

# 基于混沌灰狼优化的多无人机协同航路规划



吴坤<sup>1</sup>, 池沛<sup>1,2</sup>, 王英勋<sup>1,2</sup>, 侯琳<sup>3</sup>

1. 北京航空航天大学, 北京 100191

2. 北京航空航天大学飞行器控制一体化技术国防科技重点实验室, 北京 100191

3. 航空工业西安飞行自动控制研究所飞行器控制一体化技术国防科技重点实验室, 陕西 西安 710065

**摘要:**针对多无人机协同执行饱和和攻击任务的时空约束, 本文提出了基于混沌灰狼优化的离线航路规划方法, 实现了多无人机协同航路的有效生成。首先, 针对饱和和攻击任务的特点进行分析, 将具有时空约束的多无人机同时到达问题转化为协同航程问题; 其次, 针对将混沌映射引入灰狼优化算法中, 提出了混沌灰狼优化算法, 以提高原算法的探索能力和收敛速度; 最后, 提出了基于几何规划的航点扩展策略, 从而构造出满足任务攻击时间与攻击方位要求的航点序列。通过对单无人机航路规划问题的仿真实验验证了改进算法的寻优能力; 通过对面向饱和和攻击任务的航路规划仿真实验验证了所提方法的可行性和有效性。

**关键词:** 多无人机; 饱和和攻击任务; 协同航路规划; 混沌灰狼优化; 几何规划

中图分类号: V249

文献标识码: A

DOI: 10.19452/j.issn1007-5453.2022.10.011

航路规划(path planning)是根据任务需求规划出满足约束条件的飞行航路, 是任务规划的关键技术之一<sup>[1]</sup>。航路规划的精度与速度能够直接对任务规划的效果产生影响。航路规划系统基本框架由环境信息、航路规划目标、机体物理约束以及航路规划器组成。根据平台规模, 无人机航路规划分为单机航路规划、多机航路规划和有人/无人协同航路规划<sup>[2-3]</sup>; 根据环境信息, 可分为确定环境与突发威胁环境下的航路规划; 根据实现功能可划分为静态航路规划与动态航路规划, 其中前者为离线预规划, 后者为在线实时规划。

受计算能力的约束, 航路离线预规划问题由于不受计算时间约束, 其更关注在可预测的已知任务环境下, 如何得到评价最优的航路结果。而在线实时规划则是在已有规划结果的基础上, 针对突发意外情况进行重新规划, 对规划时间要求较高。

航路规划算法根据计算方法可以归纳为最优化方法与启发式搜索算法(HSA)。常用的最优化算法包括枚举法(EM)、牛顿法(NM)、梯度下降法(GD)、数学规划(MP)以及

动态规划(DP)方法。

最优化方法在空间和时间上具有很强的复杂度, 当规划范围增大、环境变复杂时, 其求解时间与求解难度也随之急剧增加。而启发式算法是根据直观经验构造的一种搜索算法, 在可接受的时间和空间复杂度下, 能够得到待优化问题的一个可行的近似最优解。启发式搜索算法可进一步分为确定型搜索算法与随机型搜索算法两类。

通常确定型航路规划算法首先对规划区域进行栅格化处理, 然后利用图论在地形网格的基础上搜索从起点到目标点的最短路径。确定性方法主要包括Dijkstra算法、Floyd算法<sup>[4]</sup>、A\*算法<sup>[5-6]</sup>、人工势场法(APF)<sup>[7]</sup>以及Voronoi图法<sup>[8-9]</sup>等。随机型搜索方法通过在求解空间中进行随机采样, 并按照一定的搜索规则寻找可行的较优解。典型的随机型搜索方法有粒子群算法<sup>[10]</sup>、蚁群算法<sup>[11]</sup>、快速扩展随机树(RRT)算法<sup>[12]</sup>、人工神经网络(ANN)<sup>[13]</sup>、禁忌算法(TS)<sup>[14]</sup>以及强化学习(RL)<sup>[15]</sup>等算法。

多无人机协同航路规划可以同时为多个无人机规划出多条可行航路, 使每架无人机能够沿不同的航路飞行。这

收稿日期: 2022-02-16; 退修日期: 2022-04-15; 录用日期: 2022-05-11

基金项目: 航空科学基金(20185851021)

引用格式: Wu Kun, Chi Pei, Wang Yingxun, et al. Cooperative path planning for multi-UAVs based on chaos gray wolf optimization[J]. Aeronautical Science & Technology, 2022, 33(10):82-95. 吴坤, 池沛, 王英勋, 等. 基于混沌灰狼优化的多无人机协同航路规划[J]. 航空科学技术, 2022, 33(10):82-95.

样生成的航路尽管对每架无人机来说不一定是最优的,但对整个多无人机系统来说是最优或次优的。参考文献[16]对于多无人机对地面区域的最小时间覆盖问题,首先采用K均值聚类对侦察区域进行划分,然后采用并行GA算法对每个簇中的航路规划问题进行计算。参考文献[17]提出一种分解策略,将多无人机协同航路规划问题分为三层框架结构,第一层采用Dubins为每架无人机规划可飞航路;第二层对多机航路的空间位置进行调整,生成避障与防撞航路;第三层对每条航路的长度进行调整,使得所有的航路长度近似相等,以实现同时到达的时间协同目的。参考文献[18]提出了三阶段航路预测算法(TPP)算法,首先利用A\*算法对任务航路进行估计,然后根据预估结果为每架无人机分配代价最优的航路结果,最后利用三次B样条差值算法实现可飞路径的规划。

多无人人在协同执行任务过程中,要求多无人机应保持更加严格的占位和和攻击方向。本文以有人/无人协同执行多方向饱和攻击任务为研究背景,有人机担任主要攻击决策者,多无人机根据任务约束规划出满足要求的可行飞行航路。首先针对饱和攻击任务特点进行分析,给出了面向任务需求的航路评价指标;其次针对任务的时空约束提出了基于几何规划的航路点扩展策略;最后利用改进的灰狼优化算法对离线航路规划问题进行求解,得到满足要求的规划结果。

## 1 面向饱和攻击任务的问题描述

### 1.1 航路规划问题描述

当多无人机在协同执行任务过程中,在满足任务约束的同时,每架无人机不仅需要具有自动规避威胁和障碍物的能力,还应当具有与团队内其他成员的防撞能力。而多无人机协同航路规划技术则是解决这一问题的有效途径,它是指航路规划层根据战场的环境信息,以任务需求为导向,综合考虑无人机的机动性能、燃油消耗等约束条件,为每架无人机设计出一条不仅能够满足团队间任务协同需求,还能够使团队整体生存能力达到最大的可飞的航路<sup>[9]</sup>。

通常无人机的航路是指由一组离散点列组成的集合,即无人机 $U^{(i)}(i = 1, \dots, n)$ 的航路可以表示为

$$\text{Line}_i = \{P_0^{(i)}, P_1^{(i)}, \dots, P_m^{(i)}, P_{m+1}^{(i)}\}$$

式中, $P_0^{(i)} = P_s^{(i)}$ 与 $P_{m+1}^{(i)} = P_t^{(i)}$ 分别表示无人机 $U^{(i)}$ 的起始点与目标点。 $P_k^{(i)}(k = 1, \dots, m)$ 称为导航点,表示无人机在此点附近以一定的转弯角进行航向调整,而在两个导航点之

间则主要以一定航向进行直线飞行。如图1所示,假设无人机的最小转弯半径为 $R_{\min}^{(i)}$ ,则 $P_{k,-}^{(i)}$ 表示在航段 $\|P_{k-1}^{(i)}P_k^{(i)}\|$ 上的转弯驶出点, $P_{k,+}^{(i)}$ 表示在航段 $\|P_k^{(i)}P_{k+1}^{(i)}\|$ 上的转弯驶入点。根据 $P_{k,-}^{(i)}$ 与航点 $P_k^{(i)}$ 的位置关系可将无人机的航点切换方式分为到点飞行、内切飞行以及过点飞行三种。无人机的航线由直飞时的直线段与转弯时的圆弧段组成,则根据Dubins定理可得,采用内切飞行的航点切换方式所需的转弯次数最少且形成的航线长度最短。所以本文对航线长度的估计采用内切飞行方式,则由航点 $P_{k-1}^{(i)}$ 、 $P_k^{(i)}$ 与 $P_{k+1}^{(i)}$ 组成的航线长度为

$$\begin{aligned} \text{Length}_k^{(i)} = & \|P_{k-1}^{(i)}P_k^{(i)}P_{k+1}^{(i)}\| = \|P_{k-1}^{(i)}P_{k,-}^{(i)}\| + \\ & \|P_{k,-}^{(i)}P_{k,+}^{(i)}\| R_{\min}^{(i)} \chi_k^{(i)} = \|P_{k-1}^{(i)}P_k^{(i)}\| + \\ & \|P_k^{(i)}P_{k+1}^{(i)}\| + R_{\min}^{(i)} \chi_k^{(i)} - 2R_{\min}^{(i)} \tan \frac{\chi_k^{(i)}}{2} \end{aligned} \quad (1)$$

式中, $\chi_k^{(i)}$ 表示航点 $P_k^{(i)}$ 处的转弯角。

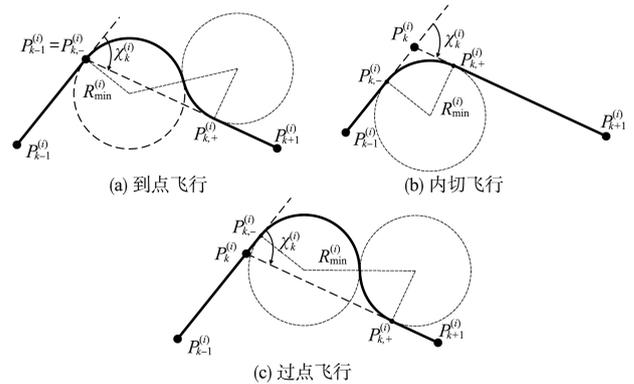


图1 航点切换方式

Fig.1 Waypoint switching mode

### 1.2 饱和攻击任务描述

当使用无人机对敌方目标进行打击时,通常由于目标强度的限制以及单架无人机的性能约束,如果采用一对一的作战模式不能形成对目标的有效打击。所以需要采用饱和攻击(SA)的作战策略对目标进行打击,即利用多架无人机对同一目标进行多方向多批次的攻击模式,从而实现一击必毁的作战效果。根据到达攻击位置的时间要求与空间要求,可将饱和攻击任务分为两种,即分批攻击任务(BAAT)与多方向攻击任务(MDAT)。其中BAAT是指多架无人机在一个规定的时间间隔 $\Delta T_a$ 内陆续到达攻击位置的贯序到达问题;而MDAT则要求多架无人机在同一时刻从多个攻击方向对目标实施打击,MDAT即为同时到达问题。所以多无人机执行SA任务的航路规划方法的关键问题在于如何将SA任务中的时间与空间约束转换到单条航路规划问题中。

