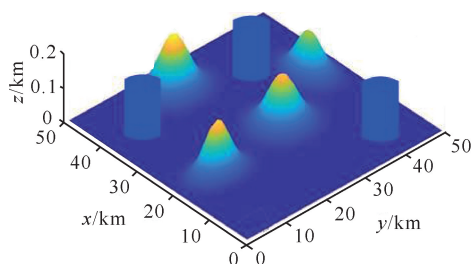


图 4 障碍物地貌图

Fig. 4 Obstacle topography map

2.5 航迹规划总代价

无人机在执行巡检任务时应在保证满足约束条件公式(6)、(8)、(14)、(15)的同时,综合考虑总航程最短、高度更低、转角最小及障碍物威胁最小的多目标优化。因此,可将航迹规划总代价函数 $C_{\text{tot}}(X_i)$ 定义为

$$C_{\text{tot}}(X_i) = \omega_1 C_{\text{dis}}(X_i) + \omega_2 C_{\text{hei}}(X_i) + \omega_3 C_{\text{ang}}(X_i) + \omega_4 C_{\text{thr}}(X_i), \quad (18)$$

约束条件为

$$\begin{cases} \text{条件公式(6)、(8)、(14)、(15)} \\ 0 < \omega_i < 1, \sum_{i=1}^4 \omega_i = 1, i = 1, 2, 3, 4 \end{cases}, \quad (19)$$

其中, $\omega_1, \omega_2, \omega_3, \omega_4$ 分别对应航程大小、高度、转角和威胁代价的权重因子,用于定义目标偏好程度。

3 G- α -BWO 算法

3.1 白鲸优化器(BWO)

BWO 是受白鲸游泳、捕食和鲸鱼坠落等行为模式启发的新型智能优化算法。算法中的平衡因子 B_f 和坠落因子 W_f 决定种群的具体行为模式, B_f 决定种群进入全局搜索或是局部开发阶段, W_f 决定种群是否进入鲸鱼坠落阶段。两个因子定义如下:

$$\begin{cases} B_f = B_0(1 - t/2T_{\text{max}}) \\ W_f = 0.1 - 0.05t/T_{\text{max}} \end{cases}, \quad (20)$$

式中, B_0 为(0,1)的随机值, t 为当前迭代次数, T_{max}

为最大迭代次数。当 $B_f > 0.5$ 时,种群进行全局搜索;当 $B_f \leq 0.5$ 时,种群进行局部开发。而当 $B_f < W_f$ 时,种群进入坠落阶段。

①全局搜索阶段。该阶段中,两个白鲸个体可以同步或以镜像方式成对游行,从而共同决定位置更新。具体模型为

$$X_{i,j}(t+1) = \begin{cases} X_{i,P_j}(t) + (X_{r,P_1}(t) - X_{i,P_j}(t)) \times \\ (1 + r_1) \sin(2\pi r_2), j = \text{even} \\ X_{i,P_j}(t) + (X_{r,P_1}(t) - X_{i,P_j}(t)) \times \\ (1 + r_1) \cos(2\pi r_2), j = \text{odd} \end{cases}, \quad (21)$$

式中, $X_{i,j}(t+1)$ 为个体 i 的 j 维更新位置; $X_{i,P_j}(t)$ 为个体 i 在 P_j 维的位置, P_j 为 d 维选取的随机值, $j = 1, 2, \dots, d$; $X_{r,P_1}(t)$ 为选择的随机个体 r , r_1, r_2 均为(0,1)的随机值,用于增强算法随机搜索。可见,该阶段算法将根据偶数(Even)和奇数(Odd)选择更新维度,从而使种群更新位置表现出两个白鲸个体的同步或镜像行为。

②局部开发阶段。该阶段中,种群个体会成群结队将小鱼引至浅水区,实现信息交流、位置分享进而合作觅食。具体模型为

$$X_i(t+1) = r_3 X_{\text{best}}(t) - r_4 X_i(t) + C_1 L_F (X_r(t) - X_i(t)), \quad (22)$$

式中, $X_i(t+1)$ 和 $X_i(t)$ 为个体 i 更新后和当前的位置, r_3, r_4 均为(0,1)的随机值; $X_{\text{best}}(t)$ 为迭代 t 时的种群最优解位置; $X_r(t)$ 为随机选择个体 r 的位置; C_1 为随机跳跃因子,用于定义 Levy 飞行的随机跳跃强度, $C_1 = 2r_4(1 - t/T_{\text{max}})$; L_F 为 Levy 飞行因子,定义为

$$L_F = 0.05 \times \frac{u \times \sigma}{|v|^{1/\beta}}, \quad (23)$$

式中,

$$\sigma = \frac{\Gamma(1 + \beta) \sin(\pi\beta/2)}{\Gamma(\frac{1 + \beta}{2}) \beta \times 2^{(\beta-1)/2}}, \quad (24)$$

式中, $\beta = 1.5$, u 和 v 均为服从正态分布的随机值, $\Gamma()$ 为伽马函数。

③鲸鱼坠落阶段。该阶段中,种群受到外来生命威胁,会迁移或因被捕食而坠入深海。此阶段结合种群个体的位置和坠落步长来更新位置。具体模型为

$$X_i(t+1) = r_5 X_i(t) - r_6 X_r(t) + r_7 X_{\text{step}}, \quad (25)$$

式中, r_5, r_6, r_7 为(0,1)的随机值; X_{step} 为坠落步长,

定义为

$$X_{\text{step}} = (u_b - l_b) \exp(-C_2 t / T_{\text{max}}), \quad (26)$$

式中, u_b 、 l_b 分别为搜索空间的上界和下界; C_2 为跳跃因子, 且 $C_2 = 2W_f \times N$, 其中 N 为种群规模。

3.2 G- α -BWO 算法的详细设计

3.2.1 PWLCM-准对立学习种群初始化

BWO 算法为了实现随机搜索的特征, 在种群初始化阶段具有伪随机性, 随机生成的种群个体分布并不均匀, 且无法满足种群多样性, 导致该算法效率低。混沌映射是一种同时满足随机性、遍历性和规律性的映射方式^[19]。为了改善 BWO 算法的初始分布均匀性, 在 G- α -BWO 算法中引入 PWLCM 生成算法的初始种群, 其数学表达式为

$$x(t+1) = \begin{cases} x(t)/p, & 0 \leq x(t) < p \\ x(t) - p/0.5 - 9, & p \leq x(t) < 0.5 \\ 1 - p - x(t)/0.5 - 9, & 0.5 \leq x(t) < 1 - p \\ 1 - x(t)/p, & 1 - p \leq x(t) < 1 \end{cases}, \quad (27)$$

