

基于改进 RRT * 算法的多无人机 协同航迹规划方法

于冰¹, 赵宇飞², 韩云龙¹, 陈侠¹

(1. 沈阳科技学院 基础与通识教学部, 沈阳 110167;
2. 中国人民解放军第 5713 工厂 动力控制装置修理分厂, 湖北 襄阳 441002)

摘要: 多机协同航迹规划作为无人机系统的核心技术, 是提高无人机作战效能的有效手段; 针对该研究, 提出基于改进 RRT * 算法的多无人机协同航迹规划方法; 在采样阶段, 引入目标偏置来优化随机采样点的选取方式, 引导随机树朝着目标点方向生长; 在生成新节点过程中, 加入引力势场来优化新节点的生成方式, 将目标点的引力方向作为随机树的生长方向, 加快算法的收敛速度; 综合考虑多机协同的约束条件, 建立多机协同的航迹规划目标函数, 给出了突发威胁航迹重规划方法; 仿真结果表明, 改进的 RRT * 算法在满足协同约束的条件下, 可以规划出较优的协同航迹。

关键词: 无人机航迹规划; RRT *; 多机协同; 目标偏置

Cooperative Path Planning Methods for Multiple UAVs Based on the Improved RRT * Algorithm

YU Bing¹, ZHAO Yufei², HAN Yunlong¹, CHEN Xia¹

(1. Department of Basic and Liberal Studies, Shenyang Institute of Science and Technology, Shenyang 110167, China;
2. Power Controls Repair Sub-plant, the 5713th Factory of Chinese People's Liberation Army, Xiangyang 441002, China)

Abstract: As a core technology of multiple unmanned aerial vehicles (UAVs), multi-aircraft cooperative track planning is an effective means to improve the combat efficiency of UAVs. For this issue, a multi-UAV cooperative track planning method based on improved rapidly-exploring random trees star (RRT *) algorithm is proposed. In the sampling stage, introducing the goal-biased strategy to optimize the selection of sampling points, and guiding the random tree to grow towards the target point. In the process of generating new nodes, the gravitational direction is added to optimize the generation mode of new nodes, and the gravitation mode of target points is used as the growth direction of the random tree to accelerate the convergence speed of the algorithm. By considering the constraints of multi-UAV cooperation, an objective function of multi-UAV cooperation is established, and a sudden threat replanting method is also presented. Simulation results show that the improved RRT * algorithm can plan optimal cooperative trajectories under cooperative constraint conditions.

Keywords: UAV trajectory planning; RRT *; multi-UAV collaboration; target bias

0 引言

自主攻击技术是协同作战的重要组成部分^[1], 而航迹规划作为无人机技术的关键技术之一, 一直受到国内外学者的青睐。多机协同航迹规划在综合考虑空间约束

及时间约束等各种约束条件为每一架无人机规划出从起点到终点的安全较优航迹^[2]。它是一个多目标的优化问题, 不仅规划出单无人机可行的航迹, 还要实现多架无人机之间的协同条件。不但需要规避障碍物以及一些突发的威胁, 而且还要协调各机航迹之间的相互关系, 比

收稿日期:2024-11-06; 修回日期:2024-12-31。

基金项目:国家自然科学基金资助项目(61906125)。

作者简介:于冰(1992-), 女, 硕士, 讲师。

通讯作者:韩云龙(1980-), 男, 硕士, 副教授。

引用格式:于冰, 赵宇飞, 韩云龙, 等. 基于改进 RRT * 算法的多无人机协同航迹规划方法[J]. 计算机测量与控制, 2025, 33(12): 215-223.

如说时间协调关系、空间协调关系等。

目前国内外对多无人机航迹规划的研究已经较为深入，并取得了一些有价值的研究成果^[3-9]。Oyana 等人^[3]提出了一种新颖的三层航迹规划方法，将现有的两种路径规划方法（全局路径和局部路径）融合到一个问题中，用于多无人机路径规划，其算法具有通用性、鲁棒性和稳定性。牛佳伟等人^[4]提出利用协同 A* 算法避开战场上的各种威胁，建立路径空间模型，第一次将协同 A* 算法应用在多机协同航迹规划中。针对传统的多机协同规划避障方法以及常规的智能避障方法的局限性，徐文钰等人^[5]在鸽群优化算法中融入了人工势场法，找到了最优的预测距离，在多架无人机遇到突发威胁时，它可以为无人机规划一条无碰撞且能够安全到达任务区域的航迹。Chen 等人^[6]将遗传算法与粒子群相结合，给出了三维轨迹规划方法。Goel 等人^[7]依据萤火虫群优化思想，给出了动态环境下三维轨迹规划方法。Zhang 等人^[8]采用鸽群优化算法给出协同规划方法等。

快速扩展随机树（RRT, rapidly-exploring random trees）算法是基于采样的规划算法的典型代表^[9]，近些年来受到了有关学者的关注^[10-13]。该算法是在 20 世纪末由爱荷华州立大学的 LaValle 教授提出的。它是一种在高维空间中也能有效规划航迹的算法。它能够迅速找到一条可行的路径，而且可以解决非完整和动力学问题，因而被广泛应用于无人机路径规划。它具有原理简单、适用性强、适合多维空间、无需预知环境也可完成对无人机的路径规划等优点，但是 RRT 算法在空间中随机扩展，会出现搜索时间过长、路径较长且不稳定等问题，于是，有关学者开始对 RRT 算法进行改进来满足实际要求^[14-17]。2010 年，Karaman^[15]提出了渐进最优快速搜索随机树（RRT*）算法。该具有渐近最优性性质，引起国内外学者的广泛关注，但是它仍然存在收敛速度慢节点、利用率低等问题。Islam 等人^[16]为了更快地收敛到最优解，在 RRT* 算法的基础上加入路径优化和智能采样策略，提出了 RRT*-Smart 算法。杨力等人^[17]将 PH 曲线与 RRT 相结合，给出新航迹规划方法等。虽然对 RRT 算法的研究已经取得了初步成果，但关于改进 RRT 算法的二维航迹规划的研究成果偏多，而有关这方面的三维航迹规划甚少，尤其是改进 RRT 算法的多无人机三维协同航迹规划还没有文献报道。