### 1.3 航路评价指标

由于航路规划的目标为在满足无人机物理性能约束与飞行任务约束的前提下生成能够对威胁进行规避的飞行路径。如图2所示,可将无人机航路评价指标分为战术生存性、飞行约束性以及任务有效性三类。

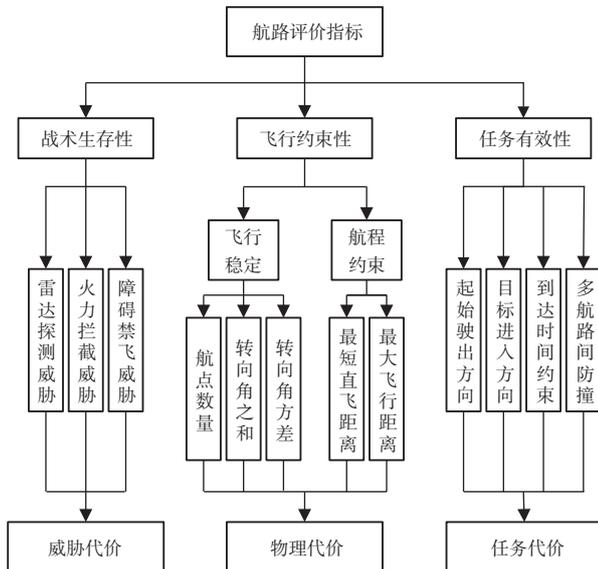


图2 典型的航路评价指标

Fig.2 Typical evaluation index of path

#### 1.3.1 战术生存性

战术生存性是指无人机对敌方威胁进行规避的能力。一般能够对无人机的飞行安全产生影响的威胁源包括:探测性威胁(如雷达)、杀伤性威胁(如地空导弹或高炮等)以及障碍物或禁飞区威胁。因为威胁程度主要与距离有关,所以在无人机航路规划问题研究中,通常将威胁区域拟合为具有威胁半径与威胁等级的圆形或矩形区域。

假设在规划空间内任意一点位置为 $p_i(x',y')$ ,则威胁源 $\text{Threat}_j$ 为探测性威胁时对 $p_i$ 的威胁程度可以表示为

$$P_{ij}^r = \begin{cases} 0 & R_{ij} > R_j^r \\ \frac{K_j}{R_{ij}} & K_j < R_{ij} \leq R_j^r \\ 1 & \text{other} \end{cases} \quad (2)$$

式中, $K_j$ 为常数,表示威胁源的探测强度; $R_j^r$ 表示探测类威胁源的威胁半径; $R_{ij}$ 为 $p_i$ 与 $\text{Threat}_j$ 间直线距离。如果 $\text{Threat}_j$ 为杀伤性威胁,则 $p_i$ 处受到的威胁值为

$$P_{ij}^k = \begin{cases} w_{\max} & R_{ij} < R^{\min_j} \\ w_{\max} \frac{R^{\max_j} - R_{ij}}{R^{\max_j} - R^{\min_j}} & R^{\min_j} < R_{ij} \leq R^{\max_j} \\ 0 & \text{other} \end{cases} \quad (3)$$

式中, $R^{\min_j}$ 与 $R^{\max_j}$ 分别为杀伤性武器的攻击近界与远界; $w_{\max}$ 为威胁源的威胁程度。

在执行任务过程中,运行航线穿过威胁区域,但禁止飞过障碍物与禁飞区,所以障碍物与禁飞区的威胁值可以表示为

$$P_{ij}^z = \begin{cases} 1 & R_{ij} \leq R_j^z \\ 0 & R_{ij} > R_j^z \end{cases} \quad (4)$$

#### 1.3.2 飞行约束性

飞行约束性是指规划的航路结果应满足无人机的物理约束,主要包括飞行稳定性以及航程约束两部分。为了使控制系统能够更加容易实现对规划航线的跟踪以及提高飞行状态的稳定性,通常需要对航路规划的航点数量、转弯角度以及相邻航点间的距离进行约束。因为无人机在导航点处进行转弯飞行,所以需要限制航点数量即应满足 $m^{(i)} \leq m_{\max}$ 。

定义无人机 $U^{(i)}$ 航路 $\text{Line}_i$ 的转弯角集合为 $\chi^{(i)} = \{\chi_1^{(i)}, \dots, \chi_{m^{(i)}}^{(i)}\}$ ,则整条航线的平滑度代价可以定义为

$$f_{\text{smooth}} = \frac{1}{m^{(i)} - 1} \sum_{j=1}^{m^{(i)} - 1} |\chi_{j+1}^{(i)} - \chi_j^{(i)}| \quad (5)$$

即通过避免进行频繁的转弯动作与转弯量而实现无人机的平稳飞行。另外,在航路规划过程中除对转弯数量进行考虑外,还需要对航程进行约束。因为无人机在完成转弯后需要一定的直飞距离使得航向稳定,特别是在经过最后一个转弯点后,需要一定的直飞距离实现目标的对准,即最小对准距离 $l_{\min}^{(i)}$ 。另外由于燃油等原因,需要对整个规划航路的最大飞行距离 $l_{\max}^{(i)}$ 进行约束。

#### 1.3.3 任务有效性

在实际执行任务过程中,无人机在起始位置通常具有一定的初始速度方向,所以在航路规划过程中,起始航向与航段 $P_0^{(i)}P_1^{(i)}$ 间的夹角应满足最大转弯角限制,即

$$\Psi_0 \in [\Psi_0 - \Delta\psi, \Psi_0 + \Delta\psi] \quad (6)$$

式中, $\Psi_0$ 为无人机的初始驶出方向; $\Delta\psi$ 为驶出方向范围限制。

根据饱和攻击任务需求,特别是多方向攻击任务的要求,无人机需要从不同的方位角度靠近目标,如图3所示, $P_e^{(i)}$ 是指无人机 $U_i$ 在最后一个导航点 $P_m^{(i)}$ 处完成转向后目标的进入点,其与目标点间的间距为最小对准距离,即有 $\|P_e^{(i)}P_t\| = l_{\min}^{(i)}$ 。 $\mu_i^{(i)}$ 为相对于目标的进入方位角,满足如下关系式

$$\begin{cases} \|P_e^{(i)}P_t\| = \|P_m^{(i)}P_t\| - R_{\min} \cdot \tan(\chi_m^{(i)}/2) \geq l_{\min}^{(i)} \\ |\arctan(P_t/P_e^{(i)}) - \mu_i^{(i)}| \leq \Delta\mu \end{cases} \quad (7)$$

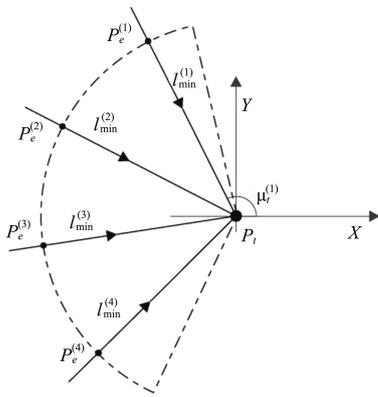


图3 目标多方向进入角限制  
Fig.3 Entry angle limit

在执行饱和攻击任务过程中,除需要对目标进入方向进行约束外,还对无人机到达攻击点的时刻有要求。定义  $U_i$  的起始时间为  $T_s^{(i)}$ , 达到目标点的时间为  $T_e^{(i)}$ , 则饱和攻击任务应满足

$$\begin{cases} T_{qs} \leq |T_s^{(i)} - T_s^{(j)}| \leq (T_{qs} + \Delta T_S) \\ T_{qt} \leq |T_e^{(i)} - T_e^{(j)}| \leq (T_{qt} + \Delta T_E) \end{cases} \quad (8)$$

式中,  $T_{qs}$  与  $T_{qt}$  分别为多无人机起始时间间隔与到达时间间隔。当  $T_{qs} = 0$  且  $T_{qt} = 0$  时的饱和攻击任务即为MDAT, 否则表示BAAT。  $\Delta T_S$  与  $\Delta T_E$  表示时间间隔误差, 即无人机从起飞到到达目标之间的时间误差应在区间  $[0, \Delta T_S + \Delta T_E]$  内。

假设每架无人机以  $v_i$  匀速飞行, 据此可将饱和攻击任务中的同时到达问题转换为航程问题, 当预期的到达时间为  $T_M$  时, 则每架无人机的总航程  $S_i$  应满足

$$S_i = v_i T_M + \Delta S_M \quad (9)$$

式中,  $\Delta S_M = (\Delta T_S + \Delta T_E)v_i$  为航程允许的误差范围, 各无人机可通过在飞行速度微调实现总航程一致的目的。

另外, 假设各无人机的速度相同时  $S_M = v_i T_M$  为定值, 则为每架无人机规划的航路长度应满足约束条件  $S_M$  即可实现同时到达的任务需求。协同航程  $S_M$  的确定应满足无人机最大航程约束条件, 即  $S_M \leq \min \{l_{max}^{(i)}\}$ 。

多无人机协同作战过程中, 除需要对时间进行约束外, 还应考虑空间中防撞问题。因为在同一规划空间内, 随着无人机数量的增多, 无人机间的航路容易出现时空重叠的现象, 即出现碰撞现象。为了避免这一现象的发生, 规划的航路在瞬时位置上应具有有一定的安全间隔, 即  $\forall i, j \in \{1, \dots, n\}$

$$\|p_i(t) - p_j(t)\| \geq d_s \quad (10)$$

式中,  $p_i(t)$  表示  $U_i$  在  $t$  时刻的位置;  $d_s$  为无人机间的安全

距离。

### 1.3.4 被控对象

多无人机协同航路规划中, 被控对象特性建模比较关键, 决定了算法性能的优劣, 本文假设无人机有一个可靠的飞行控制系统, 该系统有效地控制空气动力表面, 以精确跟踪速度和转弯率指令, 无人机的运动学方程为<sup>[20]</sup>

$$\begin{cases} \dot{x} = v \cos \psi \\ \dot{y} = v \sin \psi \\ \dot{v} = u \\ \dot{\psi} = \omega \end{cases} \quad (11)$$