式中, p 为混沌因子, 通常取 $p = 0.4$ 。

结合混沌映射虽然可以解决伪随机种群分布均匀性不足的问题, 但是混沌映射具有固有的受限周期

和不稳定的周期点, 为了解决这些问题, G- α -BWO 算法进一步在种群初始化过程中引入准对立学习机制, 其表达式为

$$x(t) = \begin{cases} \text{rand}(\text{avg}_{i,j}, x'_{i,j}), & x_{i,j} \leq \text{avg}_{i,j} \\ \text{rand}(x'_{i,j}, \text{avg}_{i,j}), & x_{i,j} > \text{avg}_{i,j} \end{cases}, \quad (28)$$

式中, $\text{avg}_{i,j} = (b_{i,j} - a_{i,j})/2$, $b_{i,j}$ 指个体 i 在 j 维上的搜索上界, $a_{i,j}$ 为搜索下界, $\text{rand}()$ 为随机选择函数。

G- α -BWO 算法种群初始化过程如下: 首先利用 PWLCM 与准对立学习机制求解, 将得到的 N 个对立解进行合并, 则得到种群规模为 $2N$ 的初始种群; 然后对该 $2N$ 初始种群按适应度大小升序排列, 选择排序前 N 的一半种群个体作为 BWO 算法最终的初始种群。

为了验证 PWLCM 的优势, 引入同为混沌映射的 Logistic 和 Tent 两种方式进行对比。图 5 为 3 种混沌映射的取值频次分布。Logistic 在两个边界 $[0, 0.1]$ 和 $[0.9, 1]$ 上的取值概率明显高于中间值的取值概率, Tent 的取值频次分布均匀性相对不足, 而本文选择的 PWLCM 的取值频次分布均匀性明显更好, 这有利于生成更加均匀的初始种群, 实现空间搜索更好的遍历, 提升算法的搜索效率。

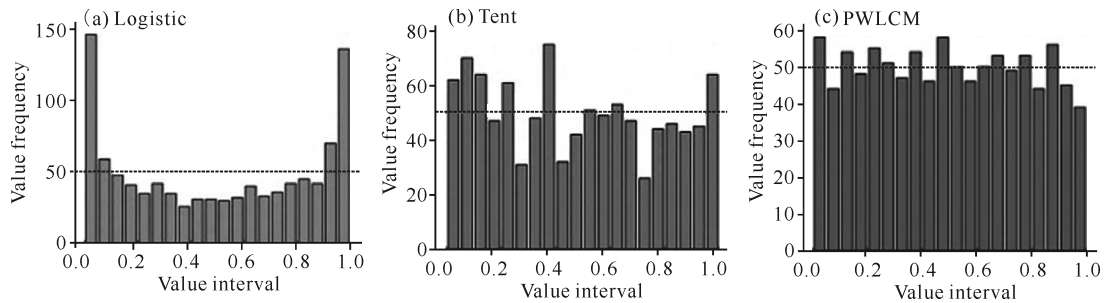


图 5 3 种混沌映射的取值频次分布

Fig. 5 Frequency distribution of values for three types of chaotic mapping

3.2.2 自适应高斯云策略

对于 BWO 算法而言, 同步或镜像方式的成对游行起到了全局寻优的作用。然而, 随着迭代的进行, 镜像种群在每个维度上的位置会根据选择的随机个体缩小搜索范围, 这样会导致算法在迭代后期的种群多样性逐步减小, 无法得到精度更高的最优解。高斯云是基于高斯分布的随机云模型, 由若干云滴构成, 模拟了自然界普遍存在的不确定性和分形现象^[20]。高斯云包含 3 个参数: 期望 E_x 、熵 E_n 和超熵 H_e 。 E_x 定义云滴的分布期望, E_n 定义云滴的不确定性, H_e 定义 E_n 的不确定性。其隶属度函数为

$$\mu = \exp[-(x - E_x)^2 / (2(E_n)^2)]. \quad (29)$$

图 6 为 3 种参数取值下的高斯云模型分布。可

见, 对于确定的期望 E_x , 熵 E_n 越大, 高斯云跨度越广; 而超熵 H_e 越大, 则高斯云显得更分散。

结合高斯云的模糊性和随机性特征, G- α -BWO 算法引入自适应高斯云模型对 BWO 算法的位置更新方式进行改进。通过按概率利用高斯云生成新的白鲸个体位置, 算法既可在小范围搜索, 又同时具备一定概率远离当前位置搜索, 从而扩大整体搜索范围。基于高斯云模型的位置更新方式定义为

$$X'_i(t+1) = \begin{cases} X'_i(t)(1 + \text{Gauss}(0, \delta^2, 0.1\delta^2)), & \text{rand} < 0.5 \\ X'_i(t), & \text{rand} \geq 0.5 \end{cases}, \quad (30)$$

