

DOI: 10.12382/bgxb.2024.1146



# 基于强化学习冲突消解的大规模无人机集群航迹规划方法

周桢林<sup>1,2</sup>, 龙腾<sup>1,2,3,4</sup>, 刘大卫<sup>5</sup>, 孙景亮<sup>1,2,3\*</sup>, 钟建鑫<sup>1,2</sup>, 李俊志<sup>1,2</sup>

(1. 北京理工大学 宇航学院, 北京 100081; 2. 飞行器动力学与控制教育部重点实验室, 北京 100081;

3. 北京理工大学重庆创新中心, 重庆 401121; 4. 陆空基信息感知与控制全国重点实验室, 北京 100081;

5. 中国兵器科学研究院, 北京 100089)

**摘要:** 面向大规模无人机集群协同作业场景, 针对航迹冲突频繁导致集群航迹规划高耗时的问题, 开展基于强化学习冲突消解的大规模无人机集群航迹规划方法研究。构建“顶层冲突消解-底层航迹规划”的双层规划架构, 降低航迹冲突的时空维度。在顶层冲突消解层, 设计基于 Rainbow DQN (Deep Q-Networks algorithm) 训练框架的冲突消解策略网络, 将每个航迹冲突的消解过程转换为二叉树拓展左、右树节点的动作选择过程, 实现不同冲突消解顺序与冲突消解结果的映射, 减少树节点的遍历, 提高冲突消解效率; 在底层航迹规划层, 将时间维度引入空间避碰策略, 提出基于节点重扩展机制的跳点搜索法(Re-planning Jump Point Search, ReJPS), 增加规划可行域, 提升航迹冲突的消解能力。仿真结果表明: 相比基于 CBS (Conflict Based Search) + A\* 方法与 CBS + ReJPS 航迹规划方法, 新方法在最优性相当的前提下, 平均规划耗时分别降低了 86.64% 和 19.65%。

**关键词:** 无人机集群; 航迹规划; 深度强化学习; 冲突搜索方法; 冲突消解

中图分类号: V249

文献标志码: A

文章编号: 1000-1093(2025)05-241146-12

## Path Planning Method for Large-scale UAV Swarms Based on Reinforcement Learning Conflict Resolution

ZHOU Zhenlin<sup>1,2</sup>, LONG Teng<sup>1,2,3,4</sup>, LIU Dawei<sup>5</sup>, SUN Jingliang<sup>1,2,3\*</sup>, ZHONG Jianxin<sup>1,2</sup>, LI Junzhi<sup>1,2</sup>

(1. School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, China;

2. Key Laboratory of Dynamics and Control of Flight Vehicle of Ministry of Education, Beijing 100081, China;

3. Beijing Institute of Technology Chongqing Innovation Center, Chongqing, 401121, China;

4. National Key Laboratory of Land and Air Based Information Perception and Control, Beijing 100081, China;

5. Research and Development Academy of Machinery Equipment, Beijing 100089, China)

**Abstract:** In the context of large-scale unmanned aerial vehicle (UAV) swarm cooperative flight scenarios, the high computational time consumption in swarm path planning is caused by frequent path conflicts. Aiming at the problem above, a large-scale UAV swarm path planning method based on reinforcement learning conflict resolution is developed. A dual-layer planning architecture, comprising a high-level layer of conflict resolution and a low-level layer of path planning, is constructed to reduce the spatial and temporal dimensions of path conflicts. At the high-level layer of conflict resolution, a conflict

收稿日期: 2024-12-24

基金项目: 国家杰出青年科学基金项目(52425211); 北京理工大学青年教师学术启动计划项目(XSQD-202201005)

\*通信作者邮箱: sunjingliangac@163.com

resolution strategy network based on the Rainbow deep Q-networks (DQN) algorithm training framework is designed. This network transforms the resolution process of each path conflict into the action selection process of left and right tree nodes of a binary tree. This approach maps different conflict resolution sequences to their outcomes, thereby reducing the traversal of tree nodes and improving the efficiency of conflict resolution. At the low-level layer of path planning, the time dimension is incorporated into the spatial collision avoidance strategy. A re-planning jump point search (ReJPS) method based on a node re-expansion mechanism is proposed, which increases the feasible planning domain and enhances the ability to resolve the path conflicts. Simulated results indicate that, compared to the path planning methods based on the conflict-based search (CBS) + A<sup>\*</sup> and CBS + ReJPS, the proposed method reduces the average planning time by 86.64% and 19.65%, respectively, while maintaining comparable optimality.

**Keywords:** unmanned aerial vehicle swarm; path planning; deep reinforcement learning; conflict-based search; conflict resolution

## 0 引言

随着无人机技术的迅猛发展,其在军事物资运输<sup>[1]</sup>、农业监测<sup>[2]</sup>和灾害响应<sup>[3]</sup>等军民领域<sup>[4-5]</sup>的应用日益广泛。尤其在前线物资自主配送、精准农业施肥、突发紧急救援等复杂任务场景中,无人机集群因其独有的高效性和灵活性而备受青睐。然而,在楼宇林立的城市环境或野外树林等受限空域中,随着集群规模的不断增大,其协同作业时的航迹冲突概率呈指数级增长。此外,任务按需更新、动态障碍规避等任务约束进一步加剧了无人机集群规划航迹的冲突概率,导致规划耗时增加<sup>[6-7]</sup>。因此,如何在受限空域中降低大规模无人机集群规划航迹冲突概率、提升规划时效性,是实现集群高效、安全航迹规划的技术瓶颈。

冲突消解作为保证集群安全的关键技术,已经受到国内外学者的广泛关注<sup>[8-10]</sup>。以计算复杂性分类,冲突消解技术可以划分为耦合式<sup>[11]</sup>与解耦式<sup>[12]</sup>。相较于耦合式,解耦冲突消解通过问题分解将复杂规划转化为关联子问题求解,具有效率高、可扩展性强的优势,更适用于大规模集群场景。

解耦式冲突消解根据规划架构分类主要包含优先级规划<sup>[13]</sup>和反应式规划<sup>[14]</sup>。优先级规划基于序列规划思想,为无人机分配优先级来消解航迹冲突,例如 EPEA<sup>\*</sup> (Enhanced Prioritized Planning A<sup>\*</sup>) 算法。但是本质上严重依赖优先级的选择,优先级的不当设置将增加航迹的冲突概率,甚至陷入局部死锁,导致航迹规划失败。对比之下,反应式规划的代表算法——基于冲突搜索(Conflict Based Search, CBS)的算法<sup>[15]</sup>采用分层规划策略,在不考虑无人

