

基于 Hybrid State A* 与增强安全管道的直升机低空航迹规划

刘冰¹, 方元¹, 高世奇², 廖文韬¹, 赖梓航², 罗灵鲲², 胡士强²

(1. 中国航空无线电电子研究所, 上海 200241;

2. 上海交通大学 航空航天学院, 上海 200240)

[摘要] 直升机低空突防已逐渐成为现代空战察打任务的核心, 而低空航迹规划算法是实现该技术的关键。尽管现有的航迹规划算法已经被应用于实际低空突防任务, 但基于“前端-后端”式的传统航迹规划算法依然存在规划航迹机动执行性差与复杂动态场景下易碰撞的缺陷。针对上述问题, 本文提出一种基于 Hybrid State A* 与增强安全管道的改进算法。首先, 基于 Hybrid State A* 算法的联合轨迹优化, 可以在状态空间中完成兼顾直升机机动特性的初始航迹高效搜索, 有效保证直升机航迹的可达性。其次, 基于初始航迹膨胀的增强安全管道, 将后端航迹优化参数限制在安全的可行域内, 进而有效提升复杂动态场景下规划航迹的安全性。在实验环节, 本研究结合 ROS 机器人仿真环境与 Rviz 数据可视化工具完成仿真验证, 通过算法间的综合对比实验, 论证了本研究所提算法对规划的航迹机动性与安全性有明显的提升。

[关键词] 直升机航路规划; Hybrid State A* 算法; 增强安全管道

[中图分类号] TP273

[文献标识码] A

[文章编号] 1006-141X(2022)03-0015-10

Helicopter Low-Altitude Trajectory Planning Based on Hybrid State A* and Strengthened Safety Pipeline

LIU Bing¹, FANG Yuan¹, GAO Shi-qi², LIAO Wen-tao¹, LAI Zi-hang²,
LUO Ling-kun², HU Shi-qiang²

(1. Chinese Aeronautical Radio Electronics Research Institute, Shanghai 200241, China;

2. School of Aeronautics and Astronautics, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract: Helicopter penetration within low-altitude has been playing a vital role in modern air combat missions, whereas the trajectory planning algorithm remains the kernel process to accomplish such a task. Although the trajectory planning algorithms have been widely applied in solving real penetration tasks, they suffered due to the poor track maneuver execution as well as the collision in the complex dynamic scenes. Indeed, the root of the problems is the weaknesses in the “front-end-back-end” style-induced trajectory planning algorithms. To remedy the problem, a novel algorithm is proposed based on the Hybrid State A* and strengthened safety pipeline techniques. To be specific, the proposed algorithm ensures the efficient searching of the original path within the state space by using the helicopter’s maneuvering characteristics and thus improving the accessibility of the helicopter’s trajectory. Then the strengthened safety pipeline is expanded based on the original path, thereby enforcing the optimized functional learning can efficiently adapt to the complex environmental changes as well as naturally reduce the potentially existing unavoidable collisions.

收稿日期: 2022-07-26

基金项目: 国家自然科学基金 (61773262, 62006152); 中国航空科学基金会 (20142057006)

引用格式: 刘冰, 方元, 高世奇, 廖文韬, 赖梓航, 罗灵鲲, 胡士强. 基于 Hybrid State A* 与增强安全管道的直升机低空航迹规划[J].

Subsequently, in the experimental section, sufficient experimental discussions are presented based on the ROS robot simulation environment and the Rviz data visualization tool to comprehensively analyze the experimental results. As a result, through the comprehensive comparison experiments among different algorithms, it can be witnessed that the proposed algorithm improves the baseline approaches in terms of the maneuverability and safety aspects.

Key words: helicopter trajectory planning; hybrid state A* algorithm; strengthened safety pipeline

随着地空一体式防空体系的完善, 超低空突防愈发成为现代空军的新型察打手段。航迹规划与引导技术作为超低空突防的关键, 成为各国研究的焦点。该技术旨在为飞行器实时规划损耗最低、威胁最小的航迹, 引导飞行器完成既定任务。直升机具有更好的贴地飞行能力, 因而结合航迹规划技术后能充分利用地势掩护躲避敌方雷达侦察, 成为低空突防的利器。虽然现有算法能实现基本飞行引导, 但仍存在规划航迹机动执行性不足与动态威胁适应性弱的问题。

航迹规划本质上属于机器人运动规划问题, 但将现有方法的规划对象推广为直升机时, 将面临规划航迹可执行性的挑战。运动规划包括“前端”与“后端”两部分^[1]。“前端”旨在搜索满足任务点通过性的无碰初始轨迹, 通常搜索算法可分为确定型搜索、随机型采样、满足动力学约束的搜索^[1-2]。由于较大飞行空间会增加随机采样的搜索成本, 当前直升机三维航迹规划主要采用 Dijkstra^[3]、A*^[4]、Theta*^[5] 等确定型搜索算法。任天翔等人提出的改进 Theta* 算法^[6], 通过引入变权重评估函数, 减少航路搜索过程中无效的节点, 提升了航迹搜索效率。王锦博等人提出的改进粒子群算法^[7], 采用惯性权重因子调节算法的全局和局部搜索能力, 并通过选择与杂交增加种群的多样性, 缩短航迹搜索的时间。上述方法虽然提升了航迹搜索效率, 但均在离散栅格空间中进行, 与直升机运动的连续性矛盾。满足动力学约束的搜索算法以 State-Lattice Search^[8]、Hybrid State A*^[9] 为代表, 利用连续状态空间缓和上述矛盾, 成为直升机规划航迹光滑性优化的潜在方向。“后端”则根据机动代价最小准则, 用样条曲线将初始轨迹拟合为光滑度更高的执行轨迹。Mellinger D. 等人^[10] 提出 Minimum Snap 方法并以优化高阶多项式系数保证轨迹平滑, 周炜等人^[11] 则采用 Bezier 曲线实现后端航迹的拟合优化。尽管上述方法提升了航迹光滑度, 但传统“前端-后端”式规划流程对直升机协调转弯与耦合控制等机动特性的考虑较