式中, $X'_i(t)$ 为式(21)计算的更新位置, $X'_i(t+1)$ 为高斯云生成位置, $rand$ 为(0,1)的均匀分布随机数, $Gauss(0, \delta^2, 0.1\delta^2)$ 为高斯云算子, 即符合 $(E_x, E_n, H_c) = (0, \delta^2, 0.1\delta^2)$ 的高斯云随机量。同时, 为了使个体的位置更新具有自适应性, 将参数 δ 定义为

$$\delta^2 = \begin{cases} 1, \text{fit}(i) \leq \text{fit}(rand) \\ \exp\left(\frac{\text{fit}(rand) - \text{fit}(i)}{|\text{fit}(i)| + \zeta}\right), \text{fit}(i) > \text{fit}(rand) \end{cases}, \quad (31)$$

式中, $\text{fit}(i)$ 为个体 i 的适应度, $\text{fit}(rand)$ 为随机个体的适应度, ζ 为极小值。可见, 若个体 i 适应度 $\text{fit}(i)$ 优于随机个体适应度 $\text{fit}(rand)$, 则算法将扩展搜索区域进行位置更新; 否则, 算法将收缩搜索区域, 提升局部精细开采精度。

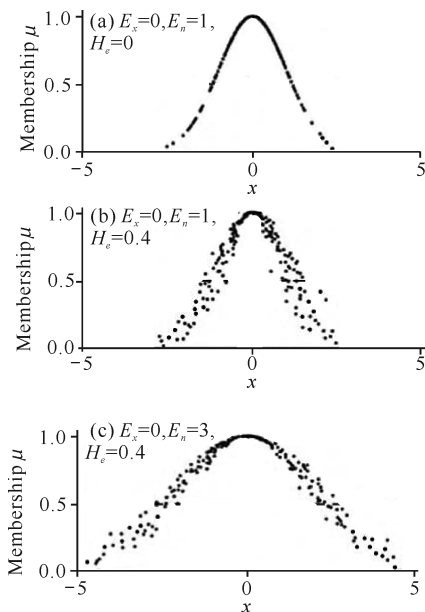


图6 高斯云分布

Fig. 6 Gaussian cloud distribution

3.2.3 自适应 α 分布变异

个体变异扰动是智能优化算法迭代求解中跳离局部最优的常用方法, 已有研究中广泛采用了基于柯西分布或高斯分布的个体变异方式, 但这种单一的变异方式无法让算法在不同迭代时期内随性能自适应调整而展现出不同的变异能力, 从而限制了算法多样性的提升。为了使 BWO 算法在不同迭代时期具有不同步长的变异能力, $G-\alpha$ -BWO 算法引入一种自适应 α 分布变异方式对最优解进行变异扰动。

α 分布是一种包含无限方差和无穷独立同分布随机变量的极限分布, 其概率密度分布函数定义为

$$f(x; \alpha, \theta, \sigma) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \psi(t) e^{-itx} dx, \quad (32)$$

式中, $\psi(t)$ 为特征函数, 定义为

$$\psi(t) = \exp(i\theta t - |\sigma t|^\alpha B_{t,\alpha}), \quad (33)$$

$$B_{t,\alpha} = \begin{cases} 1 + i\theta \text{sgn}(t) \frac{2}{\pi} \ln(t), \alpha = 1 \\ 1 - i\theta \text{sgn}(t) \tan(\pi\alpha/2), \alpha \neq 1 \end{cases}, \quad (34)$$

式中, α 为特征指数, θ 为对称因子, δ 为尺度因子。 α 用于控制随机过程的脉冲度, 取值越小, 脉冲越大, 即分布尾翼越长, 表示远离中心的随机量概率越大, 通常情况下, $0 < \alpha < 2$; θ 用于控制分布的倾斜度; σ 类似于方差值。当 $\alpha=1, \theta=0, \sigma=1, \delta=0$ 时, α 分布表示柯西分布; 当 $\alpha=2, \theta=0, \sigma=1, \delta=0$ 时, α 分布表示高斯分布。

$G-\alpha$ -BWO 算法利用 α 分布生成的随机算子对种群迭代时的全局最优解进行变异, 具体方式为

$$X_{\text{best,new}}(t) = X_{\text{best}}(t) + X_{\text{best}}(t) \cdot S(\alpha), \quad (35)$$

$$\alpha = -\exp\left(-6 \tan\left(\frac{t}{T_{\max}}\right)^4\right) + 2, \quad (36)$$

式中, $X_{\text{best}}(t)$ 、 $X_{\text{best,new}}(t)$ 分别指原始最优解和 α 分布变异后的最优解, $S(\alpha)$ 为 α 分布算子, 特征指数 α 用于控制变异步长。为了使算法在不同迭代时期具有不同程度的变异能力, 将算子 $S(\alpha)$ 设计为随迭代次数变化的正切函数变换形式。图7为 α 随迭代次数的变化曲线及不同 α 值条件下的概率分布函数。

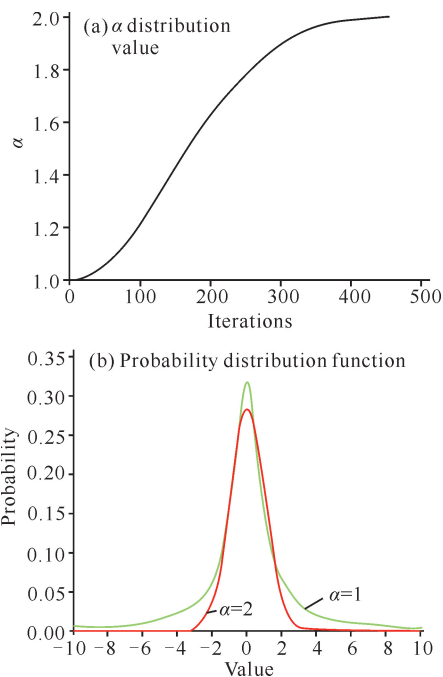
图7 α 分布取值及概率分布函数Fig. 7 α distribution value and probability distribution function