机之间碰撞约束的前提下,通过底层寻径方法规划每架无人机的最短轨迹,并根据冲突检测结果建立冲突约束树;在此基础上,由顶层规划消解集群内的航迹冲突,实现集群高效可靠规划。因其具备良好的灵活性与兼容性,得到了相关学者的广泛关注与研究,取得了较好的研究成果<sup>[16]</sup>。为进一步提升航迹规划的冲突消解能力,于连波等<sup>[8]</sup>在 CBS 算法的基础上设计了多目标代价函数,并结合速度避障法提升了突发冲突消解的能力;Ren 等<sup>[17]</sup>开发了一种基于冲突的多目标搜索(Multi-Objective Conflict Based Search, MO-CBS)算法,并从理论上证明了该方法能够找到所有成本唯一的帕累托最优解。

但是上述方法主要面向目标函数最优性提升,集群规模较小,难以满足大规模集群协同航迹规划的冲突消解与高效规划能力。Boyarski 等<sup>[18]</sup>引入优先级调整策略,提出了增强的 ICBS (Improved Conflict Based Search) 算法,提升了树扩展过程的效率。Sharon 等<sup>[19]</sup>提出了 MA-CBS (Meta-Agent Conflict Based Search) 算法,在分割策略上引入了元合并的思想,等价削减了约束树的搜索维度。文献[20]训练得到了一个基于冲突选择的策略网络,提升了冲突搜索的效能。上述方法面向改善冲突的消解顺序,通过规则式的策略机制定制冲突消解的优先级,在众多冲突中选择一对冲突进行分割。虽然在一定程度上提升了算法的效率,但是忽略了协同航迹规划的最优性,且随着集群规模的提升易陷入局部死锁。

学习类方法尤其是深度强化学习(Deep Reinforcement Learning, DRL),能够通过与环境的大量交互逐步优化策略,从而弥补反应式规划受规则掣

肘的不足。因此,亟需通过将反应式规划与学习类方法相结合,通过设计合适的网络结构与奖励机制,加快冲突消解策略网络的训练速度,进而提升集群航迹冲突的消解能力。此外,上述冲突消解方法在冲突消解策略的基础上均采用跳点搜索(Jump Point Search, JPS)法<sup>[21]</sup>、快速扩展随机树(Rapidly-expanding Random Tree, RRT)法<sup>[22]</sup>、人工势场(Artificial Potential Field, APF)法<sup>[23]</sup>等方法进行航迹规划,并未考虑冲突发生时序对冲突消解的影响,在一定程度上缩减了集群航迹规划可行域。

综上所述,本文针对大规模无人机集群在受限空域下的协同航迹规划问题,构建“顶层冲突消解-底层航迹规划”的双层规划架构;引入DRL技术,实现不同冲突消解顺序与冲突消解结果的映射,提升了冲突消解效率;提出了基于节点重扩展机制的跳点搜索法(Re-planning Jump Point Search, ReJPS),提升了航迹冲突的消解能力。

本文的主要研究贡献如下:

1)提出了一种DRL增强的无人机集群航迹规划冲突消解方法,设计了基于Rainbow DQN(Deep Q-Networks algorithm)训练框架的冲突消解策略网络,将每个航迹冲突的消解过程转换为二叉树拓展左、右树节点的动作选择,以实现不同冲突消解顺序与冲突消解结果的映射,减少树节点的遍历,提高冲突消解效率。

2)在传统JPS算法中障碍信息仅考虑空间位置的基础上,将时间维度引入障碍信息,增加规划可行域;在多机冲突消解的过程中设计节点重扩展机制,保留了节点位置相同但节点代价、回溯航迹不同的节点信息,为航迹冲突的消解提供了更多选择,提升了受限空域下大规模集群航迹规划的时效性。

## 1 问题描述

作为多智能体航迹规划的一种特殊形式,无人机集群航迹规划问题可以近似转化为:在包含 $n$ 个禁飞区 $\mathcal{O} = \{o_1, \dots, o_n\}$ 的连续多维空域 $W \subseteq \mathbb{R}^\sigma$ , $\sigma = 2, 3$ 中,航迹规划器为 $N$ 架无人机组成的集群 $U = \{u_1, \dots, u_N\}$ 规划航迹集合 $\wp = \{\xi_1, \xi_2, \dots, \xi_N\}$ ,在没有航迹冲突的前提下,尽可能高效地完成既定任务。

### 1.1 棚格地图

为了更直观地在空间内描述航迹冲突问题,本文将连续三维空域 $W \subseteq \mathbb{R}^3$ 离散化为 $m \times m$ 个单位栅格组成的棚格地图 $G$ 。航迹规划器基于 $G$ 中的态

势信息变化为 $N$ 架无人机规划无冲突航迹,具体定义如下:

$$\begin{cases} G = G_0 + G_N + G_S + G_E \\ G_0 = \{g_{xy}^{pro} | pro = 0\}_{m \times m} \\ G_N = \{g_{xy}^{pro} | pro = 1\}_{m \times m} \\ G_S = \{g_{xy}^{pro} | pro = 2\}_{m \times m} \\ G_E = \{g_{xy}^{pro} | pro = 3\}_{m \times m} \end{cases} \quad (1)$$

式中: $g_{xy}^{pro}$ 表示一个 $1 \times 1$ 的单位栅格(单位栅格的实际大小根据实际空域与任务需要定义), $x, y$ 分别表示单位栅格在笛卡尔坐标系中的坐标位置; $pro$ 表示单位栅格的当前状态信息,用来区分障碍威胁位置 $G_0$ (见图1中黑色栅格)、安全位置 $G_N$ (见图1中白色栅格)、起点位置 $G_S$ (见图1中绿色栅格)和终点位置 $G_E$ (见图1中红色栅格)。