少, 难以确保光滑航迹的可执行性。因此, 增强规划航迹的光滑性与可执行性, 是现阶段直升机低空航迹规划研究的重点。

同时, 直升机在低空突防时还面临复杂地形与敌方电子侦察等多重威胁, 因此对规划航迹动态安全性的要求更高。针对低空战场威胁, 文献 [12] 将敌方察打威胁建模为“威胁山峰”, 实现了战场与地形威胁模型的统一, 并在威胁地图中完成航迹规划。然而, 上述环境威胁属于静态建模, 未考虑其动态变化性。为确保飞行器在复杂场景下的安全避障, Liu S. 等人^[13] 提出的飞行安全管道可以有效提升飞行安全。文献 [14] 基于规划航迹线段膨胀出以椭球或凸多面几何体表示的安全空间, 将其作为航迹解空间的约束边界, 实现了无人机避障。虽然传统安全管道为规划航迹提供了一定的安全裕度, 但其依赖于初始航迹的搜索质量, 当环境威胁动态变化时, 可能出现安全空间不连续等异常现象。因此, 提升安全管道的膨胀与表征效率, 是确保航迹动态安全性的关键。

本文针对直升机传统“前端-后端”式算法规划航迹机动执行性与动态安全性弱的问题, 提出了基于 Hybrid State A* 与增强安全管道的改进算法, 主要贡献如下:

(1) 针对规划航迹的机动执行性问题, 提出在基于 Hybrid State A* 算法的初始航迹搜索中, 引入直升机机动特性建模和动力学约束;

(2) 针对规划航迹的动态安全性问题, 提出在改进前端搜索算法的基础上, 以最新威胁障碍为边界, 膨胀出增强安全管道, 作为后端航迹优化的参数空间;

(3) 通过前端搜索算法的对比实验, 验证改进前端对规划航迹的有效动力学约束和机动适应性提升。通过安全管道对比与超参数实验, 论证增强安全管道对动态威胁下规划航迹的安全性提升。

1 基于增强安全管道的航迹规划算法

本文分别以 A* 算法^[15] 和基于 B 样条曲线的航迹拟合^[13] 作为传统“前端-后端”式航迹规划流程的具体实现。首先, 基于 Hybrid State A* 算法, 对前端搜索过程引入直升机机动特性与动力学约束, 提升规划航迹的机动执行性。其次, 引入增强安全管道, 确保规划航迹的动态安全性。

1.1 Hybrid State A* 搜索

本节将在连续性更强、兼顾直升机机动特性的状态空间中, 基于 Hybrid State A* 算法进行初始航迹搜索, 以提升规划航迹的机动执行性。

1.1.1 状态空间生成

传统初始航迹通常在栅格空间中搜索, 可行节点位于栅格中心或边缘线, 因而连接形成的初始航迹较为曲折。此外, 随着栅格分辨率的提高, 冗余节点数激增也将造成搜索空间利用率低的问题。因而连续性更强的状态空间是更优的搜索选择。

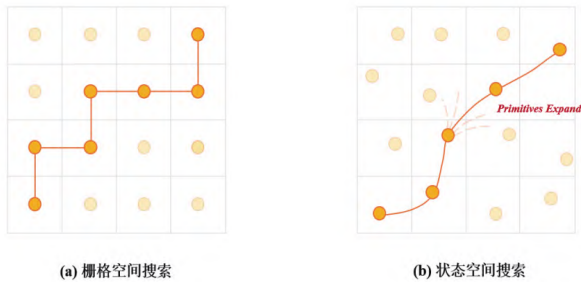


图1 栅格空间与状态空间对比

状态空间根据规划目标的运动模型, 由初始状态节点扩展生成。设直升机的状态节点 $\mathbf{x}(t) = [x(t), y(t), z(t), \dot{x}(t), \dot{y}(t), \dot{z}(t)]^T$ 。类比旋翼机的微分平坦特性^[8], 简化直升机的运动学模型, 在 x, y, z 轴表示航迹分量。同时, 考虑到直升机不同控制通道的时域响应差异, 以参数 $\lambda_x, \lambda_y, \lambda_z$ 对三轴单位控制输入量 u 的尺度变换建模该机动特性, 该过程如图2所示。

为便于状态空间生成, 以加速度控制作为直升机状态空间模型。记三轴加速度 $\ddot{x}(t), \ddot{y}(t), \ddot{z}(t)$ 对应的控制输入量为 $\mathbf{u}(t) \in \mathbb{R}^3$, 状态转移矩阵为 \mathbf{A} , 控制矩阵为 \mathbf{B} , 则直升机状态转移方程为:

$$\dot{\mathbf{x}}(t) = \mathbf{A} \cdot \mathbf{x}(t) + \mathbf{B} \cdot \mathbf{u}(t) \quad (1)$$

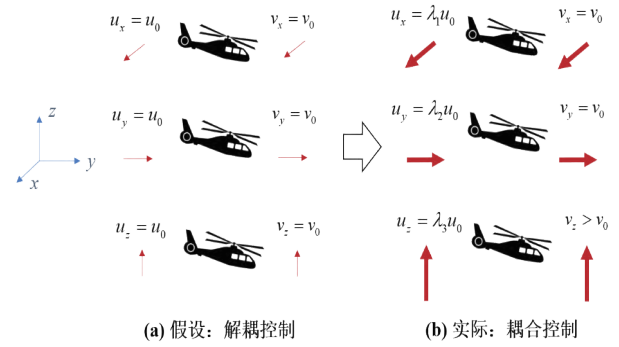


图2 直升机关联控制机动特性近似

记 \mathbf{I}_3 为 3 阶单位矩阵, $\mathbf{0}$ 为零矩阵, 则有:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_3 \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_3 \end{bmatrix} \quad (2)$$