针对上述问题，本文提出基于改进 RRT* 算法的多无人机协同航迹规划方法。首先改进 RRT 算法，引入目标偏置来优化随机采样点的选取方式，引导随机树朝着目标点方向生长。再加入引力势场来优化新节点的生成方式，将目标点的引力方向作为随机树的生长方向，加快算法的收敛速度。并给出了多机协同规划的约束条件及目标函数，并给出突发威胁航迹重规划方法以

及重规划的流程。仿真验证了提出算法的有效性。

1 航迹规划模型的建立

本文在多机协同的前提下考虑了突发威胁的情况，下面给出突发威胁模型。

1.1 任务环境建模

下面绘制两种环境如图 1 所示，用山峰作为障碍物的模型，如式（1）所示：

$$z(x, y) = \sum_e h_e \left\{ \exp \left[- \left(\frac{x - x_e}{x_{se}} \right)^2 - \left(\frac{y - y_e}{y_{se}} \right)^2 \right] \right\} \quad (1)$$

其中： h_e 为地图中第 e 座障碍物的垂直高度， (x_e, y_e) 是山峰的中心位置坐标， x_{se} 为山峰在 x 轴方向的坡度向量， y_{se} 为山峰在 y 轴方向的坡度向量， (x, y) 表示空间中投影到平面的每个点的坐标， $z(x, y)$ 为地形中每个点的高度。

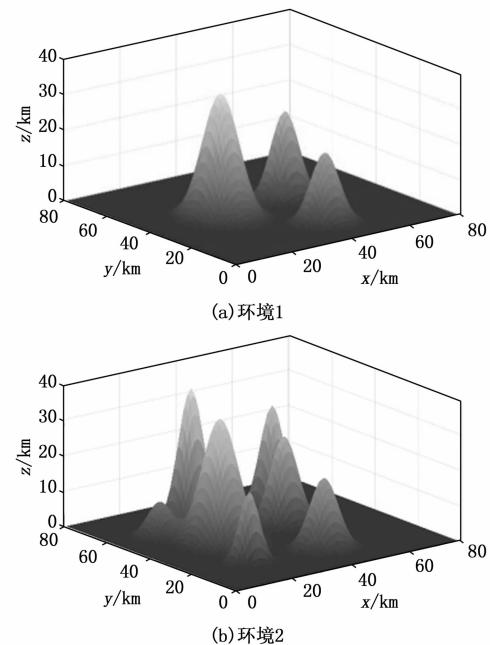


图 1 环境地形图

1.2 突发威胁模型的建立

当多架无人机按照规划好的航迹飞行时，在途中可能会遇到一些不可预料的威胁，这些突发威胁会严重影响无人机的飞行安全，所以需要考虑任务环境中的这些突发威胁。当突发威胁逼近的时候，则被无人机系统探测到，这时无人机要进行实时避障，以达到规避威胁的目的。现实环境中的突发威胁涵盖许多种情况，主要有天气情况和雷达。一般来说，突发威胁多种多样，几乎无法用具体的表达式进行描述，本文将突发威胁简化为一个球体，其表达式如式（2）：

$$(x_u - x_{wo})^2 + (y_u - y_{wo})^2 + (z_u - z_{wo})^2 = r_{wo}^2 \quad (2)$$

其中： (x_u, y_u, z_u) 为球形威胁表面的点的坐标，

(x_{wo}, y_{wo}, z_{wo}) 为球体中心坐标, r_{wo} 为突发威胁的影响半径。

2 改进 RRT 算法

本文提出的新算法在三维空间中, 为了减少路径代价, 加入了在重新选择父节点和重新布线以及扩大父节点的选择范围等措施, 然后将目标偏置策略引入到三维环境的 RRT 算法中, 实现快速抵达目标点的目的。再次加入引力势场法改进 x_{new} 的生成方式, 使其目标点对 x_{near} 产生引力, 加快算法的搜索效率。

1) 为了选取质量更高的节点, 在三维环境中加入了对新节点重新选择父节点和重新布线的过程, 其原理如图 2 (a) 和图 2 (b) 所示。

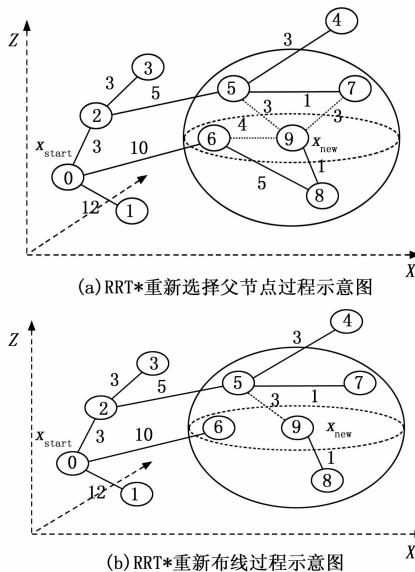


图 2 RRT* 重新更新节点过程示意图

图 2 中, 节点中的标号代表了产生该节点的顺序。图 3 中, x_{new} 是新节点, 在球域内从在 5~8 节点中选择一个 x_{new} 的父节点, 使得初始节点与 x_{new} 之间路径代价最小。不难看出, 新路径 9-5-2-0 的路径代价为 11, 而原始路径 9-8-6-0 代价为 16, 通过比较得知, 所以 5 节点作为 x_{new} 的父节点。在图 4 中, 重新考虑是否将 x_{new} 作为 6~8 的父节点, 计算 6~8 中每一节点以 x_{new} 作为父节点到 x_{start} 的路径代价与原始路径代价进行比较, 若前者小的话, 则选择 x_{new} 作为这个节点的父节点, 否则, 不更新子节点。图 4 中对于节点 8, 比较路径 8-6-0 和 8-9-5-2-0 的路径代价, 计算得知, 8-9-5-2-0 路径代价更小, 故节点 8 作为 x_{new} 的子节点。

2) 依据 Quick-RRT* 算法, 扩大了父节点的选择范围, 使其生成的航迹更短。在以 x_{new} 为球心的球域内, x_{new} 选取父节点的范围不仅包括 x_{near} , 还包括 x_{near} 的深度为 n 的祖先节点的集合。图 3 (a) 和图 3 (b) 分别为 x_{near} 深度 n 是 1 和 2 的祖先节点。