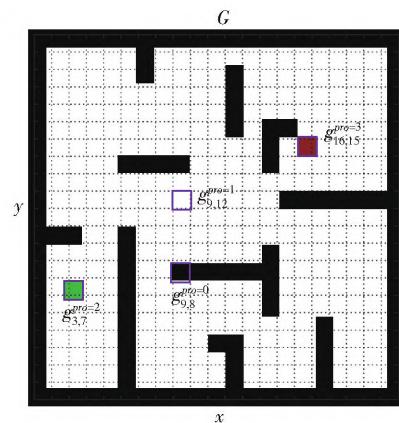


图1 棚格地图

Fig. 1 Grid-based map

### 1.2 航迹规划模型

通过将规划时间离散化为时间序列 $T = \{T_0, T_1, T_2, \dots\}$ ,无人机 $u_i$ 的航迹 $\xi_i$ 被转化为

$$\xi_i = \{p_i^{T_0}, p_i^{T_1}, \dots, p_i^{T_k}\} \quad (2)$$

$$p_i^{T_k} = \langle x_i, y_i, u_i, T_k \rangle \quad (3)$$

式中: $p_i^{T_k}$ 为无人机 $u_i$ 在 $T_k$ 时刻占用的栅格单元; $x_i, y_i$ 分别表示该栅格单元在笛卡尔坐标系中的位置。

### 1.3 航迹冲突模型

无人机的飞行安全是集群航迹规划问题的重要约束条件。考虑到禁飞区域以及邻域飞行器所占空域,无人机集群的有效航迹需要规避上述区域,即

$$p_i^{T_k} \in \{g_{xy}^{pro} | pro > 0\} \quad (4)$$

考虑集群内部避碰需求,任意两架无人机 $u_i, u_j$ 在邻域时间尺度 $\Delta t_a$ 内不能处于同一栅格,即

$$\forall u_i, u_j \in U, p_i^{T_{k_1}} \in \xi_i, p_j^{T_{k_2}} \in \xi_j \\ x_i \neq x_j \parallel y_i \neq y_j \parallel |T_{k_1}^i - T_{k_2}^j| \leq \Delta t_a \quad (5)$$

式中:  $U$  为无人机组成的集群。

一旦两架无人机违背了式(5)的冲突约束, 记此时发生的航迹冲突  $c_{ij}$  为

$$c_{ij} = \langle i, j, x_i, y_i, T_{k_1}^i, T_{k_2}^j \rangle \quad (6)$$

## 2 DRL 增强的冲突搜索方法

无人机集群航迹规划问题本质上就是在搜索各架无人机可行航迹的同时, 不断检测、管理并消解航迹冲突。基于冲突搜索的算法<sup>[15]</sup>将航迹规划过程解耦为双层搜索结构。在顶层中初始化构建并维护一个二叉约束树  $\tau$ , 树节点  $Node$  包含集群航迹冲突的约束信息  $Con$ 、航迹集合  $\wp$  和集群代价  $Cost$ , 二叉约束树选择集群代价最低的树节点, 通过 A\*、JPS 等底层规划提供满足约束信息的航迹, 消解两机之间的航迹冲突并拓展左、右树节点, 直至二叉约束树中首次出现包含无冲突航迹集合的树节点。

如图 2 所示, 虽然基于冲突搜索的算法能够在保证最优性的同时处理多无人机航迹规划问题, 但是航迹冲突数量将会增加二叉约束树的深度, 导致顶层规划搜索集群无冲突航迹的难度提高; 同时二叉约束树的深度直接增加了树节点约束信息的维度, 减小了单次底层规划的可行解空间, 延缓了拓展树节点的速度, 从而增加了整体算法的收敛时间。

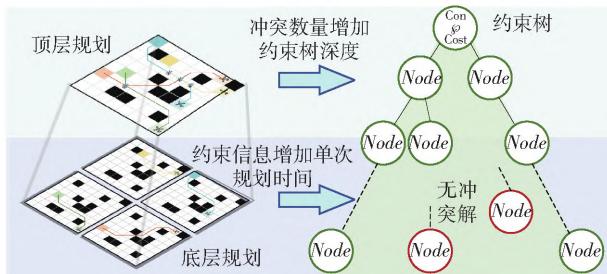


图 2 基于冲突搜索的算法的逻辑示意图

Fig. 2 Logical diagram of hierarchical search structure

对于大规模无人机集群航迹规划问题, 相比于各机最优性考量(航程最短、能量最优等性能指标), 安全性才是集群规划的首要目标。换言之, 快速找到无航迹冲突的树节点即可满足大规模无人机集群实时规划的工程需求, 而满足此要求的树节点往往并不唯一。

树节点的集群代价  $Cost$  作为树扩展的唯一判据, 直接影响二叉约束树搜索到无冲突集群航迹解时的节点数量, 航迹冲突集合  $C$  中子冲突  $c_{ij}$  的消

解优先级更是驱动了二叉约束树扩展节点的次序。因此在找到最终的包含无冲突航迹集合的树节点之前, 搜索计算的树节点个数越少, 算法时效性越好。

### 2.1 DRL 增强的航迹冲突消解策略

为解决受限空域下大规模无人机集群航迹冲突消解难导致的规划高耗时问题, 本文提出一种 DRL 增强的航迹冲突消解策略 (Deep Reinforcement Learning Enhanced Conflict Based Search, DRLCBS), 以无人机集群航迹集合中最少航迹冲突为优化目标, 融合了双深度 Q 网络、多步学习、优先经验回放机制以及噪声网络等改进方法<sup>[24]</sup>, 对航迹冲突消解二叉树模型进行优化。根据马尔可夫决策过程的要求, 设计与航迹冲突消解模型对应的状态空间、动作空间、奖励函数以及网络与参数。

#### 2.1.1 状态空间的定义描述

精简的状态空间设计不仅能准确描述无人机集群面临不同环境的状态信息, 还能降低 Rainbow DQN 模型的训练复杂程度, 提升冲突消解策略的学习效率。如图 3 所示, 本文在  $m \times m$  的栅格地图  $G$  的基础上, 定义如下状态空间  $s_t$ :

$$s_t = \{s_{map}, s_{grid}, s_{time}, s_{conflict}\} \quad (7)$$

式中:  $s_{map}$  为地图状态信息;  $s_{grid}$  为栅格状态信息;  $s_{time}$  为时间状态信息;  $s_{conflict}$  为冲突状态信息。

状态空间所包含的 4 个状态信息具体定义如下:

##### 1) 地图状态信息 $s_{map}$

用障碍威胁信息  $G_O = \{g_{xy}^{pro} | pro = 1\}_{m \times m}$  和无信息位置  $G_N = \{g_{xy}^{pro} | pro = 0\}_{m \times m}$  共同描述地图状态。

$$\begin{cases} s_{map} = \{s_{map}^{xy}\} \\ s_{map}^{xy} = \{g_{xy}^{pro} | pro = 0, 1\} \end{cases} \quad (8)$$

##### 2) 栅格状态信息 $s_{grid}$

栅格状态信息用来描述无人机集群在地图环境中的拥挤程度。地图  $G$  中的任意栅格单元  $g_{xy}$  一旦处于一个无人机  $u_i$  的飞行航迹  $\xi_i = \{p_i^1, p_i^2, \dots, p_i^k\}$ ,  $k \in \mathbb{Z}^+$  上, 则  $g_{xy}$  的拥挤程度  $C_g^{xy} = C_g^{xy} + 1$ 。

$$\begin{cases} s_{grid} = \{s_{grid}^{xy}\} \\ s_{grid}^{xy} = \{C_g^{xy}\} \end{cases} \quad (9)$$