若直升机起始状态:

$$\mathbf{x}(0) = [x(0), y(0), z(0), \dot{x}(0), \dot{y}(0), \dot{z}(0)]^T$$

则式 (1) 的通解为:

$$\mathbf{x}(t) = e^{\mathbf{A}t} \mathbf{x}(0) + \int_0^t e^{\mathbf{A}(t-\tau)} \mathbf{B} \mathbf{u}(\tau) d\tau \quad (3)$$

令三轴单位控制输入量 u_0 的可行域 $u_d = [-u_{\max}, u_{\max}]$, 并均匀划分 $2n+1$ 个离散控制输入量:

$$u_0 \in \text{discrete}(u_d) = \{-u_{\max}, -\frac{n-1}{n}u_{\max}, \dots, \frac{n-1}{n}u_{\max}, u_{\max}\} \quad (4)$$

再将 $u_0(t)$ 根据式 (5) 进行尺度变换, 体现直升机三轴控制差异。

$$\mathbf{u}(t) = \Lambda \cdot \mathbf{u}_0(t) \quad (5)$$

其中, $\mathbf{u}(t) = [u_x(t), u_y(t), u_z(t)]^T$, $\Lambda = \text{diag}(\lambda_x, \lambda_y, \lambda_z)$ 。至此, 可通过直升机起始状态 $\mathbf{x}(0)$ 、控制量可行边界 u_{\max} 以及三轴机动特性参数 $\lambda_x, \lambda_y, \lambda_z$ 根据式 (3) 扩展状态空间, 该过程简记为 Expand 规则。

1.1.2 Hybrid State A* 搜索流程

基于 Hybrid State A* 算法的前端搜索包括: (1) 状态空间扩展: 直升机运动扩展规则, 生成给定机动特性下的状态空间, 并修剪冗余节点; (2) 启发代价函数: 度量当前状态节点距离起始点和目标点的综合代价值; (3) 最优路径搜索: 依据代价最小准则, 搜索从起始点到目标点的无碰通路。

(1) 状态空间扩展

记当前待扩展状态节点为 n_c , 根据 1.1.1 中的 Expand 规则得到 N 个扩展状态节点 $\{n_i | n_0 = n_c\}$,

$i \in \{0, 1, 2, \dots, N\}$, 并将其存于 **Primitives**。为减少状态节点的冗余度, 仅保留每个栅格中距离起始点 $x(0)$ 和目标点综合代价值 f_c 最小的节点。记该修剪规则为 **Prune**, 修剪后的节点状态空间为 **nodes**。

(2) 启发式代价函数

启发式代价函数 f_c 包括机动代价 g_c 和启发代价 h_c 。其中 g_c 表征当前节点距起始点 $x(0)$ 的机动代价, 可通过转移过程中机动控制量 $u(t)$ 的时间累积度量。记采样周期为 τ , 以系数 ρ 衡量时间权重。若从起始点 $x(0)$ 到当前节点的扩展总数为 N , 则当前节点的机动代价值为^[13]:

$$g_c = \sum_{i=1}^N (\|u_i(t)\|^2 + \rho) \cdot \tau \quad (6)$$

其次, h_c 表征当前节点距目标点的启发式代价值。若当前轴向为 μ , $\mu \in \{x, y, z\}$, 状态节点的位置为 p_{uc} , 速度为 v_{uc} , 目标节点的位置为 p_{ug} , 速度为 v_{ug} 。根据 Pontryagins 最小化原理, 可通过式 (7) 估计当前点距目标点的启发代价值 h_c ^[16]。

$$\begin{cases} 1. & J^*(T) = \sum_{\mu \in \{x, y, z\}} \left(\frac{1}{3} \alpha_\mu^2 T^3 + \alpha_\mu \beta_\mu T^2 + \beta_\mu^2 T \right) \\ 2. & \begin{bmatrix} \alpha_\mu \\ \beta_\mu \end{bmatrix} = \frac{1}{T^3} \begin{bmatrix} -12 & 6T \\ 6T & -2T^2 \end{bmatrix} \cdot \begin{bmatrix} p_{ug} - p_{uc} - v_{uc}T \\ v_{ug} - v_{uc} \end{bmatrix} \\ 3. & h_c = J^*(T_0), \quad st. \frac{\partial J^*(T_0)}{\partial T_0} = 0 \end{cases} \quad (7)$$

为增强初始航迹的安全性, 记膨胀航迹点离最近障碍物的距离为 d , 安全检测阈值为 d_s , 并对 $d > d_s$ 的扩展节点施加惩罚值 h'_c 。将其归入式 (7) 3 的启发式代价值中。

$$\begin{cases} 1. & h'_c = \lambda \cdot \mathbf{I}(d - d_s) \\ 2. & h_c \leftarrow h_c + h'_c \end{cases} \quad (8)$$

最终, 将式 (6) 和式 (7) 3 的代价值相加, 即为当前节点距起始点和目标点的整体代价值 $f_c = g_c + h_c$ 。

(3) 最优路径搜索

以 **CloseList** 储存待选节点, **OpenList** 储存待扩展节点, 并将其初始化为起始点 $x(0)$ 。将上述状态空间 **nodes** 作为待检测点集。若当前遍历节点为 n_i , $n_i \notin \text{CloseList}$ 且通过碰撞安全机制 **CheckFeasible** 检测, 则根据 **EdgeCost** 规则计算当前子节点 n_i 至其父节点 n_c 的机动代价, 进而计算子节点 n_i 至起始点 $x(0)$ 的机动代价 g_{temp} 。若 $n_i \notin \text{OpenList}$, 则将其加入 **OpenList**。若 $n_i \in \text{OpenList}$ 且子节点 n_i 比其

父节点 n_c 距起始点 $x(0)$ 的机动代价更小, 则将其子节点 n_i 加入 **CloseList**。此时, 正式建立节点 n_c 与 n_i 的父子联系, 存储并计算子节点 n_i 距起点的机动代价 $g_c \leftarrow g_{temp}$ 和启发代价 h_c , 则当前节点 n_i 的整体代价值为 $f_c(n_i) \leftarrow g_c(n_i) + h_c(n_i)$ 。上述过程的伪代码如表 1 所示。