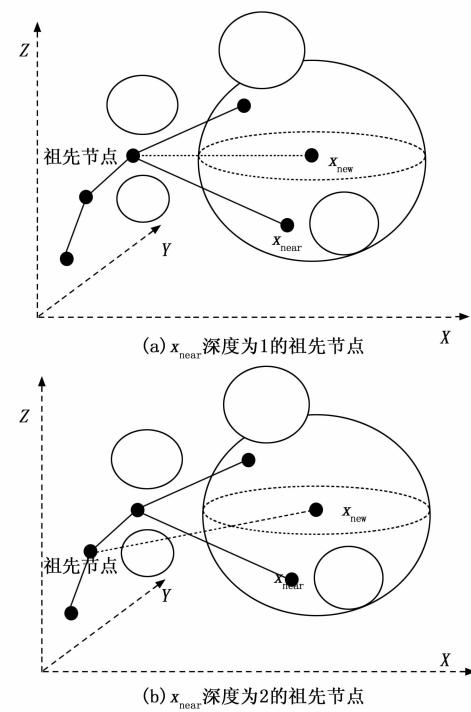


图 3 选取祖先节点示意图

在扩大选取父节点范围过程中, 先找出 x_{new} 节点的集合 x_{near} , 再找出该集合中的每一个节点的 nn 级父节点, 构成集合 $x_{nearest}$, 并将集合 x_{near} 与集合 $x_{nearest}$ 进行合并, 则称合并集合为 x_{new} 重选父节点的范围集合。

3) 将目标偏置策略引入到三维环境的 RRT 算法中, 实现快速抵达目标点的目的。在三维环境的 RRT 算法的采样过程中, 引入目标偏置来优化随机采样点的选取方式, 算法不会消耗大量时间搜索无用区域, 快速向目标点扩展。引入目标偏置后的 x_{rand} 按照公式 (3) 选取:

$$x_{rand} = \begin{cases} x_{goal} & p > p_{target} \\ rand & p \leq p_{target} \end{cases} \quad (3)$$

其中: p 是 $0 \sim 1$ 之间的数, p_{target} 是在三维环境下设定一个目标偏向阈值, 若 p 的值大于设定的偏向阈值, 则将目标点 x_{goal} 作为采样节点; 反之, 则将 $rand()$ 产生的随机点作为采样节点。

4) 加入引力势场法改进 x_{new} 的生成方式, 目标点 x_{goal} 对 x_{near} 产生引力, 设 i 是一个航迹点, 则引力场函数如式 (4) 所示:

$$U_{att}(i) = \frac{1}{2}ka p_g^2(i) \quad (4)$$

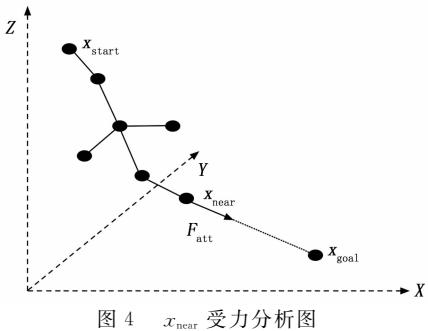
其中: ka 为目标点产生的引力场增益常数, $p_g(i)$ 为点 i 与目标点之间的距离。公式 (5) 和 (6) 分别为引力及势场函数的负梯度:

$$F_{att}(i) = gradU_{att} \quad (5)$$

$$F_{att}(i) = ka p_g(i) \quad (6)$$

通过结合引力势场法, 目标点 x_{goal} 对最近邻点 x_{near}

产生吸引力, 引力方向决定了 x_{new} 的生成方向。 x_{near} 的受力情况如图 4 所示。

图 4 x_{near} 受力分析图

5) 新算法流程如图 5 所示, 新初始化, 设 M 为最大迭代次数、目标节点等算法参数, 然后引入目标偏置策略, 选取随机采样点 x_{rand} , 确定最近邻点 x_{near} , 按照改进方法计算新节点 x_{new} , 然后对 x_{new} 重新选择父节点并重新布线, 如果新选择的父节点和 x_{new} 之间没有障碍物, 则将 x_{new} 加入随机树中, 否则需要重新采样。最后再检测 x_{new} 是否到达目标点或达到规定最大迭代次数, 若没有实现到达目的地, 则重新采样, 否则算法结束。

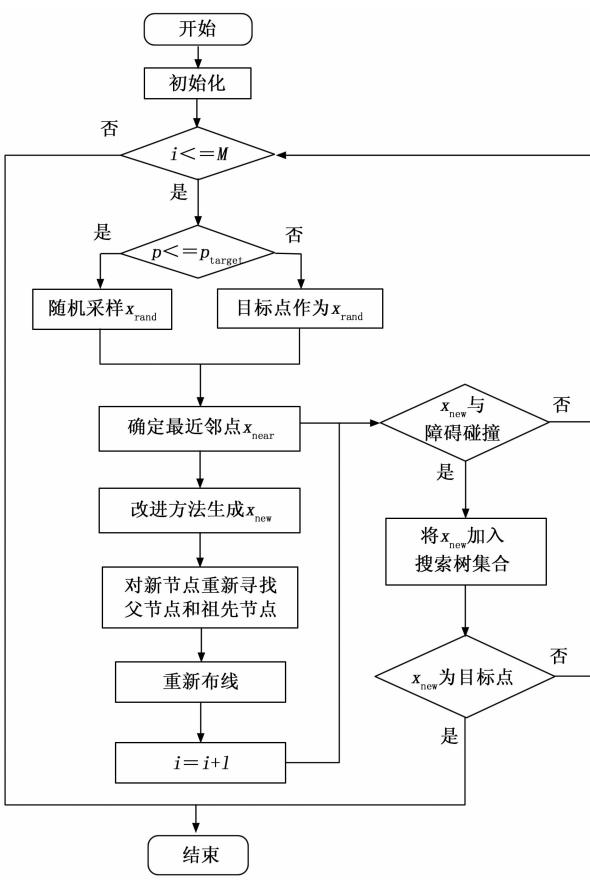


图 5 改进 RRT 算法流程图

3 改进 RRT 算法的多机航迹规划

3.1 无人机时间协同与空间协同的约束条件

多架无人机共同协作过程中, 需要考虑两机之间的

距离要大于设定的安全距离, 即满足空间协同约束。时间协同约束常常是指在一定的时间范围内多机同时到达指定目标点, 提高协同作战的效率。无人机需要避开任务环境中的各种障碍威胁。图 6 为协同航迹规划示意图。