图7表明, 迭代早期, α 值较小, 上升趋势缓慢, 此时 α 分布变异更接近于柯西算子, 易生成较大的离

群值与极端值,利于算法迭代早期扩大搜索范围。而到了迭代后期, α 值呈非线性递增,变异值逐步缩小,此时 α 分布变异更接近于高斯算子,利于算法迭代后期精细开采,提高收敛精度,并加快算法收敛。由此可见,自适应 α 分布变异可依据算法迭代进程赋予算法不同的变异能力,提升算法的整体寻优性能。

图8为G- α -BWO算法执行流程。

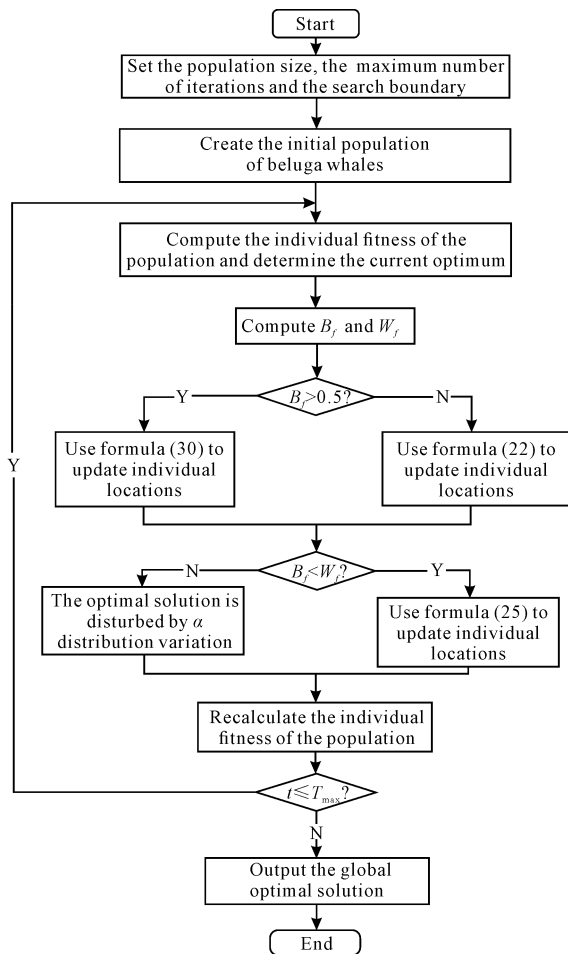


图8 G- α -BWO算法

Fig. 8 G- α -BWO algorithm

4 G- α -BWO算法求解巡检无人机航迹规划

所有航迹点组成无人机飞行路径,G- α -BWO算法的一个种群个体可视为一条航迹规划的候选解,以航迹总代价函数[式(18)]作为G- α -BWO算法的适应度函数,执行G- α -BWO算法的迭代搜索流程。算法具体步骤如下。

步骤1:设置种群规模 N 、最大迭代次数 T_{\max} 和个体搜索边界。

步骤2:设置巡检任务的三维地貌图、飞行起点和终点位置,以及山体模型、障碍物威胁区域模型及

威胁半径,建立航迹规划代价函数和约束条件,设置航迹点数量、单程最大航程、最大转弯角和爬升角。

步骤3:在搜索边界内利用PWLCM-准对立学习机制生成白鲸的初始种群结构,对应于初始的航迹候选解。

步骤4:利用式(18)计算种群个体位置(航迹解)的适应度值,确定当前的最优解。

步骤5:利用式(20)计算平衡因子 B_f 和坠落因子 W_f ,若 $B_f > 0.5$,则进行全局搜索,利用高斯云策略式(30)更新航迹;若 $W_f < B_f \leq 0.5$,则进行局部开发,利用式(22)更新航迹。

步骤6:当 $B_f < W_f$ 时,种群进入坠落阶段,利用式(25)更新航迹。

步骤7:若 $B_f \geq W_f$,则利用自适应 α 分布变异策略对当前最优解变异。

步骤8:重新计算个体适应度。

步骤9:若 $t \leq T_{\max}$,则返回步骤4执行;否则,算法结束,输出航迹最优解。

5 实验与结果分析

5.1 算法寻优测试

利用Sphere、Rastrigin、Griewank、Alpine 4个函数测试算法的搜索性能,函数三维图像如图9所示,理论最优解均为0。Sphere为非线性对称单峰函数,搜索域内有且仅有1个严格全局最优解。Rastrigin为复杂多峰函数,搜索域内具有大量局部极值点,且以正弦波跳跃,算法极易落于局部极值点,较考验全局搜索能力。Griewank同为复杂多峰函数,且其局部极值点与全局最优解极其邻近,较考验算法对局部最优的判断能力。Alpine函数沿着自变量方向会产生大量可微的局部极值,搜索最优解难度极大。通过4个不同特征的函数可以综合检验G- α -BWO算法的寻优能力。为了对比算法性能,同时引入标准BWO算法、粒子群优化(Particle Swarm Optimization, PSO)算法和一种改进BWO(Improved Beluga Whale Optimizer, IBWO)算法^[16]进行算法性能的对比如分析。PSO算法是经典智能算法,引入该算法作横向对比,分析BWO算法的优势;引入BWO算法则可对比分析改进前后算法性能是否得到提升;而引入IBWO算法则可以对比分析本文所设计的改进策略相比已有文献的改进策略是否具备性能优势,为纵向对比分析。种群规模均设为 $N=20$,最大迭代次数 $T_{\max}=500$,混沌因子 $p=0.4$ 。