表 1 Hybrid State A* 路径规划算法

```

Algorithm 1: Hybrid State A* path planning algorithms
Input: node_start, node_goal
Output: OpenList, CloseList
1 OpenList  $\leftarrow$  node_start;
2  $f(\text{node\_start}) \leftarrow h(\text{node\_start})$ ;
3 while OpenList  $\neq \emptyset$  do
4   node_current  $\leftarrow \{ \text{node} \mid \text{node} \in \text{OpenList with min } f(\text{node}) \}$ ;
5   if node_start = node_goal then
6     | solution found break;
7   primitives  $\leftarrow \text{Expand}(\text{node\_current})$ ;
8   nodes_successor  $\leftarrow \text{Prune}(\text{primitives})$ ;
9   for node to node_successor do
10    if node not in CloseList and CheckFeasible(node) then
11       $g_{temp} \leftarrow g_c(\text{node\_current}) + \text{EdgeCost}(\text{node})$ ;
12      if node not in OpenList then
13        | Add node to OpenList;
14      else if  $g_c(\text{node}) < g_{temp}$  then
15        | continue;
16    end
17     $g_c(\text{node}) \leftarrow g_{temp}$ ;
18     $f_c(\text{node}) \leftarrow g_c(\text{node}) + h_c(\text{node})$ ;
19    Set node_current as parent of node;
20  end
21  Add node_current to CloseList;
22 end

```

1.2 增强安全管道

优化后的搜索前端虽然提升了初始航迹的机动执行性, 但受制于低精度的先验地理信息和动态战场威胁, 初始航迹仍可能与感知的新增障碍物碰撞。本节提出生成增强安全管道, 提升低空突防的动态安全性。

1.2.1 增强安全管道生成算法

增强安全管道基于初始航迹搜索结果, 通过子管道膨胀与修剪生成, 分别对应图 3 中的 (a), (b), (c)。

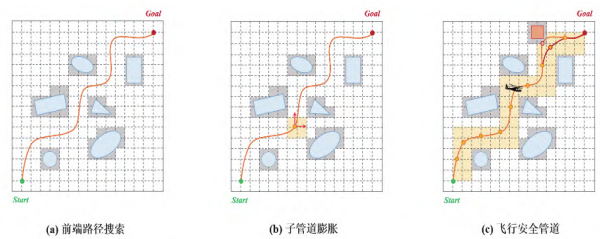


图 3 飞行安全管道生成流程

(1) 前端路径搜索

在静态全局地图中, 记障碍物点集为 O_0 , 任务起始点与目标点为 S 和 T 。前端搜索的初始航迹为 $P = \{P_i\}$, $i \in \{1, 2, \dots, N\}$, 由 N 个离散点 $P_i \in \mathbb{R}^3$ 连成, 其中 $P_1 = S$, $P_N = T$ 。

(2) 安全管道膨胀

将初始航迹 P 添加至 $OpenList$ 中, 遍历航迹点 P_i , 被访问的点将被移出 $OpenList$, 直至 $OpenList = \emptyset$ 。每个航迹点 P_i 将根据 $Expand$ 规则膨胀生成安全子管道 C_i , 对应伪代码如表 2 所示。

表 2 安全子管道膨胀

```

Algorithm 2: Safety sub-pipeline expansion
Input:  $P_j, O, \varepsilon, n$ 
Output:  $C_j$ 
1  $C_{x^+}, C_{x^-}, C_{y^+}, C_{y^-}, C_{z^+}, C_{z^-} \leftarrow P_j$ ;
2  $Safe \leftarrow True$ ;
3  $j \leftarrow 0$ ;
4 while  $Safe$  and  $j < n$  do
5   for  $\mu$  to  $x^+, x^-, y^+, y^-, z^+, z^-$  do
6      $C_\mu \leftarrow C_\mu + \varepsilon$ ;
7     if  $CheckCollide(C_\mu, O)$  then
8        $Safe \leftarrow False$ ;
9     end
10     $j \leftarrow j + 1$ ;
11 end
12 end

```

上述 $Expand$ 规则为: 记机体传感更新后的障碍物点集 O , 航迹点坐标 P_i , 膨胀步长 ε , 最大膨胀次数 n 。首先, 将安全子管道的长方体边界 $C_{x^+}, C_{x^-}, C_{y^+}, C_{y^-}, C_{z^+}, C_{z^-}$ 初始化为 P_i 对应的 x, y, z 坐标 $P_{ix}, P_{ix}, P_{iy}, P_{iy}, P_{iz}, P_{iz}$, 初始化碰撞检测变量 $Safe \leftarrow True$ 。以最大膨胀次数 n 为迭代上界, 当膨胀边界与障碍物点集通过碰撞检测 $CheckCollide$ 即 $Safe=True$ 时, 安全子管道边界持续以 ε 为步长膨胀, 即 $C_\mu \leftarrow C_\mu + \varepsilon, \mu \in \{x^+, x^-, y^+, y^-, z^+, z^-\}$, 否则膨胀结束。记膨胀后的安全子管道为 C_i , 则所有航迹点将形成膨胀管道 $C = \{C_i, i \in \{1, 2, \dots, N\}\}$ 。

(3) 膨胀管道修剪上述安全子管道可能出现图 4 所示的两种异常: (1) 子管道空间间断: 相邻航迹点 P_i, P_j 对应的子管道 C_i, C_j 不相交, 即 $C_i \cap C_j = \emptyset$ 。这将造成后端轨迹优化参数空间间断, 须引入安全管道 $\{C_i\}$ 的连续性检测, 在间断航迹点间插入补充点 P_{ij} , 重新膨胀出子管道 C_{ij} , 直至整个安全管道 $\{C_i\}$ 满足空间连续;

(2) 子管道冗余包含: 即 $C_j \subseteq C_i, i \neq j$ 。此时应剔除被包含的冗余子管道 C_j 。记该过程为 $Prune$ 规则。