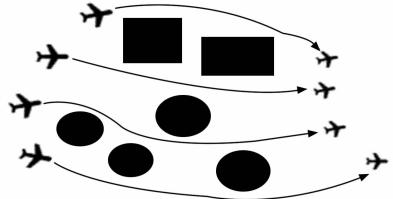


图 6 协同航迹规划示意图

3.1.1 时间协同约束

假定某次作战任务需要多架无人机共同协作去完成, 为了作战任务的隐秘性, 通常需要多架无人机沿着不同的路径飞往目标点。时间协同约束常常是指在一定的时间范围内多机同时到达指定目标点。令第 j 架无人机的航迹长度为 L_j , 第 j 架无人机的速度为 V_j ($V_{\min} \leq V_j \leq V_{\max}$), 则第 j 架无人机到达任务地点的飞行时间范围为 $t_j = [L_j/V_{\max}, L_j/V_{\min}]$, 则时间协同约束为:

$$T = \bigcap_{j=1}^n t_j \quad (7)$$

其中: T 为协同时间, n 为无人机的架数。

3.1.2 空间协同约束

多架无人机共同协作过程中, 需要考虑两机之间的距离要大于设定的安全距离, 即满足空间协同约束, 否则可能发生无人机碰撞或相互干扰。空间协同表达式如下:

$$|d_j(t) - d_k(t)| \geq d_{\min} \quad (8)$$

其中: $d_j(t)$ 为第 j 架无人机在 t 时刻的位置, $d_k(t)$ 为第 k 架无人机在 t 时刻的位置, d_{\min} 为两架无人机之间允许的最短距离。

3.2 多无人机协同航迹规划约束条件

在进行多机协同航迹规划时, 需要综合考虑其约束条件, 以确保无人机的飞行安全。多无人机协同航迹规划的约束条件如式 (9) 所示:

$$\begin{cases} b_l: L_j \leq L_{\max}, j = 1, 2, 3, \dots, n \\ b_{\varphi}: \varphi_j \leq \varphi_{\max}, j = 1, 2, 3, \dots, n \\ b_{\theta}: \theta_j \leq \theta_{\max}, j = 1, 2, 3, \dots, n \\ b_v: V_{\min} \leq V_j \leq V_{\max}, j = 1, 2, \dots, n \\ b_h: H_{\max} \geq V_{j,t} \geq H_{\min}, j = 1, 2, \dots, n \\ b_r: 0 < \frac{R_{\max}^4}{R_j^4 + R_{\max}^4} < \frac{1}{2}, j = 1, 2, \dots, n \\ b_t: T = \bigcap_{j=1}^n t_j \\ b_d: |d_j(t) - d_k(t)| \geq d_{\min} \end{cases} \quad (9)$$

其中: L_j 和 L_{\max} 分别为第 j 架飞机的航迹总长度及

最大航迹长度。 φ_j 为相邻两航迹段之间的夹角, φ_{\max} 为无人机允许的最大偏航角。 θ_j 为第 j 架无人机航迹上任一航迹段的俯仰角, θ_{\max} 为无人机飞行过程中的最大俯仰角。 V_{\min} 为无人机的最小速度, V_{\max} 为最大速度。 H_{\min} 为最小飞行高度, 无人机低空飞行时, 由于地形复杂可能有坠毁的风险, H_{\max} 为允许无人机飞行的最大高度。 R_j 表示在战场环境中第 j 架无人机与雷达之间的实际距离, R_{\max} 为雷达能检测到的最远距离。

约束条件可以用一个约束集合 B 表示:

$$B = \{b_t, b_{\varphi}, b_{\theta}, b_v, b_h, b_r, b_t, b_d\}$$

其中: b_t 为航程约束, b_{φ} 为偏航角约束, b_{θ} 为俯仰角约束, b_v 为速度约束, b_h 为高度约束, b_r 为雷达约束, b_t 为时间协同约束, b_d 为两架飞机飞行的安全距离约束。

3.3 多无人机防碰撞策略

多无人机航迹规划防碰撞策略如图 7 所示。多无人机航迹规划需要防止无人机之间发生碰撞, 而想要避免这种情况出现, 就不能让航迹出现重合或交叉的情况。针对以上问题, 本文设计了一种防碰撞策略, 首先, 利用改进 RRT 算法为无人机规划出可行的航迹, 将该架无人机的各个航迹点放到一个封禁列表中, 而为了避免规划出的航迹有交叉或重合的情况, 之后的无人机不能选取封禁列表中的航迹点; 若航迹之间存在交叉的情况, 则从候选航迹中选择无交叉或重合的航迹进行替换, 候选航迹如图 7 虚线所示, 航迹两侧的虚线分别待两架无人机飞行的候选航迹。若是候选航迹中不存在无碰撞的航迹, 则需要重新进行无人机的航迹规划。

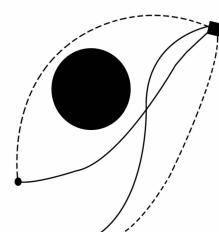


图 7 防碰撞策略示意图

3.4 目标函数的建立

建立的协同目标函数包括两部分: 一部分是各架无人机的航迹长度; 另一部分为协同时间 T_{arrival} , 即交集中的最小 $\min(T)$ 。由航迹点的定义可得第 j 架无人机的航迹长度为:

$$\left\{ \begin{array}{l} L_j = \sum_{i=1}^{m-1} d_j(Q_i, Q_{i+1}) \\ d_j(Q_i, Q_{i+1}) = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \end{array} \right. \quad (10)$$

其中: $d_j(Q_i, Q_{i+1})$ 为第 j 架无人机的航迹上两航迹点之间的欧几里得距离。

为了将多个约束的优化问题转化为非约束优化问题, 引入惩罚函数及惩罚因子 Penalty , Penalty 可表示为:

$$\text{Penalty} = \sum_{j=1}^n g_j, n = 1, 2, 3 \quad (11)$$

$$\text{其中: } g_j = \begin{cases} 0, & \text{满足约束} \\ 1, & \text{不满足约束} \end{cases}$$