式中,  $v$  和  $\psi$  分别为无人机的速度和航向,  $u$  和  $\omega$  分别为加速度和航向角速度。

## 2 基于混沌灰狼优化算法的协同规划方法

### 2.1 混沌灰狼优化算法

近年来, 引入自然进化思想的元启发式算法, 因其遵循“优胜劣汰”原则通过选择和变异来实现对问题的寻优过程, 特别适合解决较大规模的优化问题。由于元启发式算法需要通过大量的迭代计算对问题进行求解, 所以其不适用于航路的在线规划。但对于计算时间要求不高但更注重规划结果最优性的离线规划问题, 采用元启发式算法能够获得更好的规划结果。

#### 2.1.1 灰狼优化算法

灰狼优化(GWO)算法是 Mirjalili 等于 2014 年提出的一种元启发式优化算法<sup>[21]</sup>, 通过模仿自然界中灰狼群的等级制度与捕食策略以迭代计算的方式实现对猎物(最优值)的搜索。种群中灰狼个体的位置更新满足如下公式

$$X_i(t+1) = X_p(t) - A_i \cdot |C_i \cdot X_p(t) - X_i(t)| \quad (12)$$

式中:  $t$  为迭代次数;  $X_i = \{X_i^1, \dots, X_i^D\}$  表示种群中第  $i$  只灰狼的位置矢量,  $X_i^d$  为个体在第  $d$  维空间上的位置;  $X_p$  为猎物的位置矢量; 参数  $A_i$  与  $C_i$  定义为

$$A_i = 2a \cdot r_i^1 - a \quad (13)$$

$$C_i = 2 \cdot r_i^2 \quad (14)$$

式中,  $a = 2 - 2t/t_{max}$  随迭代的次数增加从 2 线性减小到 0,  $t_{max}$  表示最大迭代次数;  $r_i^1$  与  $r_i^2$  为区间  $[0, 1]$  内的  $1 \times D$  维随机矢量。则可得  $\forall A_i^d \in A_i$  有  $A_i^d \in [-a, a] \subseteq [-2, 2]$ ,  $\forall C_i^d \in C_i$  有  $0 \leq C_i^d \leq 2$ 。如图 4 所示, GWO 算法利用  $A_i^d$  与  $C_i^d$  实现了灰狼个体对猎物的开发与探索。假设  $X = \{x, y\}$  为二维空间中的某灰狼个体的位置矢量, 令猎物的位置为  $X_p = \{x_p, y_p\}$ , 则根据式(11)可得, 当  $C_i \neq I_{1 \times D}$  时, 表示对猎物周围的更优位置的进行探索, 防止陷入局部最优解。而当  $C_i =$

$I_{1 \times D}$ 时,表示灰狼个体向猎物位置(当前全局最优解)进行移动。而 $A_i$ 表示GWO算法对猎物的最优位置进行探索。因为 $a$ 从2线性递减至0,所以随机数 $A_i^d \in [-a, a]$ 的取值范围也逐渐减少,当算法迭代到一定程度时,个体只能在猎物周围小范围内进行移动。当迭代次数 $t = t_{\max}$ , $a=0$ 时,则有 $X_i(t+1) = A_i$ ,即实现整个算法对最优解的收敛。

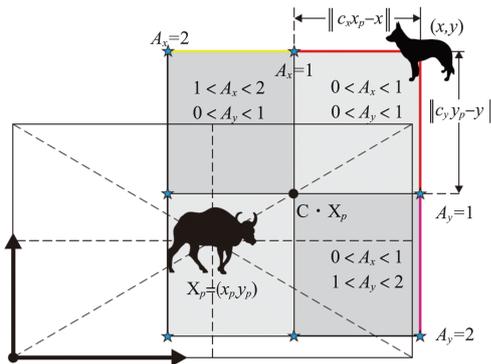


图4 灰狼优化算法原理示意图  
Fig.4 Grey wolf optimization algorithm

GWO算法通过式(11)实现了对猎物位置的探索与包围。然而在实际过程中猎物的位置是未知的,所以根据狼群中的等级制度,将当前适应度最优的前三个灰狼个体称为首领狼,分别用 $\alpha$ 、 $\beta$ 与 $\delta$ 进行表示,剩余的灰狼个体称为 $\omega$ 。在GWO算法中, $\alpha$ 、 $\beta$ 与 $\delta$ 引领整个狼群向猎物位置进行靠近。将首领狼的位置 $X_i$ , $i = \alpha, \beta, \delta$ 当成猎物的位置代入式(11)中,则可得到

$$\begin{cases} X^{(1)}(t) = X_{\alpha}(t) - A_1 \cdot |C_1 \cdot X_{\alpha}(t) - X_i(t)| \\ X^{(2)}(t) = X_{\beta}(t) - A_2 \cdot |C_2 \cdot X_{\beta}(t) - X_i(t)| \\ X^{(3)}(t) = X_{\delta}(t) - A_3 \cdot |C_3 \cdot X_{\delta}(t) - X_i(t)| \end{cases} \quad (15)$$

$$X_i(t+1) = \frac{X^{(1)}(t) + X^{(2)}(t) + X^{(3)}(t)}{3} \quad (16)$$

### 2.1.2 混沌算子

在使用群智能算法对优化问题进行求解的过程中,首先随机生成一组初始解,并选取其中适应度最小的解作为当前状态的目标解。然后种群中的其他个体通过一定的机制与策略不断向此目标解靠近。同时个体的运动方向还会发生随机偏移以实现更优解的搜索。最后在每次迭代过程中更新适应度最小解为目标解,当达到最大迭代次数后认为目标解即寻优问题的最优解。

尽管GWO算法在收敛性、全局搜索性等方面优于其他优化方法,但仍然存在易陷入局部最优解和收敛速度慢等缺点<sup>[22]</sup>。所以本节针对此问题对GWO算法进行改进,通过

引入混沌搜索策略以增加算法在解空间中的遍历性,从而提高整个算法的全局寻优能力与收敛速度。

混沌优化算法(COA)是指受混沌现象启发的一种全局性优化算法,其中混沌现象是指无固定周期的循环行为。混沌优化算法具有随机性、遍历性以及对于初始条件敏感等特点。

本文从算法参数与搜索机制两方面对GWO算法进行改进,提出了基于COA的混沌灰狼优化(CGWO)算法,以提高原算法的全局寻优能力与寻优速度。如式(13)所示,GWO算法主要利用参数 $a$ 实现了对最优解的全局搜索与局部开发能力。根据迭代次数的增加,参数 $a$ 从2线性递减至0。如图4所示,当 $a=2$ 时表示算法前期具有较强的全局寻优能力,随着 $a$ 逐渐减少算法搜索精确解的能力逐渐加强。在算法初期即搜索到最优解附近时,利用此机制能够较快地收敛到最优解。但在初期没有搜索到最优解时,容易陷入局部最优解。所以本节将GWO算法的线性递减操作改为混沌搜索,使得整个算法能够一直具有全局搜索能力。参考文献[23]已经证明采用混沌变量能够跟线性递减变量一样使算法收敛到最优解,同时具有更快收敛速度。

因为初始值能够明显地影响群智能算法的收敛速度,所以可以利用混沌变量的遍历性增加算法初期的寻优能力。假设灰狼种群的个体数量为 $N_p$ ,然后根据混沌映射可以得到 $2 \times N_p$ 组解,并对所得解的适应度进行排列,为了防止陷入局部最优解,所以选取排序后 $2 \times N_p$ 组解中的奇数项作为CGWO算法的初始解。

因为粒子群等优化算法是将当前群体中的最优解看作全局最优进行寻优的,而GWO则考虑了群体间的交流,由前三个最优解共同决定的最优解的位置,当并没有考虑个体的历史最优解。所以为了增加算法的全局搜索能力,引入粒子群算法中个体最优位置概念 $\omega_i^{(b)}$ , $i = 1, \dots, N_p$ 知道个体的寻优,则在式(14)的基础上添加

$$X^{(4)}(t) = X_{\omega_i^{(b)}}(t) - A_4 \cdot |C_4 \cdot X_{\omega_i^{(b)}}(t) - X_i(t)| \quad (17)$$

在式(15)中灰狼个体的位置更新公式为 $X^{(i)}(t)$ , $i = 1, 2, 3$ 的平均值。为了强化首领狼的领导作用,则根据适应度值确定权重系数得到灰狼个体的位置更新公式为

$$X_i(t+1) = \frac{1}{3} \left[ 1 - \frac{f(X^{(i)}(t))}{\sum_{j=1,2,3,4} f(X^{(j)}(t))} \right] X^{(i)}(t) \quad (18)$$

为了进一步加强算法的全局寻优能力,采用混沌搜索策略对式(18)的结果进一步搜索,但因为增加全局搜索能力势必会降低算法的收敛速度。为了使算法能够朝着最优解的方向发展,所以引入贪婪策略:在一定概率基础上只接

受比当前适应度更好的结果。所以可以得到基于贪婪策略混沌搜索方法为：

(1) 将  $X_i(t+1)$  映射到  $(0,1)$  区间。假设  $X_{\min}$  与  $X_{\max}$  为解空间的搜索范围, 则可得映射函数为

$$\vartheta(0) = \frac{X_i(t+1) - X_{\min}}{X_{\max} - X_{\min}} \quad (19)$$

(2) 设迭代次数为  $C_{\max}$ , 根据混沌映射进行迭代计算产生一组混沌变量  $\vartheta(i), i = 1, \dots, C_{\max}$ , 并通过逆映射将混沌变量转换至解空间, 得到混沌解序列

$$X_i^{(m)}(t+1) = X_{\min} + (X_{\max} - X_{\min})\vartheta(m) \quad (20)$$

(3) 计算混沌解序列的适应度  $f(X_i^{(m)}(t+1))$ , 得到适应度最优的结果

$$X_i^{(m)}(t+1) = \arg \max \{f(X_i^{(j)}(t+1))\} \quad (21)$$

式中,  $j = 0, 1, \dots, C_{\max}$ 。

(4) 确定贪婪阈值  $\xi_G$ , 得到新的位置更新公式为

$$X_i(t+1) = \begin{cases} X_i^{(C_{\max})}(t+1), & r_3 \geq \xi_G \\ X_i^{(m)}(t+1), & r_3 < \xi_G \end{cases} \quad (22)$$