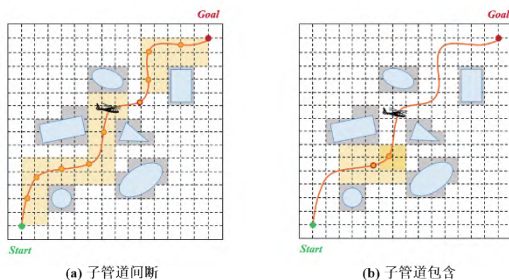


图 4 间断填补与重叠删除的异常处理

膨胀管道修剪的伪代码如表 3 所示。

表 3 安全子管道修剪

```

Algorithm 3: Safety sub-pipeline prune
Input:  $P, C, O, \varepsilon, n$ 
Output:  $C$ 
1 for  $C_i$  to  $C$  do
2   for  $C_j$  to  $C$  do
3     if  $C_j \subseteq C_i$  then
4       Remove  $C_j$  from  $C$ ;
5     else if  $C_j \cap C_i = \emptyset$  then
6       Insert  $P_{ij}$  in  $P$ ;
7        $C_{ij} \leftarrow Expand(P_{ij})$ ;
8       Insert  $C_{ij}$  in  $C$ ;
9     end
10  end
11 end

```

以 PipelineGen 规则描述增强安全管道生成, 对应伪代码如表 4 所示。最终的增强安全管道具有无碰性, 且能通过调节膨胀步长 ε 与最大膨胀次数 n 控制安全管道的膨胀速度与航迹贴合度。

表 4 增强安全管道生成

```

Algorithm 4: PipelineGen
Input:  $P, O, \varepsilon, n$ 
Output:  $C$ 
1  $j \leftarrow 0$ ;
2  $OpenList \leftarrow P$ ;
3 while  $OpenList \neq \emptyset$  do
4    $C_j \leftarrow Expand(P_j, O, \varepsilon, n)$ ;
5   Add  $C_j$  to  $C$ ;
6   Remove  $P_j$  from  $OpenList$ ;
7 end
8  $C \leftarrow Prune(C)$ 

```

1.2.2 航迹安全性约束

上述增强安全管道将作为后端航迹优化中轨迹位置参数的边界。在后端航迹优化时, 增强安全管道一般能构成凸空间, 若将待拟合航迹的 B 样条曲线控制点均限制在该凸空间中, 由于 B 样条曲线自身的凸包性质, 能保证拟合出的分段轨迹均位于该凸空间中, 从而确保了轨迹无碰的理论可行性。

若共用 $N+1$ 个控制点拟合 M 段航迹, $q_{\mu j}^i$ 表示 μ 轴方向的第 i 个控制点, 对应第 j 个分段航迹, 其中 $i \in \{x, y, z\}$ 。记 $c_{\mu j}^-$ 为当前分段航迹对应子安全管道 C_j 在 μ 轴方向上的下界, $c_{\mu j}^+$ 为上界, 则为限制优化航迹在增强安全管道中, 对所有 B 样条控制点施加约束:

$$c_{\mu j}^- < q_{\mu j}^i < c_{\mu j}^+, \quad i \in \{0, 1, \dots, N\}, \quad j \in \{0, 1, \dots, M-1\} \quad (9)$$

进一步分别建立航迹光滑度代价 f_s , 碰撞代价 f_c 以及速度与加速度代价 f_v, f_a , 并赋予相应权值 $\lambda_1, \lambda_2, \lambda_3$ 以衡量各优化准则的相对重要程度。最终, 上述各项加权和构成后端优化总代价^[15]。

$$f = \lambda_1 f_s + \lambda_2 f_c + \lambda_3 (f_v + f_a) \quad (10)$$

代价值 f_s 依据轨迹的弹性模型实现^[17]。相邻控制点间可视为存在与间距成正比的弹力,若B样条曲线的相邻控制点 \mathbf{Q}_{i-1} , \mathbf{Q}_i , \mathbf{Q}_{i+1} 呈均匀分布,即 \mathbf{Q}_i 到 \mathbf{Q}_{i-1} , \mathbf{Q}_{i+1} 有相近的距离,则此时优化航迹趋于“二力平衡”,具有较好光滑性,规避了直升机的剧烈机动。代价值 f_s 见式(11)。

$$f_s = \sum_{i=K-1}^{N-K+1} \|(\mathbf{Q}_{i+1} - \mathbf{Q}_i) - (\mathbf{Q}_i - \mathbf{Q}_{i-1})\|^2 \quad (11)$$

代价值 f_c , f_v , f_a 则通过距离阈值 d_{thr} 、速度阈值 v_{max} 、加速度阈值 a_{max} ,构建具有超调惩罚特性的虚拟势能模型^[13]。当控制点距离障碍物过近,或生成轨迹上的速度与加速度过大时,相应的虚拟势能值增大,进而增加安全距离、速度与加速度的惩罚值。三种虚拟势能模型见式(12)。

$$\left\{ \begin{array}{l} 1. \quad f_v = \sum_{\mu \in \{x, y, z\}} \sum_{i=K}^{N-K} F_v(V_{\mu}) \quad F_v(v_{\mu}) = \begin{cases} (v_{\mu}^2 - v_{max}^2)^2, & v_{\mu}^2 > v_{max}^2 \\ 0, & v_{\mu}^2 \leq v_{max}^2 \end{cases} \\ 2. \quad f_a = \sum_{\mu \in \{x, y, z\}} \sum_{i=K}^{N-K} F_a(A_{\mu}) \quad F_a(a_{\mu}) = \begin{cases} (a_{\mu}^2 - a_{max}^2)^2, & a_{\mu}^2 > a_{max}^2 \\ 0, & a_{\mu}^2 \leq a_{max}^2 \end{cases} \\ 3. \quad f_c = \sum_{i=K}^{N-K} F_c(d(\mathbf{Q}_i)) \quad F_c(d(\mathbf{Q}_i)) = \begin{cases} (d(\mathbf{Q}_i) - d_{thr})^2, & d(\mathbf{Q}_i) \leq d_{thr} \\ 0, & d(\mathbf{Q}_i) > d_{thr} \end{cases} \end{array} \right. \quad (12)$$