则建立的目标函数如下:

$$\min J = \mu_1 \sum_{j=1}^n \frac{L_j}{L_{\max}} + \mu_2 \frac{T_{\text{arrival}}}{T_{\max}} + 100000 \text{Penalty} \quad (12)$$

其中: J 为设定的目标函数值, n 为无人机的架数。 T_{\max} 为飞机到达目标点所用的最长时间。公式 (12) 中的惩罚因子 Penalty 乘了一个很大的数, 显而易见, 只有当 Penalty 为 0 时才能取得最优解。 μ_1, μ_2 为权重系数, 且满足 $\mu_1 + \mu_2 = 1$ 。

3.5 突发威胁航迹重规划

当无人机在飞行环境中遇到突发威胁时, 如果不能及时的规避, 则有坠毁的风险。所以需要重新规划航迹, 此时需偏离原来的航线, 沿着新航线飞行。动态协同航迹规划是指无人机在线进行航迹规划, 由于飞机飞行的同时在线规划路径, 允许规划的时间非常短, 所以更侧重规划的时效性。由于战场上的环境复杂多变, 无人机先沿着既定航迹飞行, 遇到突发威胁时再改变航迹。通过式 (13) 判断航迹点是否在威胁范围内:

$$\sqrt{(x_i - x_{wo})^2 + (y_i - y_{wo})^2 + (z_i - z_{wo})^2} \leq r_{wo} + \Delta s \quad (13)$$

其中: (x_i, y_i, z_i) 为航迹点坐标, (x_{wo}, y_{wo}, z_{wo}) 为球体中心点的坐标, r_{wo} 为球体半径, Δs 是安全距离。

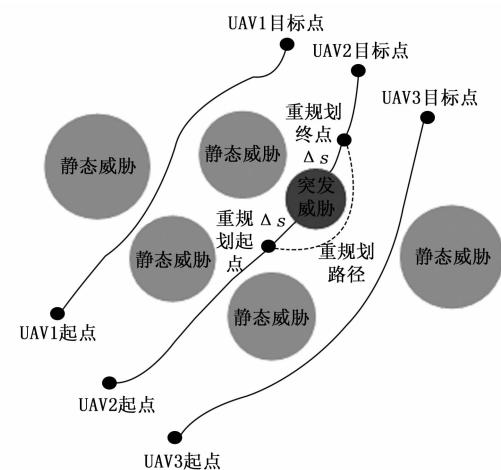


图 8 无人机航迹重规划示意图

在图 8 中, Δs 是威胁球体半径与重规划起点和终点到突发威胁球体中心距离之间的差值。设置安全距离主要是为了确保无人机在重规划起点和终点时的安

全性。

若无人机遇见突发威胁时, 其坐标满足公式 (13), 则将威胁区域中的航迹段记录下来, 此航迹段的起点和终点作为重规划的路段。

多机在按照规划好的航迹飞行的过程中, 如果无人机突然检测到前面有威胁的话, 则需要在原始航迹上局部进行重规划, 其流程如图 9 所示。

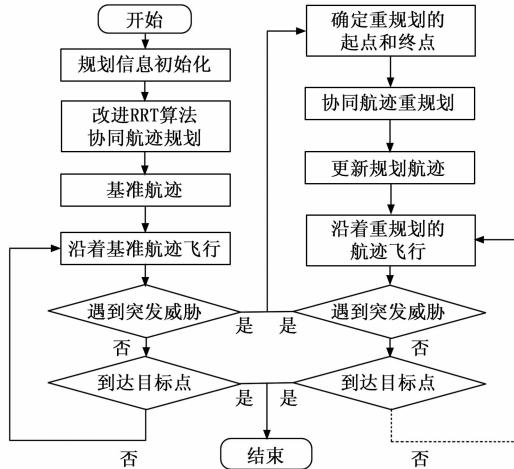


图 9 无人机航迹重规划流程图

具体操作步骤如下:

- 1) 规划信息初始化, 建立突发威胁模型。
- 2) 根据多机协同的约束条件, 结合环境信息, 采用改进的 RRT 算法规划出多机协同三维的基准航迹。
- 3) 无人机按照规划的航迹飞行, 并对周围环境进行实时感知, 若发现突发威胁, 则执行 4), 否则执行 7)。
- 4) 由式 (13) 计算出新增威胁的信息, 将航迹段记录下来, 此航迹段的起点和终点作为重规划的路段。
- 5) 使用改进算法依据 4) 的起点与终点, 在满足约束条件下进行重规划。
- 6) 更新当前遇见突发威胁的飞行轨迹, 使得无人机沿着重规划的航迹向目标点飞行。
- 7) 如果每架飞机沿着规划航线抵达任务地点, 则协同航迹规划结束, 否则执行 3)。

3.6 基于改进 RRT * 算法的多无人机协同航迹规划流程

图 10 为协同航迹规划流程图。操作步骤如下:

- 1) 建立模型, 包括任务环境建模和突发威胁模型的建立; 初始化规划信息, 如各架无人机的起点和终点的位置信息。
- 2) 给出多机协同航迹规划的约束条件。
- 3) 建立多机协同目标函数。
- 4) 使用改进的 RRT 航迹规划算法规划初始航迹。
- 5) 如果飞机遇到突发威胁, 则进行下一步, 否则,

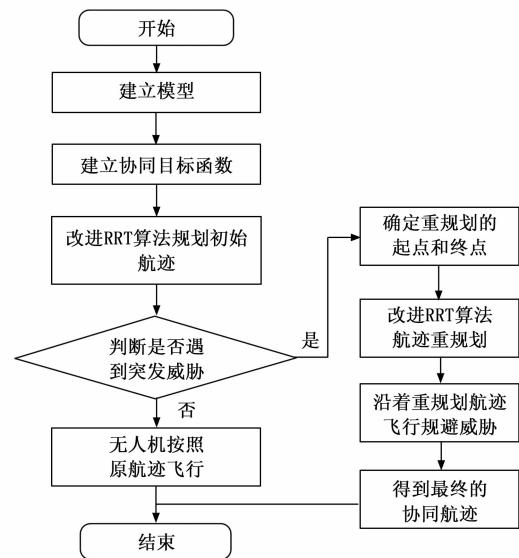


图 10 改进 RRT 算法的多无人机协同航迹规划流程图

则按照原来规划的原始航迹正常飞行。

6) 获取重规划的起点和终点信息。

7) 使用新算法对遇到突发威胁的进行局部航迹重规划。

8) 无人机避开突发威胁后继续按照原来的航迹飞行。

4 仿真分析

4.1 改进 RRT 算法仿真分析

本文使用软件 Matlab 进行仿真, 设置两种仿真环境, 分别是环境 1 和环境 2。仿真设置起点坐标为 (5, 70, 5), 目标点坐标为 (80, 30, 10)。两种环境的障碍物参数设置如表 1 和表 2 所示。将改进算法分别在环境 1 和环境 2 中和原始 RRT 算法、加入目标偏置的 RRT 算法 (GB-RRT * 算法) 仿真对比, 环境 1 仿真结果如图 11 (a) 和 (b) 所示。