式中,  $r_3$  为区间  $[0,1]$  中的随机数。

## 2.2 基于CGWO的离线航路规划方法

航路的离线规划问题是指无人机在实际飞行前, 根据任务需求预先在已知的规划空间中寻找一组满足航路评价指标的从起始点到目标点的离散点列。相比于在线规划问题, 航路的离线规划主要有如下特点:

(1) 威胁环境透明。在离线规划过程中, 认为规划范围确定且其中的地形信息明确; 同时假设敌方威胁源的位置分布、类型种类、威胁范围以及威胁等级等因素已知。

(2) 威胁源慢变化。对于运动的威胁目标或任务目标, 通常假设其运动规律已知或是将其运动过程的不确定因素通过概率的形式进行描述。

(3) 计算时间充足, 规划算法所消耗的时间不是首要考虑因素, 即离线规划是通过大量的计算时间与计算量换取高质量的规划结果。

(4) 适用场景充分。基于历史案例以及场景假设, 离线规划会对各种可能的情况进行充分考虑, 使得规划结果具有很强的适应性。从而不仅减少了实际规划过程的不确定性, 还有效地增加了任务的完成概率。

### 2.2.1 基于CGWO的单机航路规划方法

因为计算协同航路  $S_M$  的前提是确定每架无人机  $U_i$  从各自起点到目标点的最优航路  $S_0^{(i)}$ 。采用CGWO算法对单无人机航路进行规划。设灰狼个体  $G_d^{(i)} = (X_d^{(i)}, Y_d^{(i)})^T$  表示第  $i$  个体在第  $d$  维上的位置, 则  $G_d^{(i)}$  即为无人机在二维规划空

间中的位置。

由于本文中单无人机航路规划的目的是确定协同航程  $S_M$ , 所以采用曲线拟合的方式对航路  $G_d^{(i)}, d = 1, \dots, D$  进行处理, 即将点列  $\{G_d^{(i)}\}$  离散化得到  $n$  个点列  $G_s^{(i)}, s = 1, \dots, n$ 。则由式(2)~式(4)可以得到整条航线的威胁代价即为

$$f_i = \frac{1}{nN} \sum_{j=1}^n \sum_{s=1}^n P_{sj}^m, \quad m = \{r, k, z\} \quad (23)$$

整个航线的航程代价可以表示为

$$f_{\text{line}} = \frac{\sum_{j=1}^{n-1} \|G_j^{(i)} G_{j+1}^{(i)}\|}{\|P_s P_t\|} \quad (24)$$

由式(5)可得整条航线的平滑代价值为

$$f_s = \frac{1}{n-1} \sum_{i=1}^{n-1} |\chi_{i+1} - \chi_i| \quad (25)$$

所以基于CGWO算法的单机航路规划代价函数可以表示为

$$f = w_1 f_i + w_2 f_{\text{line}} + w_3 f_s \quad (26)$$

式中,  $w_i$  为代价的权重, 且有  $w_1 + w_2 + w_3 = 1$ 。

### 2.2.2 基于几何规划的航路扩展方法

执行多方向饱和攻击任务的关键是多无人机能够以预设的攻击方向进入目标区域, 且从起始点到目标点间的航程相等。其中攻击方向是指无人机经过最后一个航点转向后的飞行方向, 对于给定攻击目标的进入角的情况, 如果按照从起始点开始搜索的方法, 则预定的攻击方向为一个全局限制条件, 明显增加了问题的复杂度。本节采用逆向航路节点扩展策略: 从目标点出发以预设进入角方向进行逆向搜索, 根据机体约束条件, 寻找一条从目标点到起始点的满足任务需求的飞行航路。假设多方向饱和攻击任务的协同航程  $S_M$  已经给定, 则可根据椭圆第一定义建立基于几何规划的逆向节点搜索策略。

对于无人机  $U_i$ , 将起始点  $P_s^{(i)}$  与目标点  $P_t^{(i)}$  作为焦点, 协同航程  $S_M$  为长轴距离, 则可构造如图5所示的椭圆规划区域  $O_0^{(i)}$ 。通过式(26)所示的坐标转换公式可以得到以椭圆中心  $O_0^{(i)}$  为原点,  $P_s^{(i)}$  为左焦点,  $\overrightarrow{P_s^{(i)} P_t^{(i)}}$  为  $x$  轴的新坐标系。

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (27)$$

式中,  $\theta_0^{(i)} = \arctan(P_t^{(i)}/P_s^{(i)})$ ; 在新坐标系下, 椭圆  $O_0^{(i)}$  上任意点  $p(x', y')$  的极坐标公式可以定义为

$$\begin{aligned} \rho &= \frac{b^2}{a - a \cos \alpha'} \\ x' &= \rho \cos \alpha' - c \\ y' &= \rho \sin \alpha' \end{aligned} \quad (28)$$

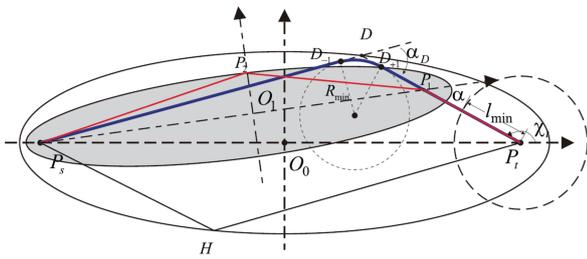


图5 基于几何规划的逆向节点搜索策略示意图

Fig.5 Reverse node search strategy based on Geometric Programming

式中:  $a, b, c$  分别为椭圆的长半轴、短半轴与半焦距;  $\alpha' \in (-\pi, \pi]$  表示  $\overrightarrow{P_s^{(i)} p}$  与  $\overrightarrow{P_s^{(i)} P_t^{(i)}}$  之间的夹角。

对于给定攻击目标的进入角  $\mu_1^{(i)}$ , 根据坐标转换与椭圆的对称性, 将  $\alpha' = \mu_1^{(i)} - \theta_0^{(i)}$  代入式(28)则可得到满足目标进入角约束的椭圆上点  $D_0^{(i)}(-x', y')$ , 同时此点满足协同航程要求, 即

$$\|P_s^{(i)} D_0^{(i)}\| + \|D_0^{(i)} P_t^{(i)}\| = S_M \quad (29)$$

则  $p_1^{(i)} = D_0^{(i)}$  即为满足多方向饱和攻击要求的扩展节点。由于本文的航点切换方式采用内切飞行策略, 则由式(1)得因为转弯而引起的航程误差为

$$\Delta l_1^{(i)} = R_{\min} \cdot \tan\left(\frac{\chi_1^{(i)}}{2}\right) - R_{\min} \cdot \frac{\chi_1^{(i)}}{2} \quad (30)$$

式中,  $\chi_1^{(i)}$  为点  $p_1^{(i)}$  处的转弯角。由于存在目标进入角的限制, 所以  $p_1^{(i)}$  只能在线段  $D_0^{(i)} P_t^{(i)}$  内移动, 则点  $p_1^{(i)}$  满足如下条件

$$p_1^{(i)} = (1 - \lambda_0^{(i)}) \cdot P_t^{(i)} + \lambda_0^{(i)} \cdot D_0^{(i)} \quad (31)$$

式中,  $\lambda_0^{(i)}$  为区间  $[0, 1]$  上的随机数。另外由于目标点存在最小最准距离  $l_{\min}^{(i)}$  的限制, 则有  $\lambda_0^{(i)} \in [\lambda_{\min}^{(i)}, 1]$ , 其中  $\lambda_{\min}^{(i)} = l_{\min}^{(i)} / \|D_0^{(i)} P_t^{(i)}\|$ 。当  $\lambda_0^{(i)} \rightarrow 1$  时有  $p_1^{(i)} \rightarrow D_0^{(i)}$ , 且随着  $\chi_1^{(i)}$  变大, 航程误差  $\Delta l_1^{(i)}$  也变大。则可以定义  $p_1^{(i)}$  处的剩余航路长度为

$$\Delta S^{(i)} = S_M - \|p_1^{(i)} P_t^{(i)}\| + \Delta l_1^{(i)} \quad (32)$$

当  $\Delta S^{(i)} > \|P_s^{(i)} p_1^{(i)}\|$  时, 表示不能直接从点  $P_s^{(i)}$  到  $p_1^{(i)}$ , 需要扩展线段航点  $p_2^{(i)}$ 。则以剩余航路长度  $\Delta S^{(i)}$  作为椭圆长轴,  $\|P_s^{(i)} p_1^{(i)}\|$  为焦距构造辅助椭圆  $O_1^{(i)}$  与相应的新坐标系。选取随机搜索方向  $\chi_2^{(i)}$ , 则根据式(26)与式(27)可得椭圆上的点  $D_1^{(i)}$ 。选取随机扩展比例  $\lambda_1^{(i)} \in [0, 1]$ , 由式(30)可以得到新的扩展节点  $p_2^{(i)}$ 。设每个航点  $p_j^{(i)}, j = 1, 2, \dots$  处的转弯角为  $\chi_j^{(i)}$ , 根据式(32)得到的剩余航路长度判断为

$$\Delta S^{(i)} = S_M - \sum_{j=0}^m |p_{j-1}^{(i)} p_j^{(i)}| + \sum_{j=1}^m \Delta l_j^{(i)} \quad (33)$$

根据剩余航路长度  $\Delta S^{(i)}$  判断是否需要扩展新的航点。因为随着航点的不断扩展, 构造的辅助椭圆的长轴与焦距之间的差距逐渐减少, 即椭圆  $O_j^{(i)}$  的离心率  $e_j^{(i)} = c_j^{(i)} / a_j^{(i)}$  逐渐增大。当离心率  $e_j^{(i)}$  超过阈值  $\xi_e$  时, 不需要从椭圆内部扩展新的航点, 即在  $(0, b)$  或  $(0, -b)$  中选择新的扩展航点。离心率阈值  $\xi_e$  的取值由剩余航路长度确定。当  $\Delta S^{(i)} \leq (\Delta T_s + \Delta T_E) v_i$  时, 即可得到满足协同航程  $S_M$  要求的  $m$  个航点序列  $P_t^{(i)}, p_1^{(i)}, \dots, p_m^{(i)}, P_s^{(i)}$ , 然后将此序列进行逆序排列即可得到满足多方向饱和攻击任务要求的航路  $\text{Line}_i = \{P_0^{(i)}, P_1^{(i)}, \dots, P_{m+1}^{(i)}\}$ 。