其中 $d(\mathbf{Q}_i)$ 为控制点 \mathbf{Q}_i 离最近障碍物的距离。至此,通过在“前端”与“后端”之间引入增强安全管道并对拟合航迹的B样条控制点施加安全性约

束的方式,确保了更新环境下优化航迹的动态安全性。

2 实验结果讨论

为论证本文提出的Hybrid State A*与增强安全管道对规划航迹机动性与安全性的提升,本小节将进行算法对比、航迹运动特性与超参数仿真实验。其中,全局地图的尺寸为 80×80 m,栅格分辨率为0.2 m,采样周期为0.002 s。

仿真实验所采用的计算机硬件处理器为Inter(R) Core(TM) i5-7200U CPU @ 2.50GHz,内存为16G。在Ubuntu18.04系统中,基于ROS Melodic机器人仿真框架与Rviz数据可视化工具进行仿真验证。

2.1 轨迹机动性优化验证

本节将验证改进前端的规划系统,在航迹搜索性能、动力学约束与机动特性方面的提升。

2.1.1 Hybrid State A*与A*前端搜索对比实验

本实验分别以Hybrid State A*和A*算法为搜索前端,保持环境威胁动态更新、任务起始与目标点不变,记录N次蒙特卡罗实验中两种算法的规划航迹长度 l ,规划用时 t 与 $rate$ 成功率如表5所示。

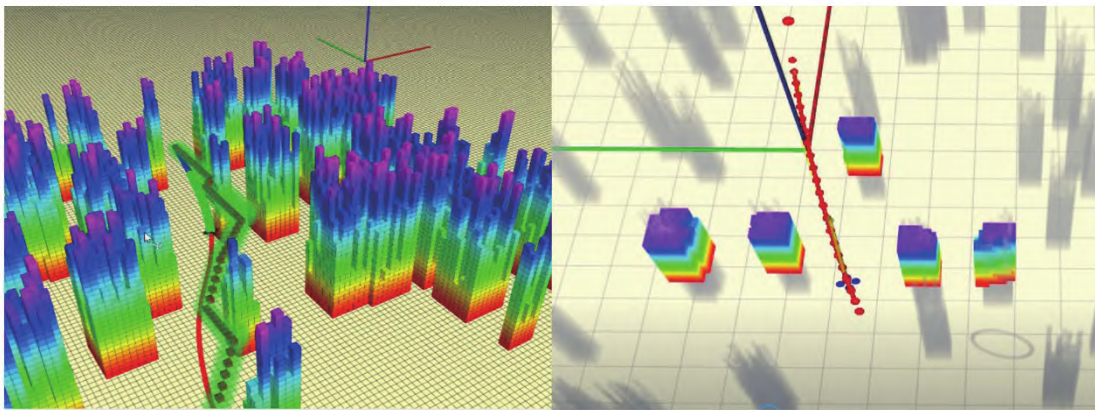


图5 基于Hybrid State A*与A*前端的航迹规划

表5 Hybrid State A*与A*的航迹规划对比

N\指标	A*			Hybrid State A*		
	l(m)	t(s)	rate(%)	l(m)	t(s)	rate(%)
20	42.5210	6.219E-04	60	40.1972	5.446E-04	77
50	39.9049	7.300E-04	67	39.5469	5.062E-04	74

在表5统计的多次仿真中, 基于 Hybrid State A* 算法的规划航迹, 具有更小的移动距离与规划用时, 且分别提升了 17% 和 7% 的规划成功率。

2.1.2 Hybrid State A* 对航迹的动力学约束验证

为验证引入动力学约束的 Hybrid State A* 算

法能对规划航迹施加有效机动限制, 在状态空间扩展时, 对速度和加速度分别施加 $v_{\max}=20\text{m/s}$, $a_{\max}=15\text{m/s}^2$ 的约束, 并记录 3 次仿真中规划航迹的速度和加速度曲线, 分别如图 6 和图 7 所示。

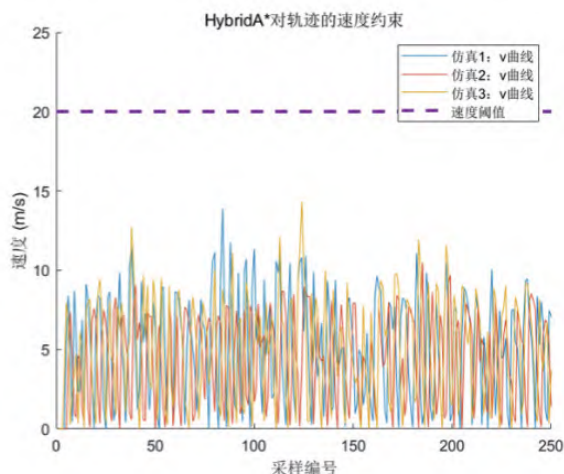
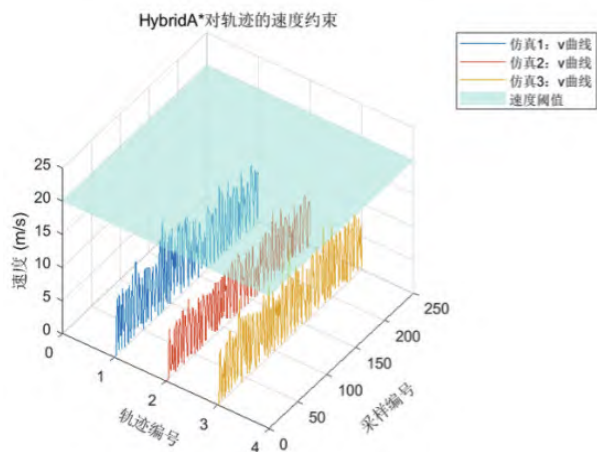