##### 3) 时间状态信息 $s_{time}$

任意栅格单元  $g_{xy}$  上无人机集群到达的期望时间总和  $T_{total}^{xy}$  作为冲突发生时间的重要参考指标, 直接反映了冲突发生的优先级顺序, 也是影响冲突消解次序的主要因素。为了提高模型的泛化能力, 归

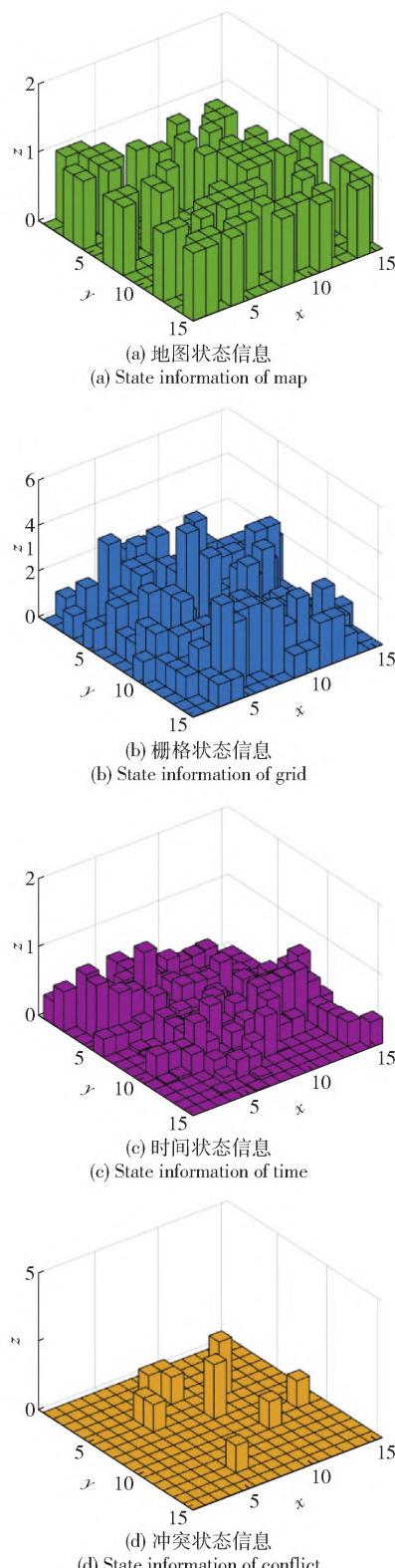


图3 无人机集群状态空间

Fig. 3 State space of UAV swarms

一化处理后时间状态信息为[0,1]的连续空间。

$$\begin{cases} s_{\text{time}} = [s_{\text{time}}^{\text{xy}}] \\ s_{\text{time}}^{\text{xy}} = T_{\text{total}}^{\text{xy}} / \max \{T_{\text{total}}^{\text{xy}}\} \end{cases} \quad (10)$$

#### 4) 冲突状态信息 $s_{\text{conflict}}$

由于约束树的结构为二叉树,单步动作空间仅解决两机之间的一个冲突,因此定义冲突状态信息如式(10)所示,以更好地表征栅格单元  $g_{xy}$  中存在的冲突个数  $N_{\text{conflict}}$ ,并以此预示冲突消解的紧迫程度。

$$\begin{cases} s_{\text{conflict}} = \{s_{\text{conflict}}^{\text{xy}}\} \\ s_{\text{conflict}}^{\text{xy}} = N_{\text{conflict}} \end{cases} \quad (11)$$

#### 2.1.2 动作空间的定义描述

基于冲突搜索的算法将集群航迹规划转化为一个约束二叉树结构中高效找到无航迹冲突的树节点的问题,因此对于二叉树的冲突消解选择,动作空间  $a_t$  为扩展左子节点和右子节点,用 0-1 空间表示。对应要消解的航迹冲突  $c_{ij} = (i, j, x_i, y_i, T_{k_1}^i, T_{k_2}^i)$ , 左节点表示无人机  $u_j$  在  $T_{k_2}^i \pm \Delta t_a$  时域内不能经过  $(x_i, y_i)$ , 用  $c_{ij}^+$  表示;同理,右节点表示无人机  $u_i$  在  $T_{k_1}^i \pm \Delta t_a$  时域内不能经过  $(x_i, y_i)$ , 用  $c_{ij}^-$  表示,定义如下:

$$a_t = \{c_{ij}^+, c_{ij}^-\} \quad (12)$$

#### 2.1.3 奖励函数的定义描述

为引导无人机集群在环境中进行交互、更快地完成航迹冲突消解,需要定制合适的奖励函数对最优目标进行量化。除了降低航迹冲突的个数,航迹冲突发生的时间也应该作为奖励函数的重要指标。冲突发生的位置越接近终点,避免该冲突发生的规划可行解空间越少,多机航迹规划越容易陷入死锁。因此,为了促使无人机集群向无冲突解集不断收敛迭代,定制考虑稠密奖励思想的奖励函数  $r_t$ ,引导无人机在选择动作空间的动作后,根据动作前后冲突列表中冲突个数、冲突代价的变化,设计奖励函数,以快速引导集群向航迹冲突少的方案收敛:

$$r_t = r_t^n + r_t^c + r_t^r + r_t^p \quad (13)$$

式中: $r_t^n$  为冲突个数减少奖励; $r_t^c$  为冲突代价降低奖励; $r_t^r$  为冲突消解完成奖励; $r_t^p$  为当前集群总航程增加惩罚。

针对任意时刻约束树  $\tau$  对应的当前节点  $Node_{\text{now}}$ ,遍历航迹集合  $\mathcal{O}_{\text{now}}$ ,将违背式(5)的所有航迹冲突约束  $c_{ij} = (i, j, x_i, y_i, T_{k_1}^i, T_{k_2}^i)$  更新在当前冲突列表  $\mathcal{C}_{\text{now}}$  中,在选择动作空间  $a_t$  后,遍历新的航迹集合  $\mathcal{O}_{\text{now}}^{a_t}$ ,更新冲突列表为  $\mathcal{C}_{\text{now}}^{a_t}$ ,冲突个数减少奖励可以表示为

$$r_t^n = \text{size}(\mathcal{C}_{\text{now}}) - \text{size}(\mathcal{C}_{\text{now}}^{a_t}) \quad (14)$$

式中: $\text{size}(\cdot)$  表示括号内子集的个数。

各个无人机规划航迹时,随着时间的推移可行

解空间逐渐减少。因此,除了期望减少冲突的数量外,还需降低航迹冲突发生的时刻。为了描述当前冲突列表  $C_{\text{now}}$  中所有的航迹冲突发生的时刻总和,