表 1 环境 1 障碍物信息

序号	位置/km	x_{se}	y_{se}
1	(25, 40, 20)	7	8
2	(50, 45, 20)	6	5
3	(30, 60, 20)	5.5	4.5
4	(45, 20, 20)	6	5.5

表 2 环境 2 障碍物信息

序号	位置/km	x_{se}	y_{se}
1	(10, 10, 20)	5	5.5
2	(25, 40, 35)	7	8
3	(50, 45, 25)	6	5
4	(30, 60, 38)	5.5	4.5
5	(45, 20, 20)	6	5.5
6	(10, 20, 25)	4.5	3.5

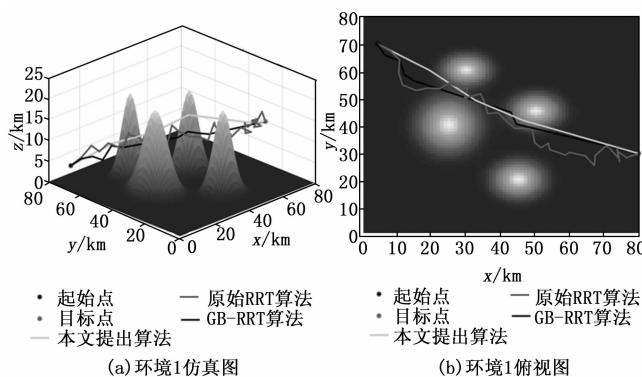


图 11 环境 1 仿真图

图 11 中, GB-RRT* 算法和本文算法与原始 RRT 相比, 生成的路径明显缩短。GB-RRT* 算法在 RRT 算法的基础上加入了目标偏置策略, 但它冗余点仍然较多, 而改进算法在引入目标偏置的基础上加入引力势场并结合 Quick-RRT* 算法, 规划出的航迹冗余点较少。

为了更准确地分析数据, 对原始 RRT 算法、GB-RRT* 算法及本文算法分别运行了 30 次, 记录算法的运行时间、航迹长度和迭代次数。用箱线图来对比 3 种算法的实验数据, 环境 1 的数据分析如图 12 和表 3 所示, 表 3 为对比 3 种算法的实验数据, 是 30 组实验数据的平均值, 显而易见, GB-RRT* 算法比 RRT 算法更加优化, GB-RRT* 算法规划的平均航迹长度比原始 RRT 算法降低了 28.8%, 平均运行时间降低了 54.4%, 平均迭代次数减少了 81.4%。而改进的算法与 GB-RRT* 算法相比, 平均航迹长度降低了 10.4%, 平均运行时间降低了 59.2%, 平均迭代次数减少了 27.3%, 证明改进算法的性能比 GB-RRT* 算法更好。改进算法与原始 RRT 算法相比, 3 种指标的降低幅度

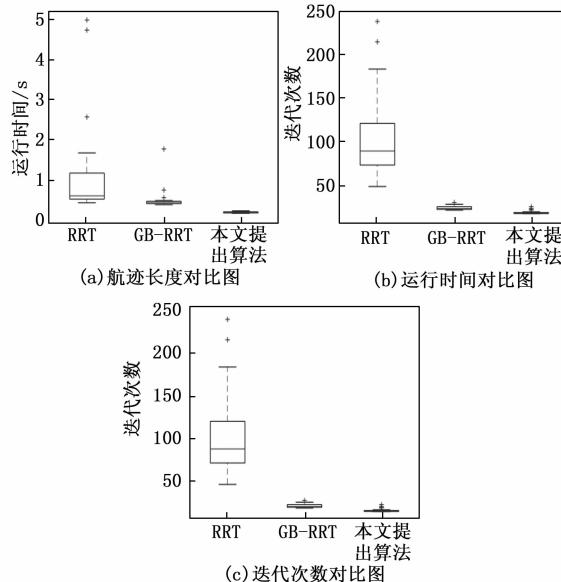


图 12 环境 1 数据对比图

更大, 分别为 36.2%、81.4% 以及 84.8%, 有大幅度的下降。数据对比的结果表明, 新算法在各性能指标上均比其它两种算法更加优化, 证明了改进算法的优越性大幅提升。

表 3 环境 1 中 30 组实验平均值

算法	平均航迹 长度/km	平均运行 时间/s	平均迭代 次数
原始 RRT	138.5065	1.1223	105
GB-RRT*	98.5990	0.5120	22
本文	88.3579	0.2091	16

与环境 1 相比, 环境 2 中存在着较多障碍物, 环境 2 地图设置为 80 km×80 km×40 km 的三维空间, 环境 2 仿真设置的起点坐标是 (5, 70, 5), 目标点坐标为 (80, 30, 10), 如图 13 所示。

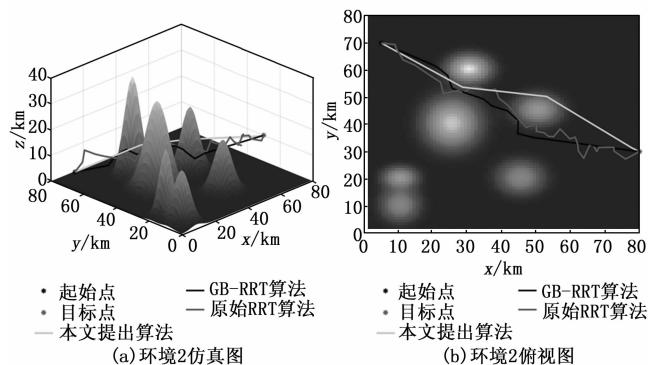


图 13 环境 2 仿真图

同样取 30 组数据, 记录环境 2 中的 3 种指标, 箱线图如图 14 所示, 算法运行 30 次所得数据的平均值如表 4 所示, 可以看出, 改进算法在 3 种指标上都有所降低, GB-RRT* 算法的平均航迹长度比原始 RRT 算法