图 6 规划航迹速度约束效果

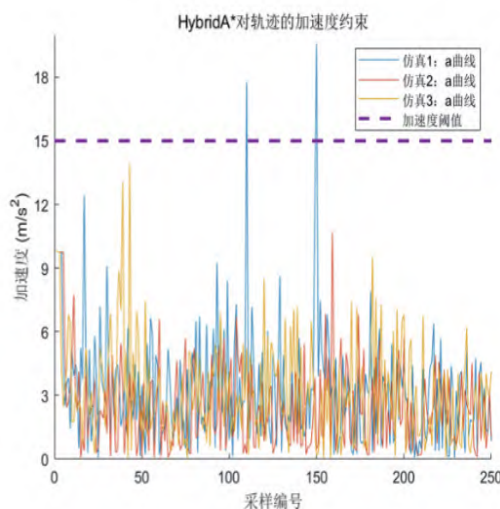
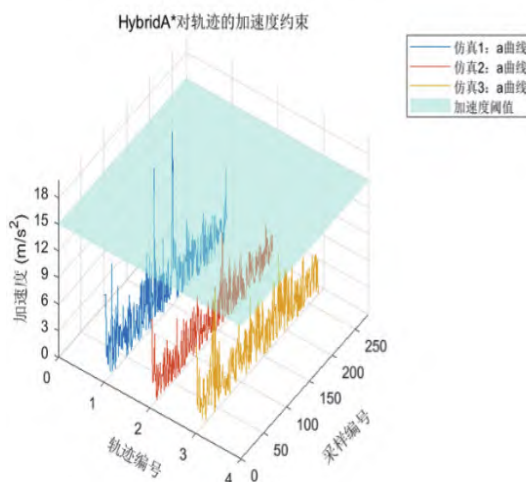


图 7 规划航迹加速度约束效果

尽管仿真中随机生成的局部苛刻地形, 使得后端优化时分段航迹时间分配较短, 导致个别时刻速度与加速度产生超调, 但航迹整体状态被限制在速度和加速度阈值以内。

2.1.3 Hybrid State A* 对航迹的机动适应性验证

为验证控制量尺度变换对规划航迹机动适应性

的提升, 分别设置不考虑三轴控制差异的对照组: $\lambda_1=1$, $\lambda_2=1$, $\lambda_3=1$, 与考虑控制差异的实验组: $\lambda_1=1$, $\lambda_2=1$, $\lambda_3=0.5$ 。分别记录规划航迹在三轴的速度与加速度曲线, 分别如图 8 和图 9 所示; 与加速度的速度的大小关系, 如表 6 所示。

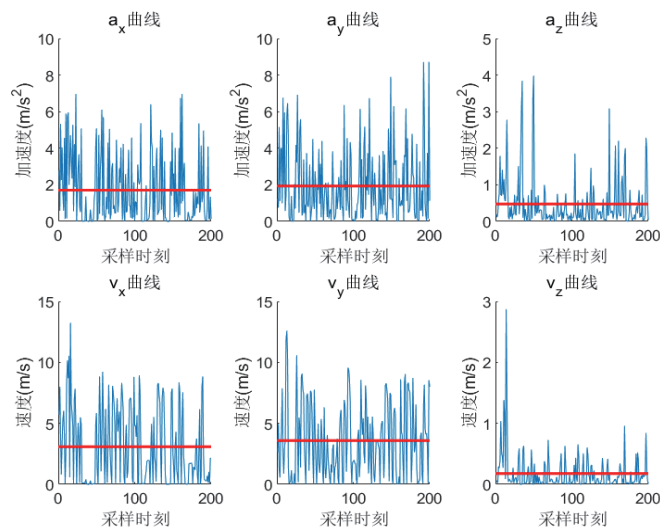


图 8 对照组三轴运动状态

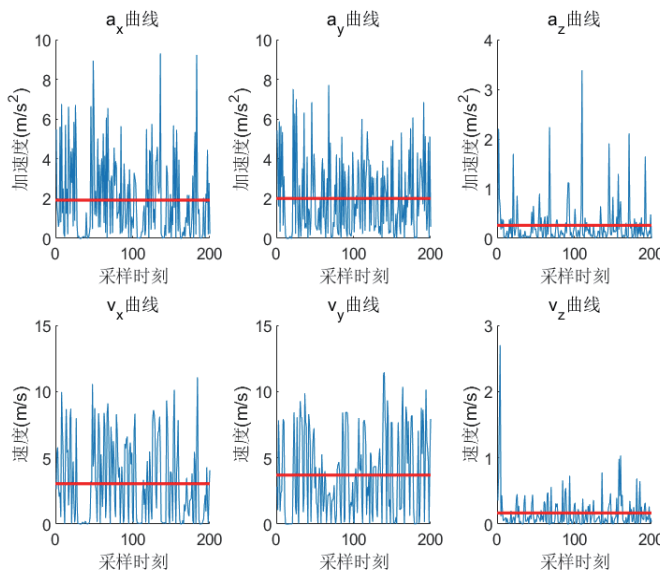


图 9 实验组三轴运动状态

表 6 加速度和速度约束效果

状态量	a_x	a_y	a_z	v_x	v_y	v_z
对照组均值	1.7081	1.9334	0.4753	3.0803	3.5871	0.1767
实验组均值	1.7857	2.008	0.2617	3.0495	3.6974	0.1669
占比 (%)	104.54	103.88	55.06	98.99	103.07	94.44

注：a 单位 (m/s²)，v 单位 (m/s)

理论上，实验组规划航迹加速度 a_z 的均值应为对照组的一半，其余分量保持不变。根据表 6，实际规划航迹的运动分量特性与理论相符，说明改进

算法能通过尺度参数的灵活调整，使规划航迹符合直升机的机动特性。

2.2 轨迹安全性优化验证

为验证增强安全管道能在动态威胁的环境下提升航迹规划成功率, 进行安全管道对比与超参数实验。

2.2.1 安全管道对比实验

为验证增强安全管道对规划航迹安全性的提升, 对前端改进的航迹规划系统引入增强安全管道, 并对比规划成功率。仿真效果如图 10 所示, 其中灰色区域为增强安全管道。参数设置为: 膨胀步长 $\epsilon=1$ (单位与栅格分辨率相同), 最大膨胀次数 $n=4$ 。