另外, 因为椭圆上任意一点到两个焦点的距离之和为椭圆的长轴距, 所以可以根据此性质实现局部备选航线的生成。假设三个连续航点为  $P_{j-1}^{(i)}, P_j^{(i)}$  与  $P_{j+1}^{(i)}$ , 可以构造辅助椭圆: 焦距为  $2c = \|P_{j-1}^{(i)} P_{j+1}^{(i)}\|$ , 长轴为  $2a = \|P_{j-1}^{(i)} P_j^{(i)}\| + \|P_j^{(i)} P_{j+1}^{(i)}\|$ , 则可选取此椭圆上任意一点代替  $P_j^{(i)}$  得到备选航点。

### 2.2.3 多机协同航路规划策略

在基于几何规划的航点扩展过程中, 已将任务要求的协同航程与目标进入角作为规划条件进行考虑, 但还需要考虑无人机间的防撞。多机间的防撞航路规划策略为:

首先规划出某架无人机的代价最优的航路  $\text{Line}_A$ , 然后将此规划结果作为约束条件对另一架无人机的航路进行规划, 得到与  $\text{Line}_A$  不相撞的航线  $\text{Line}_B$ , 最后以此类推得到满足防撞要求的  $n$  条协同航路。

式(10)给出了无人机航路间的安全间隔限制, 但航路间的间隔过远会影响整体航路的协同性能, 无人机航路应满足协同间隔

$$\|p_i(t) - p_j(t)\| \leq d_c \quad (34)$$

式中,  $\forall i, j \in \{1, \dots, n\}$ 。

如图6所示, 通过引入双曲正切函数可以将无人机间的间隔限制约束条件映射到  $[0, 1]$  间<sup>[34]</sup>, 令  $\Delta d$  表示缓存距离且有  $0 < \Delta d \ll d_s$ , 则有

$$\begin{cases} f^{\min_c} = \sum_{i \neq j} \frac{1}{2} \left[ 1 - \tanh\left(\frac{\|p_i(t) - p_j(t)\| - d_s - \Delta d/2}{\Delta d/8}\right) \right] \\ f^{\max_c} = \sum_{i \neq j} \frac{1}{2} \left[ 1 + \tanh\left(\frac{\|p_i(t) - p_j(t)\| - d_c + \Delta d/2}{\Delta d/8}\right) \right] \end{cases} \quad (35)$$

所以由式(23)、式(25)与式(35)可得多无人机协同航路的评价函数为

$$f = w_a \sum_{i=1}^n (f_1^{(i)} + f_s^{(i)}) + w_b (f^{\min_c} + f^{\max_c}) \quad (36)$$

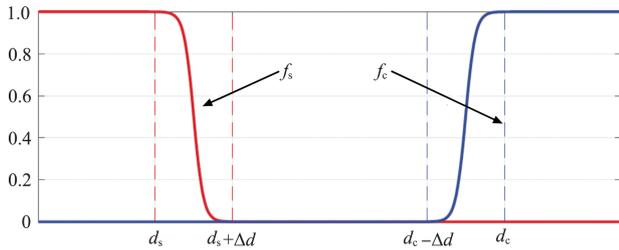


图6 双曲正切函数映射示意图

Fig.6 Hyperbolic tangent function mapping

式中： $w_a + w_b = 1$ 为代价权重。

区别于单机航路规划,采用CGWO算法对多机协同航路规划问题求解的变量为辅助椭圆的随机搜索方向 $\chi_j^{(i)}$ 与扩展比例 $\lambda_j^{(i)}$ ,其中 $i = \{1, \dots, n\}$ 表示无人机序号, $j \in [1, m]$ 为规划的航点序号。基于几何规划与CGWO算法的多无人机协同航路规划流程如图7所示。

### 3 仿真分析与验证

#### 3.1 混沌算子选取

利用混沌映射的遍历性以及其对初值的敏感性,本文所提出的混沌灰狼优化(CGWO)算法通过引入混沌映射以提高原算法的全局寻优能力。典型的混沌映射总共有12

种,见表1,而不同的混沌映射将对原算法性能具有不同的改进程度。如图8所示,当初始值为 $x_0=0.58$ ,最大迭代次数为 $t_{max} = 100$ 时,不同的混沌映射具有不同变化规律。根据结果可得,由于混沌映射的非周期性,使得映射结果具有较大的离散性。但不同的映射具有不同的离散程度。所以为了实现对原算法更好的改进,本节在同一仿真环境下采用不同的混沌映射,通过统计结果确定CGWO算法中所使用的混沌映射。

假设CGWO算法的种群数量 $N_p = 30$ ,最大迭代次数 $t_{max} = 200$ ,维度 $D = 5$ 。对于同一场景的单无人机航路规划问题,算法初始化以及变量 $a$ 分别采用不同的混沌映射进行200次运算,表2给出了规划结果的最大值、最小值以及平均值如图9与图10分别给出了统计结果的盒图。

混沌初始化映射选取:根据统计结果可得,当CGWO算法采用Piecewise映射( $M_9$ )、Chebyshev映射( $M_{10}$ )以及Intermittency映射( $M_{12}$ )时得到的航路规划结果最优。将三种映射做进一步比较可得,CGWO算法采用Intermittency映射能够得到比较满意的结果,且实现简单。

混沌变量 $a$ 选取:由表3可得,采用Intermittency映射( $M_{12}$ )、Sinusoidal映射( $M_6$ )与Gauss映射( $M_3$ )时能够得到较优的规划结果。如图10所示, $M_6$ 存在较大的离散性,另外,

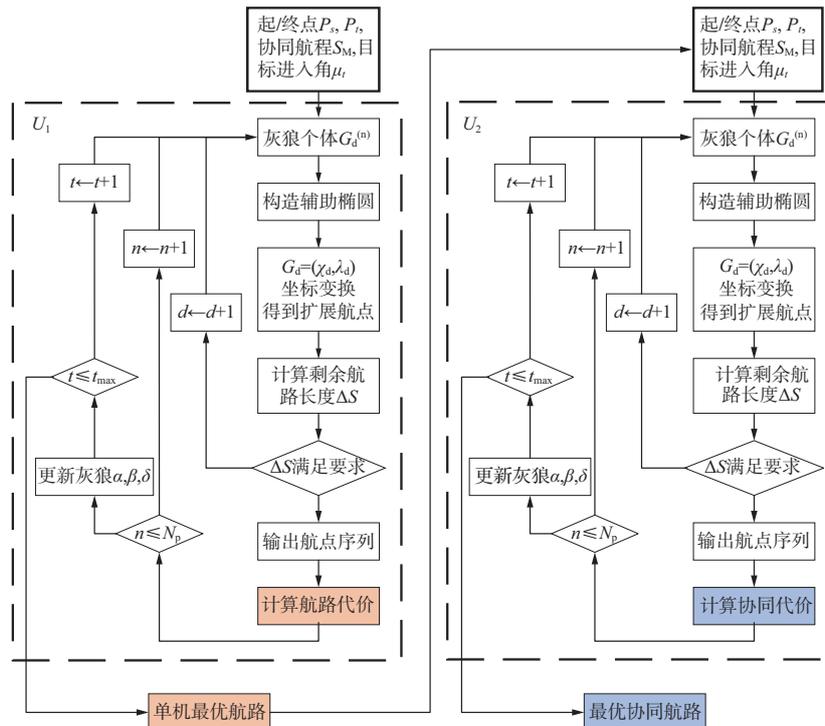


图7 多无人机协同航路规划流程

Fig.7 Flow chart of multi UAV cooperative path planning

表1 典型的混沌映射  
Table 1 Typical chaotic mapping

映射	性质
M <sub>1</sub> Logistic	$x_{k+1} = \mu x_k(1 - x_k)$ $[0, 1], \begin{cases} x_0 \notin \{0, 0.25, 0.5, 0.75, 1\} \\ \mu = 4, 1 \leq \mu \leq 4 \end{cases}$
M <sub>2</sub> Circle	$x_{k+1} = x_k + b - \frac{a}{2\pi} \sin 2\pi x_k \pmod 1$ $[0, 1], a = 0.5, b = 0.2$
M <sub>3</sub> Gauss/Mouse	$x_{k+1} = \begin{cases} 0 & x_k = 0 \\ \frac{1}{x_k} \pmod 1 & x_k \neq 0 \end{cases}$ $[0, 1], \frac{1}{x_k} \pmod 1 = \frac{1}{x_k} - \left\lfloor \frac{1}{x_k} \right\rfloor$
M <sub>4</sub> Iterative	$\sin(a\pi/x_k)$ $[-1, 1], a = 0.85, a \in (0, 1)$
M <sub>5</sub> Tent	$x_{k+1} = \begin{cases} x_k/0.7 & x_k < 0.7 \\ -\frac{10}{3}(1 - x_k) & x_k \geq 0.7 \end{cases}$ $[0, 1]$
M <sub>6</sub> Sinusoidal	$x_{k+1} = ax_k^2 \sin(\pi x_k)$ $[0, 1], \begin{cases} a = 2.3, x_0 = 0.7 \text{ 时} \\ x_{k+1} = \sin(\pi x_k) \end{cases}$
M <sub>7</sub> Singer	$x_{k+1} = \mu \left( \frac{7.86x_k - 23.31x_k^2 + 28.75x_k^3 - 13.302875x_k^4}{1} \right)$ $[0, 1], \mu = 1.07, \mu \in [0.9, 1.08]$
M <sub>8</sub> Sine	$x_{k+1} = \frac{a}{4} \sin(\pi x_k)$ $[0, 1], a = 4, a \in (0, 4]$
M <sub>9</sub> Piecewise	$x_{k+1} = \begin{cases} x_k/d & 0 \leq x_k < d \\ \frac{x_k - d}{0.5 - d} & d \leq x_k < 0.5 \\ \frac{1 - d - x_k}{0.5 - d} & 0.5 \leq x_k < 1 - d \\ (1 - x_k)/d & 1 - d \leq x_k < 1 \end{cases}$ $(0, 1), d = 0.3, d \in (0, 0.5)$
M <sub>10</sub> Chebyshev	$x_{k+1} = \cos(\kappa \arccos(x_k))$ $[-1, 1], \kappa = 4$
M <sub>11</sub> Liebovitch	$x_{k+1} = \begin{cases} \alpha_1 x_k & 0 < x_k \leq d_1 \\ \frac{d_2 - x_k}{d_2 - d_1} & d_1 < x_k \leq d_2 \\ 1 - \alpha_2(1 - x_k) & d_2 < x_k < 1 \end{cases}$ $d_1 = 0.3, d_2 = 0.8,$ $d_1, d_2 \in (0, 1), d_1 < d_2$ $[0, 1], \alpha_1 = \frac{d_2(1 - (d_2 - d_1))}{d_1}$ $\alpha_2 = \frac{((d_2 - 1) - d_1(d_2 - d_1))}{d_2 - 1}$
M <sub>12</sub> Intermittency	$x_{k+1} = \begin{cases} \varepsilon + x_k + cx_k^m & 0 < x_k \leq d \\ \frac{x_k - d}{1 - d} & d < x_k < 1 \end{cases}$ $\varepsilon = 0.1, d = 0.5, m = 2$ $(0, 1), c = \frac{1 - \varepsilon - d}{d^m}$