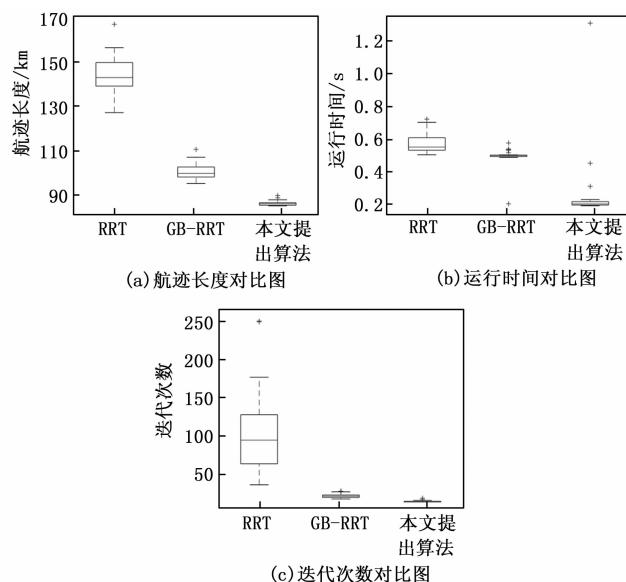


图 14 环境 2 数据对比图

降低了 29.6%，平均运行时间降低了 14.3%，平均迭代次数减少了 77.2%。与 GB-RRT* 算法相比，新算法平均航迹长度降低了 13.6%，平均运行时间降低了 49.7%，平均迭代次数减少了 30.4%。因此，新算法规划的航迹性能有大幅度的提升。

表 4 环境 2 中 30 组实验平均值

算法	平均航迹 长度/km	平均运行 时间/s	平均迭代 次数
原始 RRT	143.626 5	0.573 6	101
GB-RRT*	101.105 7	0.491 4	23
本文	87.402 6	0.247 1	16

4.2 多无人机航迹规划仿真分析

为了验证算法的有效性，每架无人机的起点和终点都不同，设无人机的最大速度 $V_{\max} = 250$ m/s，最小速度 $V_{\min} = 120$ m/s。两种环境的障碍物参数如表 5 和表 6 所示，两种环境中无人机的起点和终点的位置如表 7 和表 8 所示。

表 5 环境 1 障碍物参数

序号	位置/km	x_{se}	y_{se}
1	(25, 40, 35)	8	7
2	(50, 43, 25)	5	6
3	(45, 18, 20)	5.5	6

表 6 环境 2 障碍物参数

序号	位置/km	x_{se}	y_{se}
1	(15, 55, 10)	5	5
2	(25, 40, 35)	8	7
3	(50, 43, 25)	5	6
4	(30, 60, 38)	4.5	5.5
5	(45, 18, 20)	5.5	6
6	(20, 20, 20)	3.5	4.5
7	(55, 55, 30)	5	5

表 7 环境 1 中无人机起始点与目标点位置信息

无人机序号	起始点坐标	目标点坐标
1	(2, 55, 7)	(68, 42, 23)
2	(3, 35, 5)	(70, 30, 30)
3	(2, 25, 6)	(65, 23, 25)
4	(2, 5, 5)	(65, 5, 20)

表 8 环境 2 中无人机起始点与目标点位置信息

无人机序号	起始点坐标	目标点坐标
1	(5, 45, 5)	(70, 45, 25)
2	(3, 35, 5)	(70, 30, 30)
3	(5, 23, 5)	(65, 20, 20)
4	(4, 5, 5)	(70, 10, 10)

环境 1 的多无人机协同航迹规划方法的仿真结果如图 15 (a) 和 (b) 所示。由图可知，改进的 RRT 算法规划出的航迹拐点较少，路径较优，能够以较短的航程到达目标点，且多架无人机之间能够保持安全的飞行间距，在飞行过程中没有出现航迹段的重叠和碰撞威胁，在满足协同约束情况下很好地实现多无人机航迹规划。

对环境 1 中的协同规划仿真结果进行整理分析可得到表 9 中的数据。由各架无人机的航迹长度和无人机的飞行速度可以得出无人机的到达时间，由表 9 可知，各架无人机在区间 [298.0, 550.8] 内协同到达目标点。

表 9 环境 1 协同规划仿真结果

无人机序号	航迹长度/km	到达时间/s	协同到达时间/s
1	74.5	[298.0, 620.8]	[298.0, 550.8]
2	72.2	[288.8, 601.7]	
3	66.1	[264.4, 550.8]	
4	66.4	[265.6, 553.3]	

环境 2 的多无人机协同航迹规划方法的仿真如图 15 (c) 和 (d) 所示。与环境 1 相比，环境 2 中的障碍威胁更多，环境更复杂一些，这就导致了路径长度以及多无人机的协同时间也会有所增加。对环境 2 中的协同规划仿真结果进行整理分析可得到表 10 中的数据。由表 10 可知，各架无人机在区间 [290.0, 525.0] 内协同到达目标点。

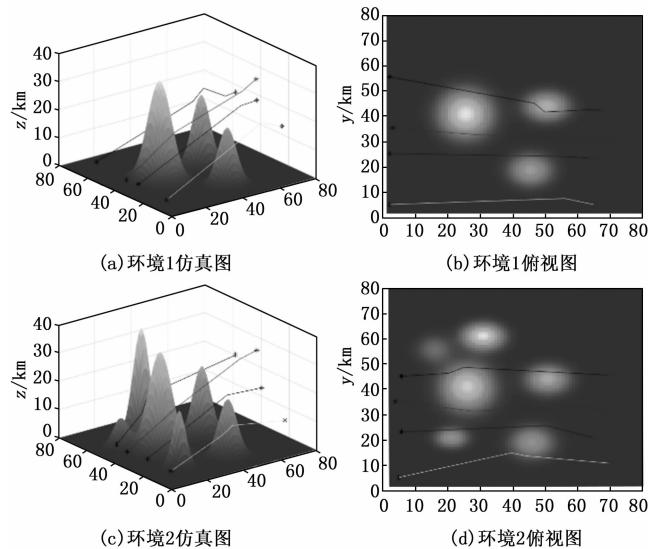


图 15 多无人机航迹规划仿真图

当无人机沿着预定航线飞行时，其中一架无人机的路径上突然出现威胁，此时必须重新规划航线以规避威胁。依据重规划的起点与终点，在满足约束条件下进行重规划。分别在两种环境中对突发威胁下的多机协同进行仿真，仿真图如图 16 所示。图 16 中，黑色球体为突