定义航迹冲突代价为  $\sum_{l_1=1}^{\text{size}(C_{\text{now}})} T_k^{l_1}$ 。冲突代价降低奖励的计算公式为

$$r_t^c = \sum_{l_1=1}^{\text{size}(C_{\text{now}})} T_k^{l_1} - \sum_{l_2=1}^{\text{size}(C_{\text{now}})} T_k^{l_2} \quad (15)$$

由于无航迹冲突的树节点并不唯一,一旦搜索到满足要求的子节点,需要针对完成冲突消解的方案进行额外奖励:

$$r_t^r = \begin{cases} 0, & \text{size}(C_{\text{now}}^{a_t}) > 0 \\ mN, & \text{其他} \end{cases} \quad (16)$$

为了避免无人机集群在消解航迹冲突的同时增加集群总航程,削弱集群规划的最优性,需要对冲突消解方案进行航程惩罚约束:

$$r_t^p = \text{len}(\varphi_{\text{now}}) - \text{len}(\varphi_{\text{now}}^{a_t}) \quad (17)$$

式中: $\text{len}(\cdot)$  表示括号内规划航迹的总航程。

#### 2.1.4 网络与参数设计

在 DRL 中,网络具体类型的选取与输入及状态空间维度相关,当状态空间维度为三维时,卷积神经网络较为适用<sup>[25]</sup>。网络的层数和神经元个数决定了神经网络的泛化能力。考虑到过多的网络层数与神经元个数会造成神经网络的训练难度过大,而过少的网络层数与神经元个数则会导致神经网络学习能力不足,影响最终的收敛效果,本文借鉴 Rainbow DQN 模型,设计了一种改进的深度卷积神经网络架构,旨在处理状态空间的输入并有效估计冲突消解动作(树节点扩展)的价值,如图 4 所示。网络的第一部分由三层卷积层组成,分别通过 32 个  $5 \times 5$  的卷积核、64 个  $4 \times 4$  卷积核、64 个  $3 \times 3$  卷积核逐级提取输入状态空间的特征,进一步压缩空间维度,最终输出 64 个  $5 \times 5$  的特征信息。通过扁平化处理提升学习效率;最终通过全连接层进行策略学习,将卷积层提取的特征转化为策略评分,得到最优树扩展策略网络,实现对二叉树节点扩展的预示选择。Rainbow DQN 模型通过深度神经网络对大规模集群航迹规划问题的状态空间进行特征提取,以估计树节点左右扩展动作(对应两架无人机分别规避航迹冲突的方案选择)在当前状态下的 Q 值,从而形成离散的动作空间选择。

#### 2.2 基于节点重扩展机制的跳点搜索法

JPS 算法是针对栅格地图的一种优化的航迹搜索算法,该算法利用航迹中的剪枝规则与跳点判定

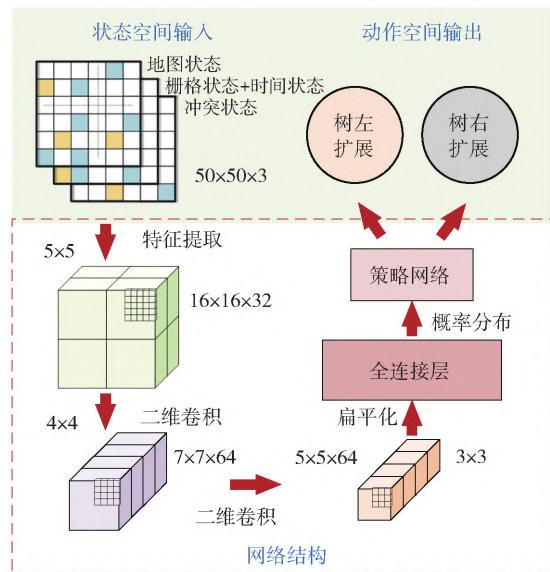


图 4 网络设计

Fig. 4 Design of network

来减少搜索空间,加速 A\* 算法的搜索过程,从而提高计算效率<sup>[21]</sup>。

JPS 算法已经证明了其在单机航迹规划的时效性上都有较强的性能优势<sup>[21]</sup>。但是对于集群协同规划场景(见图 5(a)),无人机  $u_1$ 、 $u_2$  存在航迹冲突,图 5(b)、图 5(c)分别展示了  $u_2$  和  $u_1$  规避对方航迹的规划方案,航迹最优性下降甚至造成规划无解。这表明了空域栅格的占有率具有强时效性,将其他无人机经过的栅格单元视为障碍(见图中的红色栅格)虽然适用于 JPS 算法的规划要求,但是不考虑时间维度的障碍信息削减了可行空间,进而造成集群规划无解的情况,往往与事实相悖。因此,为了扩大集群规划的可行解空间,本文在 JPS 算法中障碍信息仅考虑空间位置的基础上,将时间维度引入障碍信息,并设计了考虑时序约束的节点重扩展机制。

#### 2.2.1 考虑邻域时间尺度的地图状态信息更新

以图 5(a)中发生的冲突为例,为无人机集群规划航迹时,将其他无人机的航迹视作拥有时间维度的障碍信息,无人机  $u_1$  的航迹为  $\xi_1 = \{p_1^{T_0}, p_1^{T_1}, \dots, p_1^{T_8}\}$ ,无人机  $u_2$  的航迹为  $\xi_2 = \{p_2^{T_0}, p_2^{T_1}, \dots, p_2^{T_8}\}$ 。两机分别在各自  $p_1^{T_3}, p_2^{T_3}$  处发生航迹冲突(见图 5(a)中的航迹交叉点)。对于 JPS 算法搜索跳点的循环中,地图状态信息涵盖了障碍信息与邻域时间尺度内其他无人机的位置信息。如图 5(d)所示,对于无人机  $u_2$  在  $p_2^{T_2}$  处的跳点搜索,有且仅有两个蓝色栅格不满足  $p_2^{T_2}$  的邻域时间要求(见式(5)),相较于