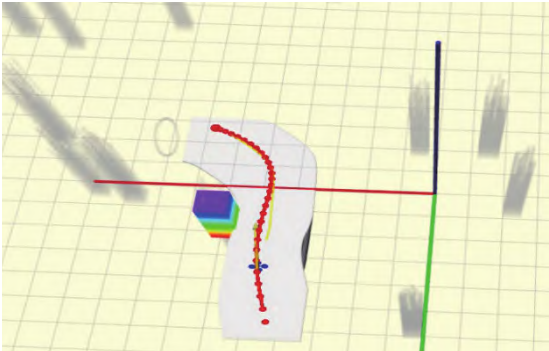


图 10 引入增强安全管道的航迹规划

表 7 增强安全管道对比

N 指标	Hybrid State A* 无安全管道			Hybrid State A* 有安全管道		
	l(m)	t(s)	rate(%)	l(m)	t(s)	rate(%)
20	39.9049	5.06E-04	74	40.0278	11.64E-03	96
50	45.2885	5.15E-04	79	45.3842	12.13E-03	95

在表 7 统计的多次仿真中, 尽管引入增强安全管道后的规划用时略有增加, 但规划成功率分别提升了 22% 和 16%, 验证了其对航迹安全性的提升。

2.2.2 安全阈值对航迹规划的影响

为验证安全阈值对规划航迹的影响, 进行超参数实验。根据仿真中障碍物的稠密情况, 分别设置安全阈值 $d_{thr}=1,1.5,2\text{m}$, 并记录实验结果, 如表 8 所示。

表 8 安全阈值超参数测试

安全阈值 \ 指标	l(m)	t(s)	rate(%)
1	45.3248	1.244E-02	90
1.5	45.4003	1.331E-02	92
2	45.3842	1.213E-02	95

较大的安全阈值增加了规划航迹与障碍物边界的安全裕度, 提升了规划航迹的安全性, 同时膨胀至障碍物边界所需的迭代次数减少, 规划用时减少。真实场景中可根据地形情况, 灵活调节安全阈值, 实现安全而高效的航迹规划。

3 结论

本文针对传统“前端 - 后端”式航迹规划算法, 所面临的直升机机动约束和动态环境威胁两个挑战, 首先利用引入机动约束的 Hybrid State A* 算法, 提

升初始搜索航迹的可执行性, 其次通过增强安全管道, 将优化航迹限制在安全区域内, 提升了航迹的安全性。最终, 通过改进算法的对比与超参数调节实验, 验证了基于 Hybrid State A* 与增强安全管道的直升机航迹规划算法, 对规划航迹机动性与安全性的提升。未来可建立更精确的直升机动力学扩展建模, 采用多面立方体优化安全管道表征, 对该算法做进一步提升。

参 考 文 献

[1] 邹凛浩, 林有希, 蔡建国. 移动机器人运动规划算法研究进展 [J]. 机床与液压, 2021, 49(23): 154-158.

[2] 王维平, 刘娟. 无人飞行器航迹规划方法综述 [J]. 飞行力学, 2010, 28(02): 6-10+15.

[3] Dijkstra, E.W. A note on two problems in connexion with graphs[J]. Numer. Math. 1, 269–271 (1959).

[4] P. E. Hart, N. J. Nilsson and B. Raphael. A formal basis for the heuristic determination of minimum cost paths[J]. IEEE Transactions on Systems Science and Cybernetics, 1968, 4(2): 100-107.

[5] De Filippis, L., Guglieri, G., Quagliotti, F. Path planning strategies for uavs in 3d environments[J]. Journal of Intelligent & Robotic Systems, 2012, 65: 247–264.

[6] 任天祥, 贺建良, 邹杰. 基于改进 Theta* 算法的直升机

三维航路规划 [J]. 电光与控制, 2022, 29(02): 45-48.

[7] 王锦博, 宋伟, 尚帅, 兰庭信, 盛守照. 基于改进粒子群算法的无人直升机航路规划 [J]. 机械与电子, 2021, 39(06): 66-69.

[8] Pivtoraiko M, Kelly A. Kinodynamic motion planning with state lattice motion primitives[C]//2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2011: 2172-2179.

[9] Dolgov D, Thrun S, Montemerlo M, et al. Practical search techniques in path planning for autonomous driving[J]. Ann Arbor, 2008, 1001(48105): 18-80.

[10] Mellinger D, Kumar V. Minimum snap trajectory generation and control for quadrotors[C]//2011 IEEE international conference on robotics and automation. IEEE, 2011: 2520-2525.

[11] 周炜, 王小平, 孙浩水, 陈勇. 基于贝赛尔曲线的四旋翼无人机轨迹优化 [J]. 电子测量与仪器学报, 2019, 33(10): 53-58.

[12] 李钦, 李锋, 杨振发, 龙际梦. 面向直升机低空突防的三维航线规划算法研究 [J]. 测绘工程, 2015, 24(07): 34-38.

[13] Liu S., Watterson, M., Mohta, K., Sun, K., Bhattacharya, S., Taylor, C. J., & Kumar, V. Planning dynamically feasible trajectories for quadrotors using safe flight corridors in 3-d complex environments[J]. IEEE Robotics and Automation Letters, 2017, 2(3): 1688-1695.

[14] 柳皓. 基于安全飞行走廊的四旋翼无人机编队避障系统 [D]. 四川: 电子科技大学, 2021.

[15] Zhou B, Gao F, Wang L, et al. Robust and efficient quadrotor trajectory generation for fast autonomous flight[J]. IEEE Robotics and Automation Letters, 2019, 4(4): 3529-3536.

[16] Mueller M W, Hehn M, D' Andrea R. A computationally efficient motion primitive for quadrocopter trajectory generation[J]. IEEE transactions on robotics, 2015, 31(6): 1294-1310.

[17] Quinlan S, Khatib O. Elastic bands: Connecting path planning and control[C]//Proceedings IEEE International Conference on Robotics and Automation. IEEE, 1993: 802-807.