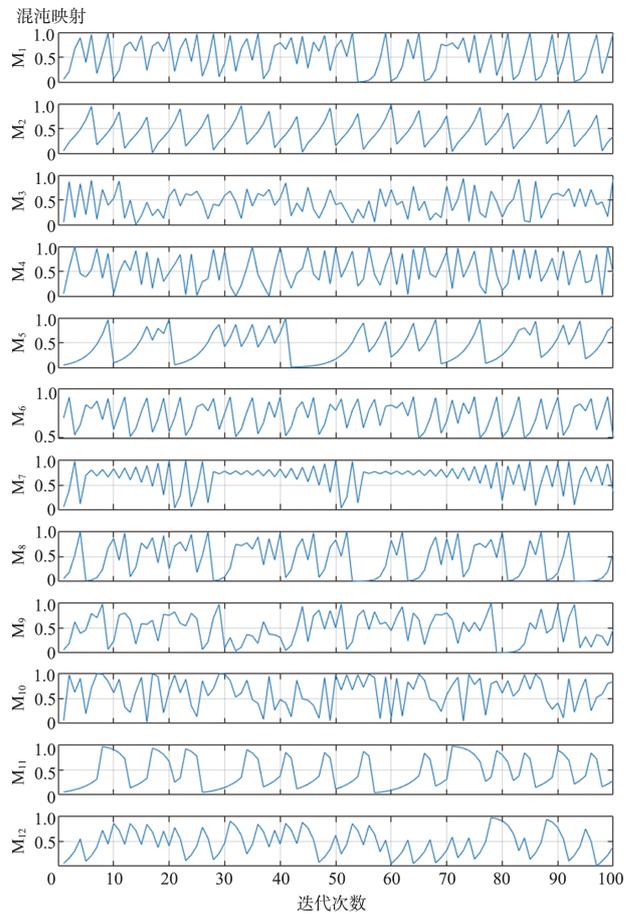


图8 典型的混沌映射示意图

Fig.8 Schematic diagram of typical chaotic maps

表2 不同混沌映射对航路规划结果的影响

Table 2 Influence of different chaotic maps on path planning results

	混沌初始化		
	Max	Min	Mean
M <sub>1</sub>	1.5014	1.1120	1.1987
M <sub>2</sub>	1.5004	1.1174	1.1991
M <sub>3</sub>	1.6742	1.1186	1.1886
M <sub>4</sub>	1.3616	1.1201	1.1893
M <sub>5</sub>	1.6911	1.1179	1.1910
M <sub>6</sub>	3.2147	1.1208	1.5462
M <sub>7</sub>	1.5855	1.1069	1.2152
M <sub>8</sub>	1.6621	1.1270	1.2005
M <sub>9</sub>	1.3571	1.1094	1.1911
M <sub>10</sub>	1.5052	1.1064	1.2134
M <sub>11</sub>	1.4705	1.1122	1.1989
M <sub>12</sub>	1.6756	1.1095	1.1861

因为M<sub>12</sub>对混沌初始化具有较为明显的优化结果,所以混沌变量a选择M<sub>12</sub>作为混沌算子。同时为了便于计算,式(21)同样采用M<sub>12</sub>作为混沌算子。

表3 不同混沌变量 $\alpha$ 对航路规划结果的影响  
Table 3 Influence of different chaotic variables  $\alpha$  on path planning results

	混沌变量 $\alpha$		
	Max	Min	Mean
$M_1$	1.3666	1.1590	1.2280
$M_2$	1.5707	1.1663	1.2194
$M_3$	1.5323	1.1682	1.2141
$M_4$	1.3042	1.1802	1.2313
$M_5$	1.4373	1.1806	1.2264
$M_6$	1.8493	1.1557	1.2713
$M_7$	1.3511	1.1813	1.2568
$M_8$	1.3827	1.1606	1.2261
$M_9$	1.2944	1.1676	1.2272
$M_{10}$	1.3439	1.1602	1.2569
$M_{11}$	1.4931	1.1650	1.2186
$M_{12}$	1.2851	1.1718	1.2226

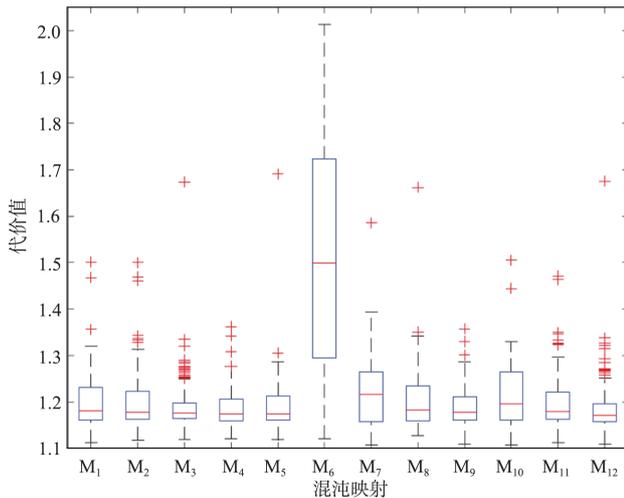


图9 混沌初始化规划结果统计盒图

Fig.9 Statistical box diagram of chaos initialization

### 3.2 算法的比较以及有效性验证

为了验证本节所提出的算法的有效性,采用粒子群算法、灰狼优化算法与混沌灰狼优化对单无人机航路规划问题进行求解,可以得到如图11所示航路规划结果。其中种群数量 $N_p = 50$ ,最大迭代次数 $t_{max} = 150$ ,维度 $D = 5$ ,规划空间内威胁分布的情况见表4。本文在设计代价函数时进行了无量纲(量纲一)处理,所以仿真分析不考虑实际单位。

将式(25)中航路代价的权重系数设为 $w_i = 1/3, (i = 1,2,3)$ ,则航路代价随迭代次数的变化情况如图12所示。通过对比可得,CGWO算法的初始代价高于其他两种算法,为了增加在解空间中的遍历性,CGWO算法选取的初始解

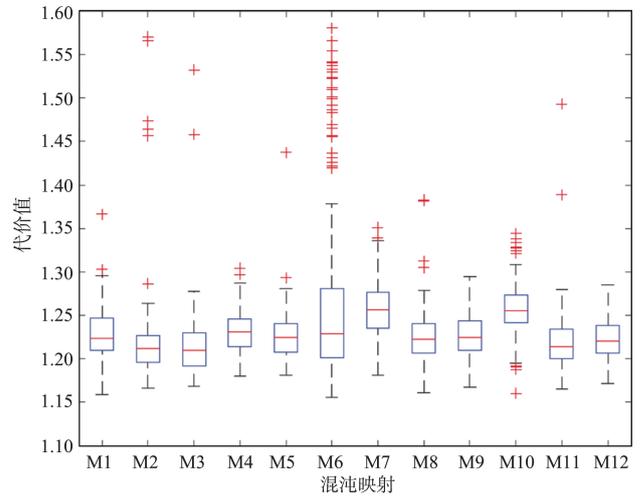


图10 混沌算子 $a$ 规划结果统计盒图

Fig.10 Statistical box diagram of chaotic operator  $a$

表4 规划空间中威胁的分别情况  
Table 4 Threats in the planning space

	类型及位置	参数
1	雷达 (334.2,698.8)	$R=173.7, K=0.77, w=0.76$
2	禁飞 (30.67,744.1)	$R=69.7$
3	禁飞 (480,904.8)	$R=115.1$
4	高炮(576.8,183)	$R^{max}=160.9, R^{min}=107.8, w=0.9$
5	高炮(886.6,28.7)	$R^{max}=76, R^{min}=56.24, w=0.82$
6	雷达(168,978.7)	$R=113.5, K=0.81, w=0.63$
7	高炮(500.5,471.1)	$R^{max}=147, R^{min}=108.78, w=0.99$
8	禁飞(682,42.5)	$R=49$
9	高炮(521.7,96.8)	$R^{max}=50.8, R^{min}=37.08, w=0.97$
10	高炮(659.7,518.6)	$R^{max}=62.6, R^{min}=43.75, w=0.97$