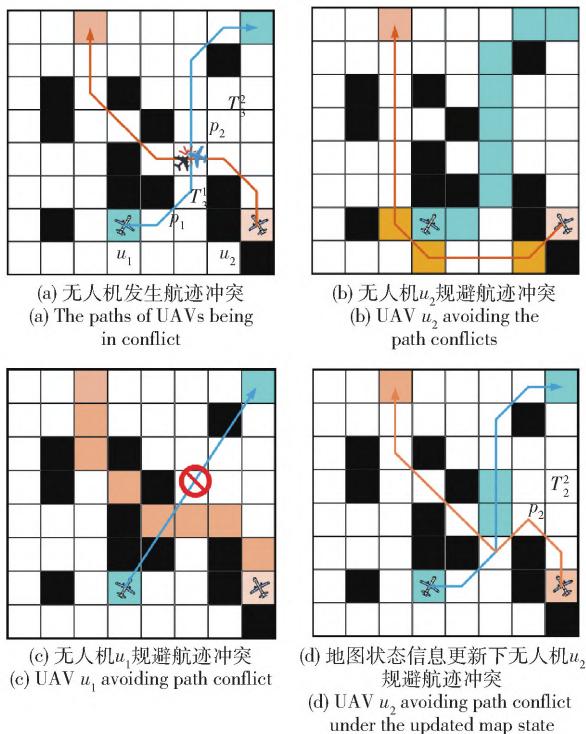


图 5 经典 JPS 算法对于多机规划的不足

Fig. 5 Limitations of the classic JPS algorithm for UAV swarm path planning

不考虑邻域时间约束的搜索方法,规划可行解空间(见图中的白色栅格)显著增加。

## 2.2.2 考虑时序约束的节点重扩展机制

JPS 算法隶属于图搜索方法,在航迹探索过程中,Open 表对于同一栅格位置仅保留最低的节点代价,避免重复扩展该点而无谓地增加无人机航程,导致规划死锁概率增大的同时降低航迹规划的最优性。当无人机  $u_i$  面临航迹冲突  $c_{ij}^-$  约束时,其在  $T_{k_1}^i \pm \Delta t_a$  时域内不能经过  $(x_i, y_i)$ 。基于节点代价最小的探索策略将所有非最小节点代价扩展至  $(x_i, y_i)$  的栅格单元从 Open 表中移除,而这其中往往包含了不在  $T_{k_1}^i \pm \Delta t_a$  时域内经过  $(x_i, y_i)$  的航迹规划点的情况。

考虑时序约束的节点重扩展机制在节点探索的过程中,保留了节点位置相同但节点代价、回溯航迹不同的节点信息。如图 6 所示,无人机从起点  $g_{1,3}$  出发前往终点  $g_{2,4}$ ,考虑到规避其他无人机的航迹冲突约束  $T_{k_1}^i \pm \Delta t_a = 1.4 \pm 0.8$ ,节点扩展的过程中产生 3 种不同节点代价的航迹方案。方案 1、2 虽然航迹较短,但是无人机到达终点的时间  $T_{k_1}^i$  违背了航迹冲突约束。方案 3 的到达时间  $T_{k_1}^i = 4.2$  虽然最优化较差,但是满足集群安全性要求,考虑时序约束

的节点重扩展机制有效提升了集群航迹冲突消解的可行域。

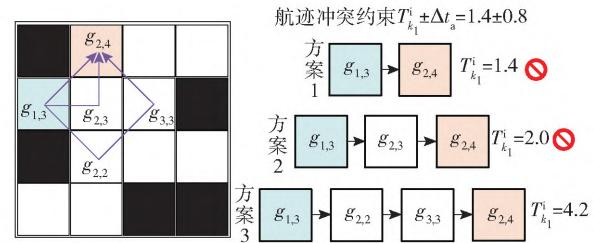


图 6 节点重扩展方案示意图

Fig. 6 Schematic diagram of node re-expansion plan

综合上述研究,给出基于节点重扩展机制的跳点搜索法流程如下(伪代码见表 1)。

表 1 算法伪代码

Table 1 Pseudocode of algorithm

### 算法: 基于节点重扩展机制的跳点搜索法

**Input:**  $g_{xy}^{pro=2}, g_{xy}^{pro=3}, C_{now}, Con, Open, Close$

**Output:**  $\xi_i = \{p_i^1, p_i^2, \dots, p_i^k\}, k \in \mathbb{Z}^+$

```

1 Get positiong  $g_{xy}^{pro=2}, g_{xy}^{pro=3}$ ;
2 Initialize  $C_{now}, Con, Open, Close$ ;
3 While  $Open \neq \emptyset$  do
4   Find  $g_{xy} = \min(Open)$ 
5   Do prune rules:  $Flag = 1$ 
6   while  $g_{xy}$  expands no jump point || jump point in  $Con$ 
7     If  $g_{xy}$  reach boundary
8        $Flag = 0$ ;
9     Break;
10    else
11       $g_{xy}$  = diagonal search  $g_{xy}$ 
12    End if
13  End while
14  If  $Flag$ 
15     $g_{xy}$  from Open to Close, jump point in Open
16  End if
17  If  $g_{xy} = g_{xy}^{pro=3}$ 
18    Return  $\xi_i = \{p_i^1, p_i^2, \dots, p_i^k\}, k \in \mathbb{Z}^+$ 
19  End if
20 End while

```

**步骤 1** 初始化起点位置  $g_{xy}^{pro=2}$ 、终点位置  $g_{xy}^{pro=3}$ ; 考虑此时冲突列表  $C_{now}$  与约束树节点的约束信息  $Con$ , 将起点位置加入 Open 表, 置 Close 表为空。

**步骤 2** 迭代搜索,当 Open 表为空时规划失

败,暂无可行航迹,结束循环。

**步骤 3** Open 表不为空时,依据启发式函数  $f(g_{xy}) = g(g_{xy}) + h(g_{xy})$  选择节点代价值  $f(g_{xy})$  最小的栅格单元,作为当前节点进行扩展。

**步骤 4** 扩展规则采用剪枝规则,更新考虑邻域时间的地图状态信息,采用直线搜索,判断水平与竖直方向是否存在跳点且该跳点不处于约束信息  $Con$  中,如果存在,则将扩展出的跳点加入 Open 表中,如果不存在,则根据父节点方向斜向前进一格,重复执行步骤 4,直至到达地图边界。

**步骤 5** 从 Open 表中移除当前节点  $g_{xy}$ ,Close 表中加入当前节点  $g_{xy}$ 。

**步骤 6** 判断  $g_{xy}$  是否为终点位置  $g_{xy}^{pro=3}$ ,如果是则算法结束,如果不是,则重复执行步骤 3。

**步骤 7** 由终点位置  $g_{xy}^{pro=3}$  开始回溯航迹上每个栅格单元,直至起点位置,并依次连接完成航迹规划。

### 3 数值仿真实验

为验证本文所提 DRL 增强的冲突搜索方法(DRLCBS + ReJPS)在大规模无人机集群航迹规划中航迹冲突的消解能力与实时规划能力,本节设置了三组想定测试:想定 1 验证底层航迹规划算法(ReJPS)的规避航迹冲突的能力;想定 2 验证本文所提冲突搜索方法处理不同规模无人机集群冲突消解的能力;想定 3 设置随机仿真环境,通过蒙特卡洛仿真测试与 CBS + ReJPS, CBS + A\*, CBS + MSA\*<sup>[26]</sup> 等方法进行对比,验证本文所提方法的最优性与鲁棒性。