发威胁, 图 16 中间位置的浅灰色线条即为改进 RRT 算法重新规划后的航迹。环境 1 突发威胁仿真运行时间为 0.32 s, 环境 2 突发威胁仿真运行时间为 0.49 s, 说明本文改进 RRT 算法能够快速重规划得到避开突发威胁的较优航迹。

表 10 环境 2 协同规划仿真结果

无人机序号	航迹长度/km	到达时间/s	协同到达时间/s
1	70.2	[280.8, 585.0]	[290.0, 525.0]
2	72.5	[290.0, 604.2]	
3	63.0	[252.0, 525.0]	
4	68.7	[274.8, 572.5]	

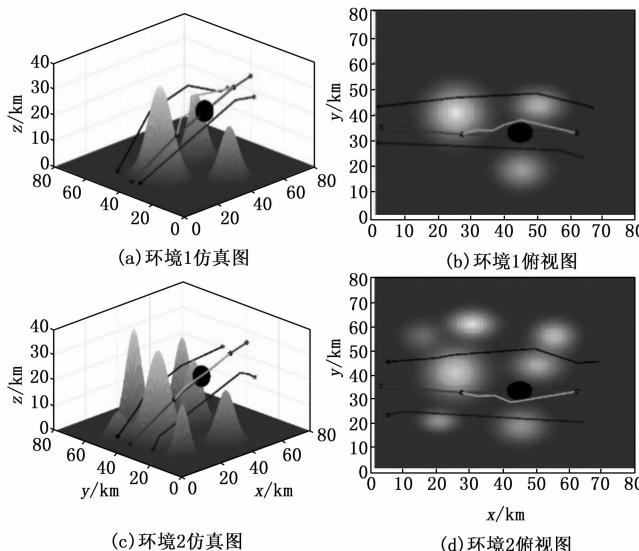


图 16 突发威胁仿真图

5 结束语

本文研究了多无人机协同三维航迹问题。建立了多机协同航迹规划目标函数模型及协同约束条件, 提出了一种改进 RRT 算法的多无人机协同三维航迹规划方法。并给出了满足约束的突发威胁航迹重规划方法以及重规划的流程。仿真结果表明, 提出的改进 RRT 算法在满足协同约束的条件下, 可以规划出较优的协同飞行航迹, 并且能够快速地规避突发威胁。为了更加切合实际情况, 现实环境中不仅有静态威胁, 还有移动威胁。今后将对规避移动威胁的情况进行深入研究。

参考文献:

- [1] DAI L, CAO Q, XIA Y, et al. Distributed MPC for formation of multi-agent systems with collision avoidance and obstacle avoidance [J]. Journal of the Franklin Institute, 2017, 354 (4): 2068–2085.
- [2] LIU Y, LAKSHMI N S, ROBERT J E, et al. Computational evaluation of the steady and pulsed Jet effects on the performance of a circulation control wing section [R]. AIAA-2004-56, 2004.
- [3] OYANA S, LI J, USMAN M. Three-layer multi-UAVs path planning based on ROBL-MFO [J]. Guidance, Navigation and Control, 2022, 2 (3): 106–134.
- [4] 牛佳伟. 基于协同 A* 算法的多无人机航迹规划 [J]. 舰船电子工程, 2021, 41 (9): 66–68.
- [5] 徐文钰, 敖海跃, 刘燕斌. 基于鸽群优化算法的多无人机局部航迹重规划 [J]. 战术导弹技术, 2022, 1: 46–52.
- [6] CHEN Y B, YU J Q, MEI Y S, et al. Modified central force optimization (MCFO) algorithm for 3D UAV path planning [J]. Neurocomputing, 2006, 171: 878–888.
- [7] GOEL U, VARSHNEY S, JAIN A, et al. Three dimensional path planning for UAVs in dynamic environment using glow-worm swarm optimization [J]. Procedia Computer Science, 2018, 133: 230–239.
- [8] ZHANG D F, DUAN H B. Social-class pigeon-inspired optimization and time stamp segmentation for multi-UAV cooperative path planning [J]. Neurocomputing, 2018, 313: 229–246.
- [9] 王全. 基于 RRT 的全局路径规划方法及其应用研究 [D]. 长沙: 国防科技大学, 2014.
- [10] 马新国, 马希青. 融合改进 RRT 和 Dijkstra 算法的机器人动态路径规划 [J]. 组合机床与自动化加工技术, 2023 (2): 5–9.
- [11] 宋晓琳, 周南, 黄正瑜, 等. 改进 RRT 在汽车避障局部路径规划中的应用 [J]. 湖南大学学报 (自然科学版), 2017, 44 (4): 30–37.
- [12] 冯楠. 自主移动机器人路径规划的 RRT 算法研究 [D]. 大连: 大连理工大学, 2014.
- [13] 刘成菊, 韩俊强, 安康. 基于改进 RRT* 算法的 RoboCup 机器人动态路径规划 [J]. 机器人, 2017, 39 (1): 8–15.
- [14] 李加东. 基于 RRT 算法的非完整移动机器人运动规划 [D]. 上海: 华东理工大学, 2014.
- [15] KARAMAN S, FRAZZOLI E. Sampling-based algorithms for optimal motion planning [J]. The International Journal of Robotics Research, 2011, 30 (7): 846–894.
- [16] ISLAM F, NASIR J, MALIK U, et al. RRT*-smart: rapid convergence implementation of RRT* towards optimal solution [C] //Mechatronics and Automation (ICMA), 2012 International Conference. IEEE, Chengdu, 2012: 1651–1656.
- [17] 杨力, 丘森辉, 周伟东, 等. 基于改进 PH 曲线的无人机航迹规划 [J]. 计算机仿真, 2018, 35 (8): 1–6.