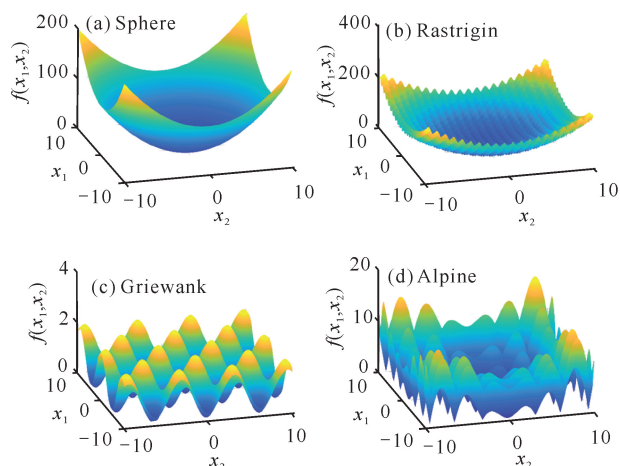


图9 测试函数三维图像

Fig.9 3D images of test functions

表1 寻优结果

Table 1 Optimization result

| 算法 Algorithm | Sphere | | | Rastrigin | | | Griewank | | | Alpine | | |
|----------------------|----------------------------|----------------------|------------|----------------------------|----------------------|------------|----------------------------|----------------------|------------|----------------------------|----------------------|------------|
| | 最优解 Optimal solution | 平均值 Mean value | 标准差 Std | 最优解 Optimal solution | 平均值 Mean value | 标准差 Std | 最优解 Optimal solution | 平均值 Mean value | 标准差 Std | 最优解 Optimal solution | 平均值 Mean value | 标准差 Std |
| PSO | 5.24E-10 | 1.56E-04 | 2.33E-04 | 1.04E-08 | 3.36E-02 | 4.30E-02 | 5.37E-01 | 8.17E+01 | 2.94E+01 | 2.82E-01 | 1.17E+00 | 2.59E-01 |
| BWO | 4.15E-46 | 9.67E-28 | 1.37E-27 | 3.18E-26 | 9.47E-17 | 9.65E-17 | 3.02E-09 | 1.73E-05 | 2.35E-05 | 4.65E-09 | 1.58E-06 | 1.05E-06 |
| IBWO | 0.00E+00 | 2.89E-197 | 0.00E+00 | 1.37E-104 | 2.11E-99 | 4.09E-100 | 0.00E+00 | 2.54E-196 | 0.00E+00 | 5.86E-101 | 1.17E-98 | 1.72E-99 |
| G- α - BWO | 0.00E+00 | 5.36E-230 | 0.00E+00 | 0.00E+00 | 1.76E-118 | 1.12E-118 | 0.00E+00 | 8.14E-201 | 0.00E+00 | 0.00E+00 | 6.86E-107 | 7.63E-107 |

5.2 航迹规划实验

在规模为 $50 \times 50 \times 0.20$ (单位: km) 的三维地形空间内建立无障碍物、简单障碍物和复杂障碍物 3 种场景, 展开多层面实验测试。其中, 无障碍物场景为仅有 4 座山体的基准地形; 简单障碍物场景在无障碍物场景的基础上, 再部署 3 个障碍物威胁源; 复杂障碍物场景中部署有 6 座山体及 6 个障碍物威胁源。从搜索空间看, 3 个场景逐步压缩了算法搜索可行航迹的区域, 极其考验算法的鲁棒性和稳定性。设置起点坐标 $(2, 2, 0.02)$, 终点坐标 $(50, 50, 0.05)$, 单位为 km, 航迹点数量为 5, 高度区间为 $[0.01, 0.20]$, $\gamma_c = 0.4$, $\gamma_h = 0.8$, $\gamma_{le} = \gamma_{re} = 0.5$, 单程最大航程约束 $L_{\max} = 150$ km, 最大转弯角 $\alpha_{\max} = 60^\circ$, 最大爬升角 $\beta_{\max} = 60^\circ$, $\omega_1 = \omega_4 = 0.3$, $\omega_2 = \omega_3 = 0.2$, 赋予航程大小和威胁代价较高权重, 以确保生成更短更安全的路径, 同时兼顾考虑飞行高度和飞行转角代价。算法种群规模统一设为 20, 最大迭代次数为 500。同时将规划航迹以 B 样条曲线进行平滑处理。

在无障碍物场景中, 4 座山体中心坐标分别为 $(30, 44)$ 、 $(23, 30)$ 、 $(20, 10)$ 、 $(45, 22)$, 山体峰高对应

表 1 为算法在函数上独立运行 30 次的寻优结果。从搜索的最优解看, G- α -BWO 算法成功搜索到 4 个函数的理论最优解。可见, 对不同复杂性的单峰函数和多峰函数, 该算法都具有更强的搜索能力, 能够适应于不同特征的函数极值求解问题, 稳定性和鲁棒性更好。在对比算法中, 除 IBWO 算法可在 Sphere 和 Griewank 函数上搜索到最优解外, 其他算法均不同程度地陷入局部最优, 反映出这些算法在迭代过程中难以开辟新的搜索区域, 从而无法逼近最优解。标准差也反映出 G- α -BWO 算法的稳定性, 这表明该算法所引入的高斯云与 α 分布变异策略在防止种群多样性逐步贫化而易生成局部最优解上能够起到很好的抑制作用, 使算法提升了搜索精度。

0.11、0.12、0.13、0.14 km。图 10 是 G- α -BWO 算法在无障碍物场景下前 30 次迭代搜索的航迹规划结果。可见, 分布于山体两侧的航迹较少, 分布于中间位置的航迹居多, 该区域显然属于离全局最优解较近的区域, 这反映出随着搜索的迭代, G- α -BWO 算法有能力通过更好的搜索机制引导种群向全局最优解逼近, 证明该改进算法有效可行。