仿真硬件为 Intel Core i7-6700 CPU 3.40 GHz, 8G RAM。

#### 3.1 ReJPS 规避航迹冲突能力测试

为验证 ReJPS 算法中引入时间维度并设计节点重扩展机制能够有效增加规划可行域、提升航迹冲突的消解能力,在  $14 \times 14$  的地图中随机生成 60 个障碍威胁栅格。无人机集群从相同起点出发到达相同终点位置(起点、终点位置不做空域冲突检测),规划满足式(5)冲突约束的航迹,其中邻域时间尺度  $\Delta t_a = 1$ 。

分别对无人机  $u_1 \sim u_5$  进行 ReJPS 航迹规划,前序无人机的规划航迹作为拥有时间维度的障碍信息,约束后续无人机的航迹规划。如图 7 所示,ReJPS 能够引导无人机集群在受限空域下规划无航迹冲突的航迹。航迹冲突检测如表 2 所示,各架无人

机到达航迹冲突栅格时到达时间均满足邻域时间尺度要求,无航迹冲突发生。

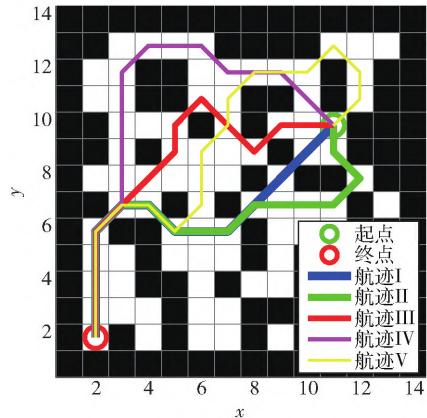


图 7 ReJPS 算法规避航迹冲突示意图

Fig. 7 Illustration of ReJPS algorithm for avoiding path conflicts

表 2 航迹冲突检测

Table 2 Conflict detection

航迹冲突坐标	无人机到达时间				
	$u_1$	$u_2$	$u_3$	$u_4$	$u_5$
(8, 7)	4.24	5.83			
(9, 12)	2.83	6.24			
(7, 10)	4.83	9.66			
(5, 6)	7.65	9.24	14.49		
(3, 7)	10.07	12.66	11.49	14.66	16.90

对于同一想定,分别对无人机  $u_1 \sim u_5$  进行 JPS 航迹规划,如图 8 所示,JPS 仅能完成  $u_1$  的航迹规划。对于  $u_2$  的航迹规划,考虑障碍威胁栅格、 $u_1$  航迹约束和邻域时间约束  $\Delta t_a = 1$ ,可行解空间如图 9 所示,其中栅格的颜色代表  $u_2$  飞行至该位置时与其他无人机的飞行间隔,即数值越高安全性越好,而由于 JPS 算法在探索空间时,Open 表对于同一栅格位置仅保留最低的节点代价值,无法重复访问同一栅格位置,因此即使探索到栅格  $g_{3,7}$ ,但  $g_{3,7}$  的到达时间与  $u_1$  到达  $g_{3,7}$  的时间不满足邻域时间的约束要求,因此发生了规划死锁,最终无法找到规划可行解。

由此可见,ReJPS 算法通过引入时间维度并设计节点重扩展机制,能够有效增加规划可行域,进而提升航迹冲突的消解能力。

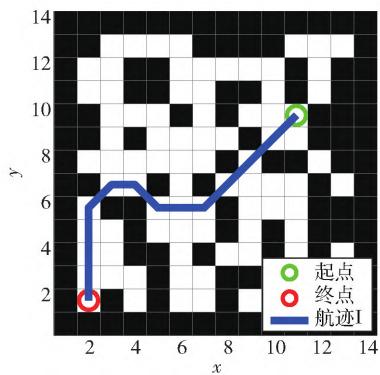


图 8 JPS 算法规避航迹冲突示意图

Fig. 8 Illustration of JPS algorithm for avoiding path conflicts

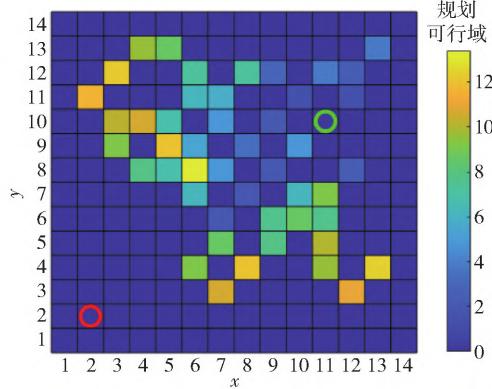


图 9 规划可行解空间

Fig. 9 Feasible solution space of path planning

### 3.2 不同规模无人机集群航迹冲突消解能力测试

随着无人机集群在单位空域内的密度增加,局部航迹冲突的消解可能会诱发新的冲突,进而造成冲突的链式反应,而单机航迹长度的增加会增大碰撞的概率,因此无人机规模、空域范围、航迹长度都将直接影响无人机集群的冲突数量。而为了验证本文所提方法(DRLCBS + ReJPS)在不同规模集群下的航迹冲突消解适应性,本节实验固化了空域范围和航迹长度:在 $50 \times 50$ 地图中随机生成500个障碍威胁栅格;在随机生成的地图 $G = \{g_{xy}^{pro}\}_{50 \times 50}$ 中随机生成各个无人机的起点位置和终点位置,为了避免各架无人机的起点、终点过近导致航迹冲突个数不足,从而无法体现本文方法的冲突消解能力,随机生成的起点 $(x_s, y_s)$ 和终点 $(x_e, y_e)$ 需要满足条件 $|x_s - x_e| + |y_s - y_e| \geq 10$ 。

无人机集群的最大规模为128,如图10所示,小规模( $< 16$ )集群协同航迹规划几乎不存在航迹冲突,只需要利用ReJPS进行底层航迹规划,DRLCBS维护的二叉约束树直接收敛到无约束树节点。

随着无人机集群规模的增加,集群内部航迹冲突的数量呈指数级增长,冲突消解的难度随之增加,二叉约束树的深度增加导致DRLCBS与ReJPS的计算次数增加,最终导致冲突消解的规划时间增加。

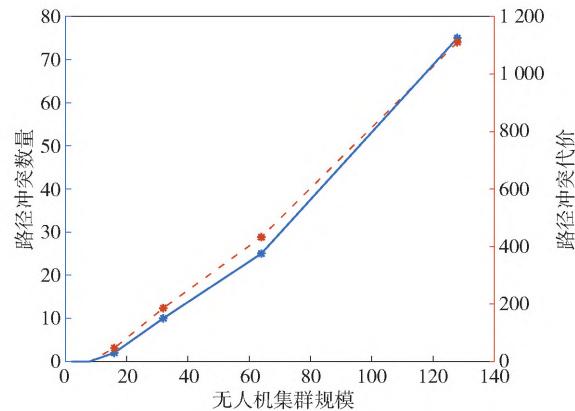
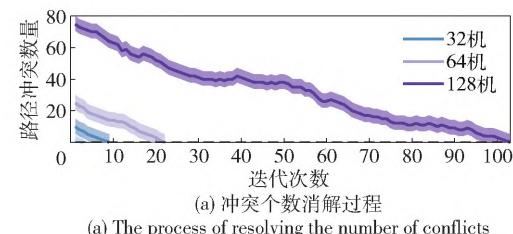