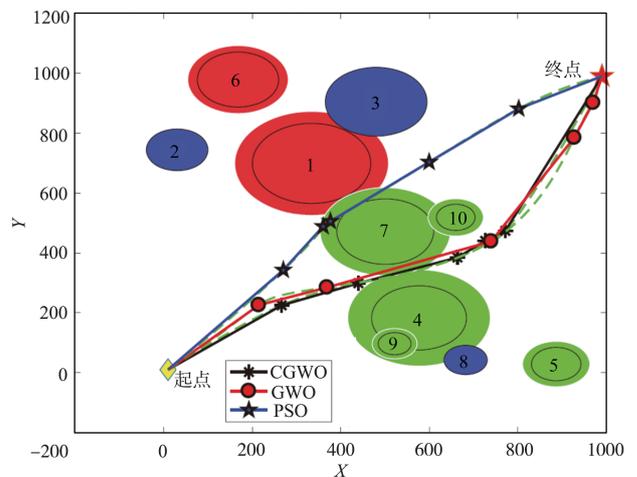


图11 单机航路规划结果

Fig.11 Path planning results of single UAV

为 $2 \times N_p$ 组混沌初始解按序排列后的奇数项,所以存在随机初始解优于混沌初始解的情况。但是随着寻优过程的进

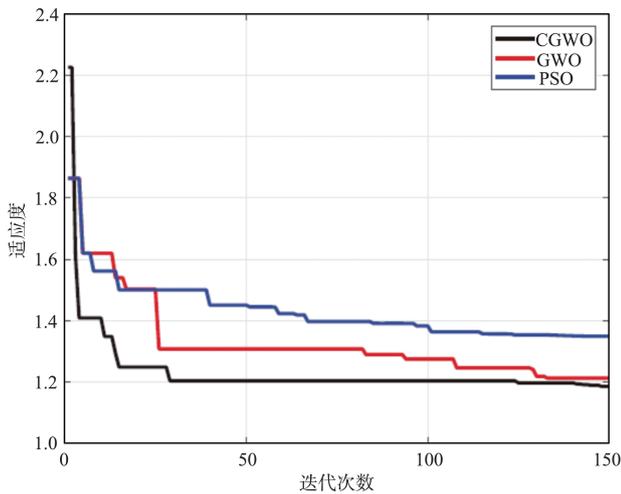


图12 规划算法适应度变化情况

Fig.12 Comparison between fitness of different planning algorithms

行,CGWO算法很快找到了最优解。航路规划的结果为  $1.18(\text{CGWO}) > 1.21(\text{GWO}) > 1.35(\text{PSO})$ 。

通过将式(25)中航路的代价权重进行调整可以得到不同规划目标下的规划结果。如图13所示,定义  $w_i (i = 1,2,3)$  的取值分别为  $(0.8,0.1,0.1)$ 、 $(0.1,0.8,0.1)$  与  $(0.1,0.1,0.8)$  时采用CGWO算法可以分别得到航程最优、安全最优以及平滑最优的规划结果。其中种群数量  $N_p = 30$ ,最大迭代次数  $t_{\max} = 150$ ,维度  $D=11$ 。

为了进一步验证本节所提的CGWO算法的性能,采用随机生成威胁环境的方式进行仿真试验。在  $1000 \times 1000$  的规划空间内设置10个威胁源,威胁源的位置、威胁半径与威胁类型随机生成。CGWO算法、GWO算法与PSO算

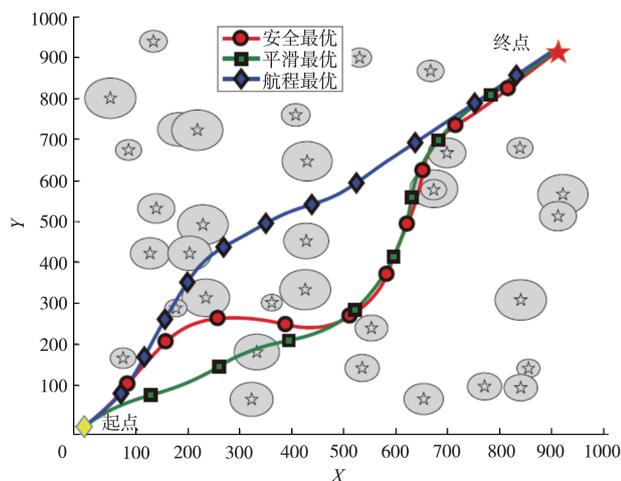


图13 代价权重不同时的规划结果

Fig.13 Planning results with different cost weights

法的种群数量为30,维度为  $D=5$ ,最大迭代次数80,图14与图15给出了600次的仿真统计图,结果证明了CGWO算法在解决单机航路规划问题上具有明显优势。

在每次仿真过程中,三种算法都对同一个威胁环境进行仿真试验,而在不同的仿真中,由于规划环境不同,所以不能对仿真结果直接进行对比。假设算法的最优代价值分别为  $f_{\text{CGWO}}$ 、 $f_{\text{GWO}}$  与  $f_{\text{PSO}}$ ,则采用式(36)所示的归一化操作,即可得到图14所示的统计盒图。CGWO算法的仿真结果明显优于其他两种算法,能够得到代价值更优的规划结果。

令  $\Delta \bar{f}_{\text{GWO}} = \bar{f}_{\text{CGWO}} - \bar{f}_{\text{GWO}}$ ,  $\Delta \bar{f}_{\text{PSO}} = \bar{f}_{\text{CGWO}} - \bar{f}_{\text{PSO}}$ ,则得到如图15所示的结果示意图。在600次仿真中,可以得到CGWO算法优于GWO的情况为74%,优于PSO的情况为54.8%。

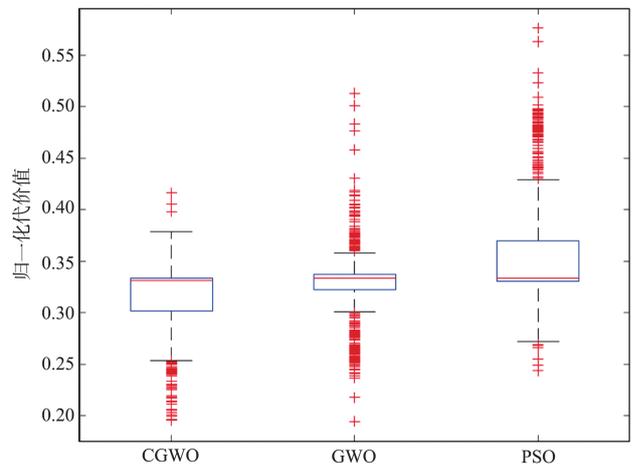


图14 单机航路规划结果统计盒图

Fig.14 Statistical box diagram of path planning

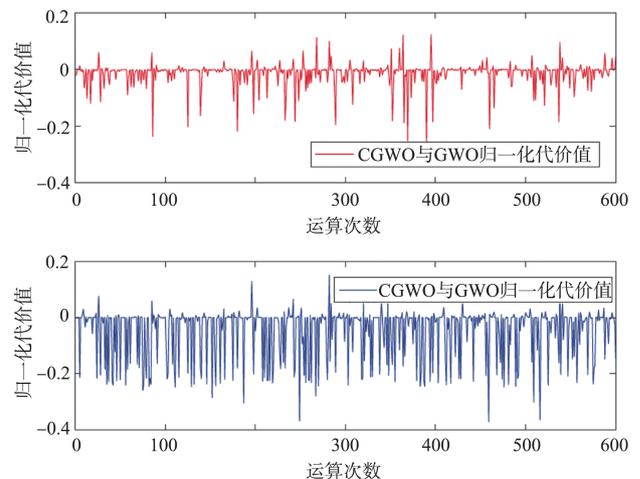


图15 单机航路规划结果最优代价值差

Fig.15 Optimal surrogate price difference of single UAV path planning results

$\Delta \bar{f}_{GWO}$  与  $\Delta \bar{f}_{PSO}$  的平均值分别为  $-0.0129$  与  $-0.0466$ , 相应的方差值分别为  $0.0017$ 、 $0.0080$ , 即 CGWO 算法的平均水平优于 GWO 与 PSO 算法。另外根据仿真结果可得 GWO 算法的寻优能力强于 PSO 算法。

### 3.3 多机协同规划结果分析

假设三架无人机  $U^{(i)} (i = 1, 2, 3)$  分别从  $(10, 350)$ 、 $(0, 0)$  以及  $(360, 30)$  出发, 目标位置为  $(990, 990)$ 。假设目标的进入角分别为  $\mu_1 = 150^\circ$ ,  $\mu_2 = 270^\circ$ ,  $\mu_3 = 390^\circ$ , 最小对准距离  $l_{min}^{(i)} = 100$ 。根据图 7 所示的多无人机协同航路规划流程, 采用 CGWO 算法对航路规划问题进行求解, 得到的规划结果如图 16 所示。

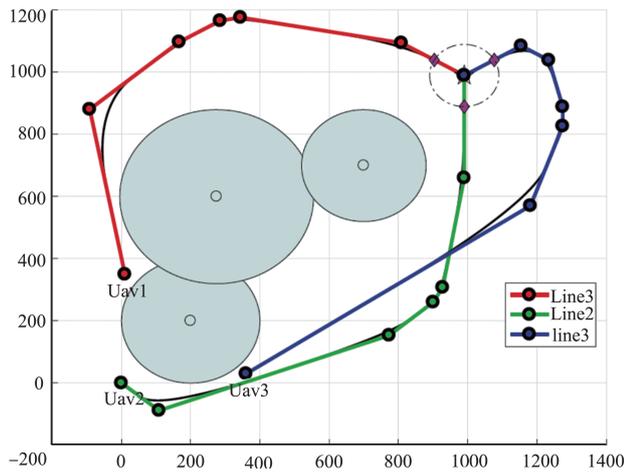


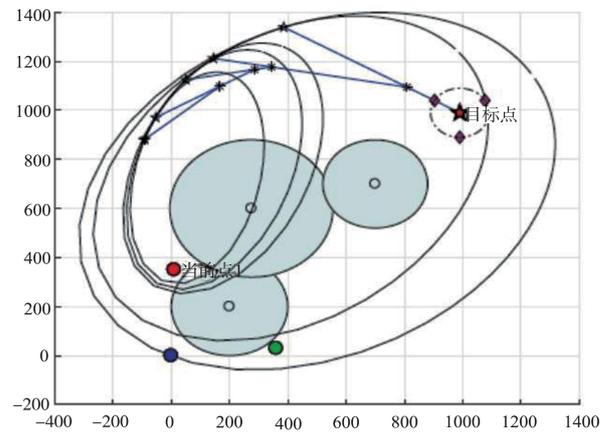
图 16 多无人机航路规划结果

Fig.16 Multi UAVs path planning results

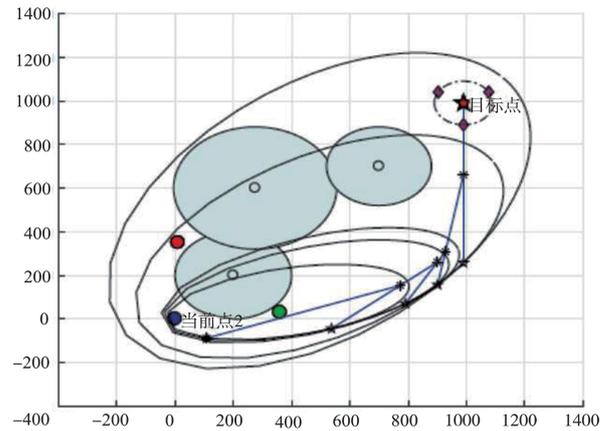
假设各无人机以相同的速度匀速飞行, 则可以得到协同航程为  $S_M = 1755.7$ 。各无人机基于几何规划策略构造的航路规划结果如图 17 所示。图中五角星表示根据方位角  $\chi_j^{(i)}$  确定的辅助椭圆  $O_j^{(i)}$  上的点, 辅助椭圆根据剩余航程  $\Delta S^{(i)}$  进行构造。根据式(30)将参数  $(\lambda_j^{(i)}, \chi_j^{(i)})$  解码为参考的航路点, 即如图中的符号\*所示。