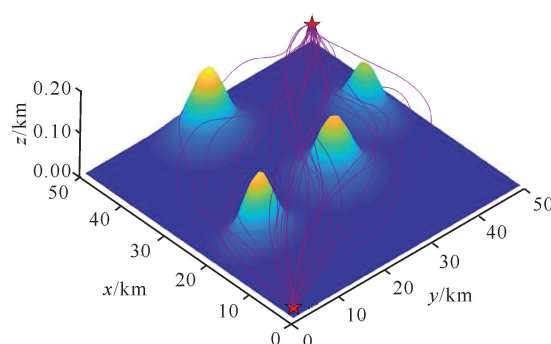


图10 无障碍物场景 30 次迭代航迹示意图

Fig. 10 Trajectory diagram of 30 iterations of non-obstacle scene

图 11 为算法完成迭代后在无障碍物场景下的航迹规划解。4 种算法均可以得到规划航迹可行解, 表

明在无障碍物环境下算法均是可行的,均能够成功避开山体碰撞,但航迹走向略有不同。3种对比算法均选择从起点邻近位置山体绕行较远的路线,航程偏大;G- α -BWO算法则在两个邻近山体间成功搜索到更优解。PSO算法航程较大,BWO和IBWO算法相对具有更大的飞行高度,而G- α -BWO算法航程最小。

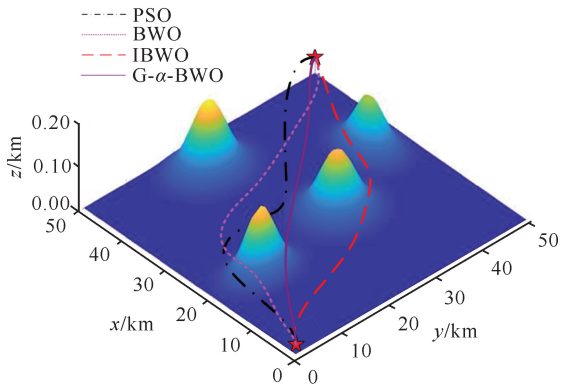


图 11 无障碍物场景侧视图

Fig. 11 Side view of non-obstacle scene

在简单障碍物场景中,山体位置、高度与无障碍物场景相同。3个障碍物位置坐标分别为(25,9)、(15,33)、(40,40),威胁半径对应5、8、5。图12为算法在简单障碍物场景下的航迹规划解。由图12可见,各算法依然可以找到航迹规划的可行解。G- α -BWO算法在绕过两座山体抵达终点前成功避开障碍物威胁,得到了最优航迹,而3种对比算法的航程明显大于G- α -BWO算法。

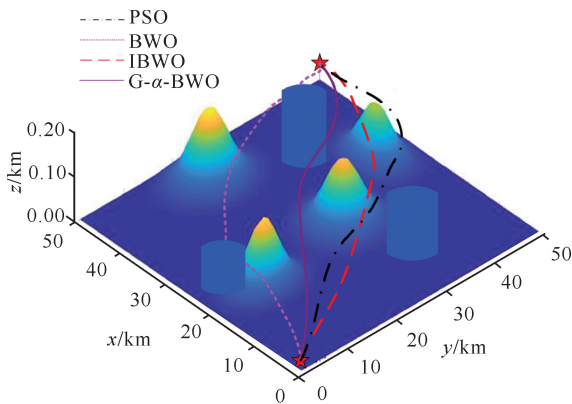


图 12 简单障碍物场景侧视图

Fig. 12 Side view of simple obstacle scene

在复杂障碍物场景中,6座山体中心坐标分别为(15,7)、(28,42)、(36,33)、(20,20)、(29,30)、(41,20),山体峰高分别对应为0.10、0.12、0.14、0.15、0.17、0.16 km,6个障碍物的位置坐标分别为(8,12)、(30,14)、(18,27)、(40,30)、(9,38)、(29,44),威

胁半径分别为4.0、5.0、5.0、3.0、5.0、5.5,图13为算法在复杂障碍物场景下的航迹规划解。此场景中山体和障碍物部署较多,算法不仅容易陷入局部最优航迹,而且容易得到与障碍物发生碰撞的不可行航迹。如PSO算法就与左侧圆柱体障碍物发生穿越。G- α -BWO算法成功在山体与障碍物间找到最优解,表明在自适应高斯云和 α 分布变异策略下算法搜索最优航迹的成功率更高。

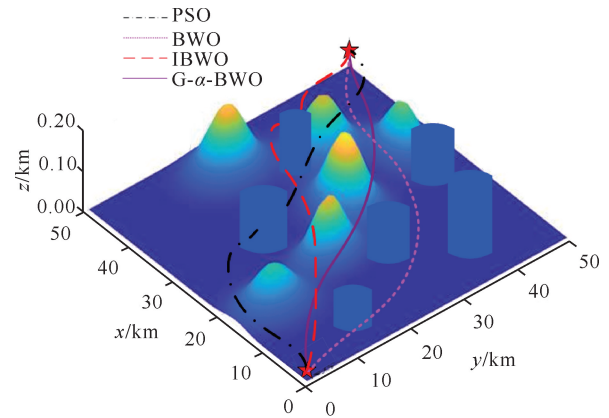


图 13 复杂障碍物场景侧视图

Fig. 13 Side view of complex obstacle scene

除了搜索精度,算法搜索最优解的速度也同样重要。图14显示了复杂障碍物场景下的算法收敛速度。初始迭代处对应的代价值体现算法的初始种群质量。相对于对比算法,G- α -BWO算法的初始航迹代价更小,说明初始个体分布所代表的航迹规划解也较优,证实PWLCM-准对立学习种群初始化策略能够提升初始航迹解的质量。同时,G- α -BWO算法不仅得到的航迹代价最小,而且还能更快地进入收敛状态,这表明改进搜索机制对提升BWO算法的搜索精度和收敛速度都具有直接效果。而对比算法不同程度陷入局部最优解,无法确保丰富的搜索空间和种群

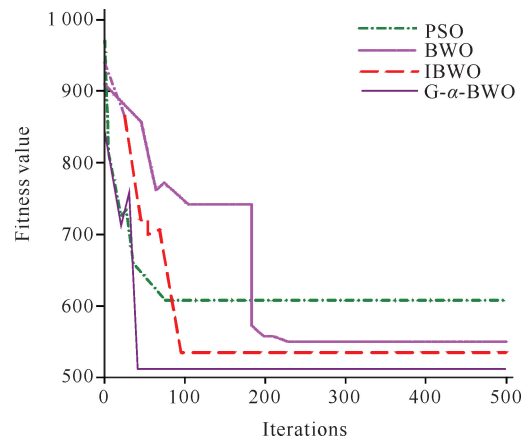


图 14 收敛曲线(复杂障碍物场景)

Fig. 14 Convergence curve (complex obstacle scene)

多样性,其航迹规划并不是最优的。

表 2 是 3 种场景下算法运行 30 次的指标统计结果。3 个场景中 G- α -BWO 算法在目标代价函数的最优解、最差解、平均值及标准差上均表现更好。随着场景复杂度的增加,G- α -BWO 算法在最优航迹求解上的优势甚至能够有所提升。而 G- α -BWO 算法在 3 种场景下的标准差更小,表明该算法不仅对最优解的寻优精度更高,而且稳定性也更好,能够搜索到质量更好的航迹规划解。表 2 还统计了目标函数中各代

价分量的均值结果,即航程代价、高度代价、转角代价及障碍物威胁代价的均值结果。在航迹代价的均值上,3 个场景下 G- α -BWO 算法的代价均值分别比 PSO、BWO 和 IBWO 平均降低了 26.35%、15.88% 和 11.47%。由此可见,G- α -BWO 算法在确保总体代价更优的前提下,还能兼顾各分量代价,尤其在规划场景复杂度增加后,依然保持了算法稳定性,G- α -BWO 算法的航迹代价表现依然优于 3 种对比算法。

表 2 航迹代价统计指标

Table 2 Statistical indexes of trajectory cost

| 场景 Scene | 算法 Algorithm | 指标 Index | | | | | | | |
|---------------------------|------------------|----------------------------|------------------------------|----------------------|------------|----------------------------|------------------------|-----------------------|----------------------------|
| | | 最优解 Optimal solution | 最差解 The worst solution | 平均值 Mean value | 标准差 Std | 航程代价 Range size cost | 高度代价 Height cost | 转角代价 Angle cost | 障碍物威 胁代价 Threat cost |
| Non-obstacle scene | PSO | 485.78 | 612.24 | 549.85 | 13.63 | 101.36 | 436.41 | 349.84 | 426.73 |
| | BWO | 440.97 | 569.67 | 485.52 | 11.52 | 95.77 | 429.79 | 328.52 | 402.19 |
| | IBWO | 421.72 | 540.39 | 449.76 | 11.02 | 90.05 | 429.58 | 340.39 | 379.74 |
| | G- α -BWO | 346.19 | 434.11 | 381.56 | 9.14 | 81.38 | 423.07 | 313.56 | 347.26 |
| Simple obstacle scene | PSO | 650.27 | 779.14 | 712.39 | 15.46 | 113.61 | 442.63 | 357.16 | 479.31 |
| | BWO | 568.27 | 668.60 | 627.37 | 14.32 | 106.34 | 436.33 | 346.35 | 435.89 |
| | IBWO | 556.24 | 658.36 | 606.74 | 11.49 | 99.56 | 430.49 | 347.04 | 427.75 |
| | G- α -BWO | 540.34 | 609.14 | 554.85 | 8.51 | 93.04 | 426.18 | 329.55 | 396.03 |
| Complex obstacle scene | PSO | 725.18 | 864.05 | 795.64 | 17.25 | 121.49 | 451.72 | 369.43 | 627.72 |
| | BWO | 656.07 | 708.46 | 686.94 | 17.44 | 115.38 | 441.67 | 356.48 | 572.11 |
| | IBWO | 605.19 | 693.45 | 656.37 | 15.31 | 107.47 | 440.11 | 352.53 | 506.04 |
| | G- α -BWO | 542.86 | 623.72 | 586.07 | 10.25 | 97.42 | 434.59 | 341.47 | 467.18 |

利用箱型图对比算法在复杂障碍物场景下的性能稳定性,结果如图 15 所示,G- α -BWO 算法的适应度函数中位值、四分位值及边界值均对比算法更低。箱型图的大小体现航迹规划求解的稳定性,即反映航迹规划代价函数值的分布。箱型图面积越小,反映解的分布越集中,稳定性越好。G- α -BWO 算法的箱型图分布更为集中,影线更短,异常值数量也更少。IBWO 算法稳定性略差,航迹规划代价波动性较大,异常值也较多。BWO 算法整体优于 PSO 算法,但也拥有较多异常值。综合来看,G- α -BWO 算法能够更快找到适应度更优的候选解,同时快速收敛于航迹规划最优解。

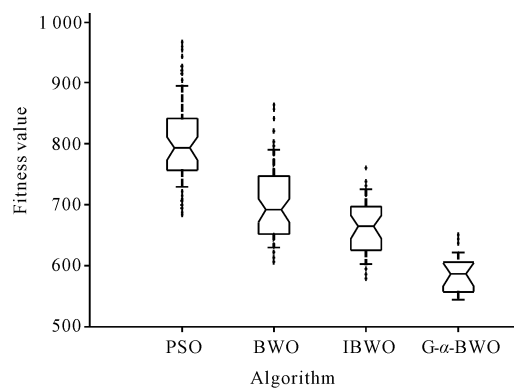


图 15 复杂障碍物场景箱型图

Fig. 15 Complex obstacle scene box diagram

6 结论

为了解决复杂环境中电力巡检无人机航迹规划容易出现精度差、得到局部最优航迹的问题,本文提出一种巡检无人机航迹规划算法—— $G-\alpha$ -BWO算法。在提升算法寻优性能方面,引入PWLCM-准对立学习初始化提升初始航迹和质量,利用自适应高斯云策略提升算法的全局最优航迹搜索能力,设计自适应 α 分布变异策略使算法跳离局部最优航迹,丰富搜索空间。本文建立了考虑航程大小、航程高度、航迹转角和障碍物威胁的代价函数,利用 $G-\alpha$ -BWO算法求解巡检最优航迹。结果表明,该算法搜索的航迹不仅可以安全避障,而且具有更小的航迹代价,证实改进策略有效可行。