## 4 结论

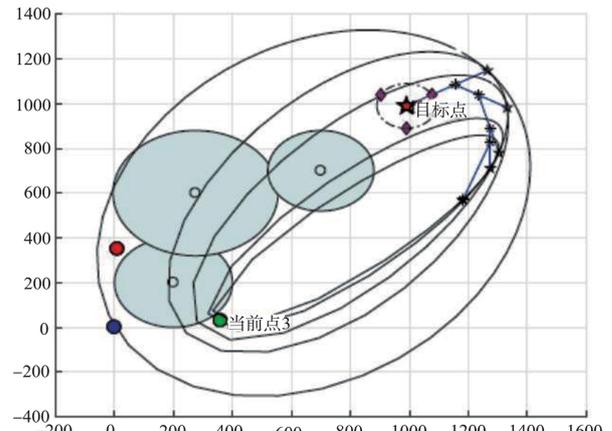
本文针对多无人机协同执行饱和和攻击任务, 根据椭圆第一定义提出了基于几何规划的从目标点出发的逆向航点扩展策略, 从而构造出满足任务攻击时间与攻击方位要求的航点序列。将混沌映射引入灰狼优化算法中, 提出了基于混沌灰狼优化算法的多无人机协同航路规划方法。仿真结果表明, 提出的 CGWO 算法能够根据任务需求生成满足方位与时间约束的协同航路, 同时算法的收敛速度与寻优精度明显优于其他算法。



(a)



(b)



(c)

图 17 基于几何规划的航路构造结果

Fig.17 Path construction results based on geometric planning

由于本文假设无人机为匀速运动, 以便将“按时到达”问题转换为路径长度对话。但在实际工程应用中并不一定要全部无人机按相同的速度到达, 所以本文接下来的工作针对不同到达速度要求下的“同时到达”问题展开研究, 并扩展到三维航路规划, 为实验验证提供必要的研究基础。 **AST**

## 参考文献

- [1] Tang Yuan, Zhou Rui, Sun Guibin, et al. A novel cooperative path planning for multi-robot persistent coverage in complex environments[J]. IEEE Sensors Journal, 2020, 20(8): 4485-4495.
- [2] 吴兆香, 欧阳权, 王志胜, 等. 基于人工智能的无人机区域侦察方法研究现状与发展[J]. 航空科学技术, 2020, 31(10): 57-68.
- Wu Zhaoxiang, Ouyang Quan, Wang Zhisheng, et al. Status and development of regional reconnaissance methods of UAV based on artificial intelligence[J]. Aeronautical Science & Technology, 2020, 31(10): 57-68. (in Chinese)
- [3] 王村松. 多无人机编队在线协同航路规划方法研究[D]. 南昌: 南昌航空大学, 2016.
- Wang Cunsong. Research on on-line cooperative path planning for multi-UAV formation[D]. Nanchang: Nanchang Hangkong University, 2016. (in Chinese)
- [4] Lyu Desheng, Chen Ziwei, Cai Zesu, et al. Robot path planning by leveraging the graph-encoded Floyd algorithm[J]. Future Generation Computer Systems, 2021, 122: 204-208.
- [5] 池恫, 余磊. 民用飞机航路智能规划技术研究[J]. 航空科学技术, 2020, 31(10): 46-50.
- Chi Yi, Yu Lei. Research on civil flight intelligent trajectory planning technology[J]. Aeronautical Science & Technology, 2020, 31(10): 46-50. (in Chinese)
- [6] Zhang Jing, Wu Jun, Shen Xiao, et al. Autonomous land vehicle path planning algorithm based on improved heuristic function of A-Star[J]. International Journal of Advanced Robotic Systems, 2021, 18(5): 17298814211042730.
- [7] Xu Tianying, Zhou Haibo, Tan Shuaixia, et al. Mechanical arm obstacle avoidance path planning based on improved artificial potential field method[J]. Industrial Robot, 2022, 49(2): 271-279.
- [8] Huang Shengkai, Wang Wenjune, Sun Chungsun. A path planning strategy for multi-robot moving with path-priority order based on a generalized voronoi diagram[J]. Applied Sciences, 2021, 11(20): 9650.
- [9] 徐琳, 宋万强, 严冲, 等. 基于威胁评估和扩展 Voronoi 图的战术飞行轨迹规划方法[J]. 航空科学技术, 2019, 30(1): 35-43.
- Xu Lin, Song Wanqiang, Yan Chong, et al. Tactical flight trajectory planning methods based on threat assessments and improved Voronoi graphics[J]. Aeronautical Science & Technology, 2019, 30(1): 35-43. (in Chinese)
- [10] Tian Shasha, Li Yuanxiang, Kang Yilin, et al. Multi-robot path planning in wireless sensor networks based on jump mechanism PSO and safety gap obstacle avoidance[J]. Future Generation Computer Systems, 2021, 118: 37-47.
- [11] Ling Fuyuan, Chen Jinchao, Du Chenglie. Multi-obstacle path planning of UAV based on improved ant colony system algorithm[C]//2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC). IEEE, 2020: 1731-1735.
- [12] Wang Wei, Zuo Lei, Xu Xin. A learning-based multi-RRT approach for robot path planning in narrow passages[J]. Journal of Intelligent & Robotic Systems, 2018, 90(1): 81-100.
- [13] Duan Haibin, Huang Linzhi. Imperialist competitive algorithm optimized artificial neural networks for UCAV global path planning[J]. Neurocomputing, 2014, 125: 166-171.
- [14] Wang Lei, Luo Chaomin. A hybrid genetic Tabu search algorithm for mobile robot to solve AS/RS path planning[J]. International Journal of Robotics and Automation, 2018, 33(2): 161-168.
- [15] He Lei, Nabil A, Song Bifeng. Explainable deep reinforcement learning for UAV autonomous path planning[J]. Aerospace Science and Technology, 2021, 118: 107052.
- [16] Cekmez U, Ozsiginan M, Sahingoz O K. Multi-UAV path planning with parallel genetic algorithms on CUDA architecture [C]//Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion, 2016: 1079-1086.
- [17] Shanmugavel M, Tsourdos A, White B, et al. Co-operative path planning of multiple UAVs using Dubins paths with clothoid arcs[J]. Control Engineering Practice, 2010, 18(9): 1084-1092.
- [18] Sun Xiaolei, Liu Yanfang, Yao Weiran, et al. Triple-stage path prediction algorithm for real-time mission planning of multi-UAV[J]. Electronics Letters, 2015, 51(19): 1490-1492.
- [19] 吴坤, 谭劭昌. 基于改进鲸鱼优化算法的无人机航路规划[J]. 航空学报, 2020, 41(S2): 724286.
- Wu Kun, Tan Shaochang. Path planning of UAVs based on

- improved whale optimization algorithm[J]. *Acta Aeronautica et Astronautica Sinica*, 2020, 41(S2): 724286. (in Chinese)
- [20] Wu Kun, Cai Zhihao, Zhao Jiang, et al. Target Tracking based on a nonsingular fast terminal sliding mode guidance law by fixed-wing UAV[J]. *Applied Sciences*, 2017, 7(4):333.
- [21] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimize[J]. *Advances in Engineering Software*, 2014, 69:46-61.
- [22] Cai Zhihao, Zhou Hui, Zhao Jiang, et al. Formation control of multiple unmanned aerial vehicles by event-triggered distributed model predictive control[J]. *IEEE Access*, 2018, 6: 55614-55627.
- [23] Wang Gaige, Guo Lihong, Gandomi A, et al. Chaotic krill herd algorithm[J]. *Information Sciences*, 2014, 274:17-34.

## Cooperative Path Planning for Multi-UAVs Based on Chaos Gray Wolf Optimization

Wu Kun<sup>1</sup>, Chi Pei<sup>1,2</sup>, Wang Yingxun<sup>1,2</sup>, Hou Lin<sup>3</sup>

1. *Beihang University, Beijing 100191, China*

2. *National Key Laboratory of Science and Technology on Aircraft Control, Beihang University, Beijing 100191, China*

3. *National Key Laboratory of Science and Technology on Aircraft Control, AVIC Xi'an Flight Automatic Control Research Institute, Xi'an 710065, China*

**Abstract:** Aiming at the space-time constraints of multi UAVs cooperative execution of saturation attack mission, an off-line path planning method based on chaotic gray wolf optimization(CGWO) is proposed to realize the effective generation of multi UAVs cooperative path. Firstly, according to the characteristics of saturation attack mission, the simultaneous arrival problem of multiple UAVs with space-time constraints is transformed into a cooperative range problem; Secondly, aiming at introducing chaotic mapping into gray wolf optimization algorithm, a chaotic gray wolf optimization algorithm is proposed to improve the exploration ability and convergence speed of the original algorithm; Thirdly, a waypoint expansion strategy based on geometric programming is proposed to construct a waypoint sequence that meets the requirements of mission attack time and attack azimuth; Through the simulation of single UAV path planning problem, the optimization ability of the improved algorithm is verified. The feasibility and effectiveness of the proposed method are verified by path planning simulation for saturation attack mission.

**Key Words:** multi UAVs; saturation attack mission; cooperative path planning; chaos gray wolf optimization; geometric programming

**Received:** 2022-02-16; **Revised:** 2022-04-15; **Accepted:** 2022-05-11

**Foundation item:** Aeronautical Science Foundation of China(20185851021)