参考文献

- [1] 李刚,智宏鑫. 电力巡检机器人路径规划方法综述[J]. 电力科学与工程,2024,40(4): 1-11.
- [2] 戴永东,王永强,高超,等. 电力输电线路无人机巡检航线智能规划方法[J]. 重庆理工大学学报(自然科学), 2023,37(9):253-260.
- [3] 黄宸希,韩韬,吴雪琼,等. 基于改进A*算法的机器人路径规划与避障技术[J]. 电子设计工程,2024,32(1): 24-28.
- [4] 朱熠,田辉,郝向宇,等. 基于数字地形平滑和人工势场的无人机近地三维路径规划[J]. 兵器装备工程学报, 2024,45(1):265-271.
- [5] 张志威,贾云伟,王永霞,等. 基于改进的快速扩展随机树的快速路径规划算法[J]. 天津理工大学学报,2022, 38(3):14-19.
- [6] 田周泰,柴梦娟,刘广怡,等. 基于改进人工鱼群算法的无人机三维航迹规划[J]. 信息工程大学学报,2024, 25(1):80-84.
- [7] 郭志明,娄文忠,李涛,等. 基于改进蝗虫优化算法考虑任务威胁的多无人机协同航迹规划[J]. 兵工学报, 2023,44(S2):52-60.
- [8] 王康,司鹏,陈莉,等. 基于改进沙猫群算法的无人机三维航迹规划[J]. 兵工学报,2023,44(11):3382-3393.
- [9] 丁敏,夏兴宇,邹永杰,等. 基于改进蝴蝶优化算法的无人机3-D航迹规划方法[J]. 南京航空航天大学学报, 2023,55(5):851-858.
- [10] YAN Z P,ZHANG J Z,YANG Z W, et al. Two-dimensional optimal path planning for autonomous underwater vehicle using a whale optimization algorithm [J]. Concurrency and Computation: Practice and Experience,2020,33(9): e6140.
- [11] GUAN Y R,GAO M S,BAI Y F. Double-ant colony based UAV path planning algorithm [C]//ICMLC'19: Proceedings of the 2019 11th International Conference on Machine Learning and Computing. New York, United States: Association for Computing Machinery,2019: 258-262.
- [12] LIANG X M,KOU D C,WEN L. An improved chicken swarm optimization algorithm and its application in robot path planning [J]. IEEE Access,2020,8:49543-49550.
- [13] CHEN J,CHEN X,FU Z F. Improvement of the seagull optimization algorithm and its application in path planning [J]. Journal of Physics: Conference Series, 2022,251:109215.
- [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] ZHONG C T,LI G,MENG Z. Beluga whale optimization: a novel nature-inspired metaheuristic algorithm [J]. Knowledge-based Systems,2022,12(4):1-12.
- [16] 孔云,周学良,冷杰武. 基于改进白鲸优化算法的低碳柔性工艺规划[J]. 现代制造工程,2024(1):80-88.
- [17] 孟冠军,黄江涛,魏亚博. 混合白鲸优化算法求解柔性作业车间调度问题[J]. 计算机工程与应用,2024, 60(12):325-333.
- [18] 陈奕君,郑嘉利,李芷芊,等. 基于改进型白鲸算法的RFID网络规划[J]. 计算机科学,2024,51(3): 317-325.
- [19] PENG J,JIN S Z,ZHANG D, et al. S-Box construction method based on the combination of quantum chaos and PWLCM chaotic map [J]. Journal of Cognitive Informatics and Natural Intelligence,2021,15(4):1-17.
- [20] 窦唯,刘晓阳. 基于小波包分解与高斯云模型的故障诊断方法[J]. 机械设计与制造工程,2023,52(1):107-111.

A Adaptive Planning Algorithm for the Trajectory of Patrol Unmanned Aerial Vehicle

SHI Chengyu¹, WANG Tianjiao¹, ZHANG Yingying¹, GU Yuexin¹, FENG Xiao^{1,2**}

(1. State Grid Electric Power Space Technology Company Limited, Beijing, 102200, China; 2. School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing, 100876, China)

Abstract: The conventional methods have poor accuracy and tend to converge to a local optimal trajectory in the trajectory planning for Unmanned Aerial Vehicle (UAV) in complex environments. In view of these problems, an adaptive Gaussian cloud and α stable distribution variation Beluga Whale Optimizer (G- α -BWO) algorithm for tracking UAV is proposed. Piecewise Linear Chaotic Map (PWLCM)-based quasi-oppositional learning initialization is introduced to improve the diversity and quality of the initial trajectory solutions. An adaptive Gaussian cloud strategy is used to improve the search ability of the algorithm for the global optimal trajectory. An adaptive α stable distribution variation is designed to make the algorithm skip a local optimal trajectory, enrich the search space and improve the convergence speed. Firstly, the trajectory planning model is constructed for patrol UAV, and the weight cost function considering the flight range, height, angle and obstacle threat is designed. The three-dimensional trajectory planning problem is transformed into a multi-objective optimization problem with constraints. Then, the optimal trajectory of patrol UAV was iteratively searched by G- α -BWO. The results showed that the improved algorithm avoided obstacles and had less cost, being helpful to improve the efficiency and security of patrol tasks.

Key words: patrol UAV; trajectory planning; Gaussian cloud; α stable distribution; Beluga Whale Optimizer (BWO)

责任编辑:梁 晓



微信公众号投稿更便捷

联系电话:0771-2503923

邮箱:gxxk@gxas.cn

投稿系统网址:<http://gxxk.ijournal.cn/gxxk/ch>