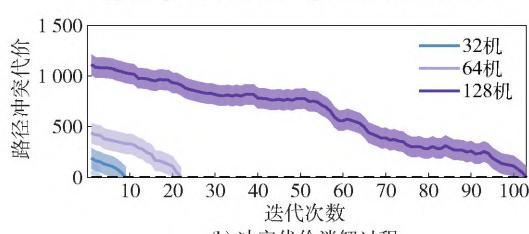
图 10 不同规模的无人机集群初始航迹冲突数量与代价

Fig. 10 Initial number and cost of path conflict for UAV swarms with different scales

为了进一步体现算法在不同规模无人机集群航迹规划中冲突消解迭代次数与航迹冲突数量、航迹冲突代价的影响,本文给出了不同规模无人机集群航迹规划冲突消解过程(见图11)。由图11可见:集群规模较小时,一次迭代可以消解一个航迹冲突,冲突消解过程线性向下直至冲突消解完毕;随着集群规模的提升,冲突个数的消解过程在迭代中有所震荡,主要是因为集群在受限空域下拥挤造成航迹冲突耦合性较强,二叉树两两冲突消解的同时造成新的耦合冲突的出现,冲突消解能力受到制约,但是随着迭代次数的提升,128机的大规模集群依然能



(a) 冲突个数消解过程



(b) 冲突代价消解过程

Fig. 11 Deconfliction of UAV swarms with different scales

在 DRLCBS + ReJPS 方法引导下规划满足约束的无航迹冲突解, 冲突消解能力得到验证。

由图 12 中统计的规划时间可知, 32 机、64 机规模规划耗时均在亚秒级内; 128 机规模的无人机集群规划耗时为 11.41 s, 顶层规划耗时约占总耗时的 48.92%。以上结果表明本文所提 DRL 增强的冲突搜索方法在面向不同规模无人机集群时均具有良好的航迹冲突消解能力与实时规划能力。

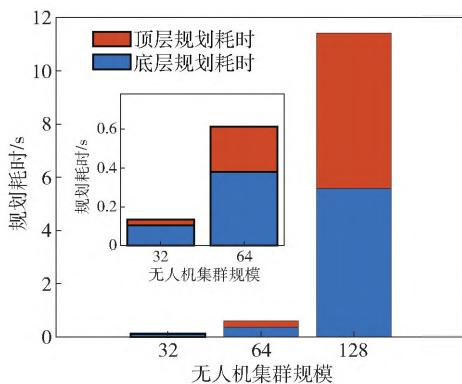


图 12 不同规模的无人机集群航迹冲突消解时间

Fig. 12 Deconfliction time of UAV swarms with different scales

### 3.3 蒙特卡洛仿真测试

为进一步验证 DRLCBS + ReJPS 算法在消解大规模无人机集群航迹冲突的性能优势, 在随机栅格地图仿真场景下开展 500 次蒙特卡洛仿真测试, 想定设置同 3.2 节, 依然在随机生成的地图  $G = \{g_{xy}^{pro}\}_{50 \times 50}$  中随机生成各个无人机的起点位置和终点位置, 随机生成的起点  $(x_s, y_s)$  和终点  $(x_e, y_e)$  需要满足条件  $|x_s - x_e| + |y_s - y_e| \geq 10$ 。由于本文所提方法采用基于冲突搜索的双层规划方法, 参与对比的算法也沿用双层规划架构如表 3 所示。顶层规划通过比较算法耗时验证冲突消解方法的时效性, 底层航迹规划方法通过比较航程耗时验证算法的最优性。

表 3 蒙特卡洛仿真对比方法

Table 3 Different methods in Monte Carlo simulation

对比方法	顶层规划方法	底层规划方法
1	CBS	A*
2	CBS	MSA*
3	CBS	ReJPS
4	DRLCBS	ReJPS

图 13 展示了 4 种对比方法的集群总航程, 作为评估冲突消解方案最优性的指标。由图 13 可见:

CBS + A\* 方法最优性最好, 略优于 CBS + ReJPS 方法与 DRLCBS + ReJPS 方法, 主要是因为 JPS 航迹规划方法的最优性不及 A\*; MSA\* 方法本质上作为一种滚动规划方法, 最优性低于 A\*、JPS 等直接式规划方法。但是整体而言, CBS + A\* 方法与 CBS + ReJPS、DRLCBS + ReJPS 方法的最优性相当。

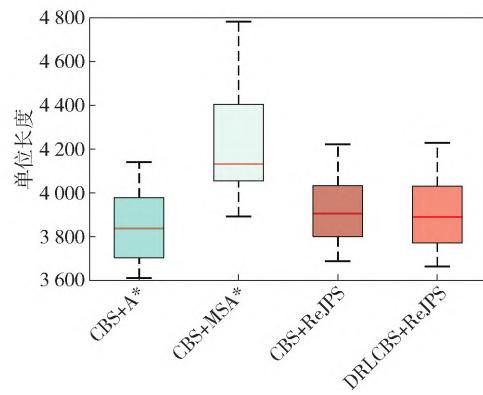


图 13 航程对比

Fig. 13 Comparison of flight times

蒙特卡洛规划结果如图 14 所示, 在相同的仿真场景与(CBS)算法条件下, 基于 ReJPS 的底层航迹规划方法相比于 A\* 算法, 规划耗时平均降低 56.95%; 相比滚动规划 MSA\* 算法, 规划耗时平均降低 80.08%。相比于 CBS 冲突消解方法, 经过 DRL 增强的 DRLCBS 冲突消解的迭代次数更少, 减少了底层规划算法的调用次数, 底层航迹规划耗时也随之减少。在相同的仿真场景与底层航迹规划算法(ReJPS)条件下, DRLCBS 相比于 CBS 冲突消解能力平均提升 20.84%。综合基于冲突搜索的双层规划框架, DRLCBS + ReJPS 在大规模无人机集群冲突消解总耗时最少, 相比于 CBS + A\* 方法、CBS + MSA\* 方法与 CBS + ReJPS 方法, 平均规划耗时分别降低了 86.64%、93.83% 和 19.65%, 实验结果证明了 DRLCBS + ReJPS 冲突消解方法的实时性。

### 4 结论

本文探究了无人机集群规模对于本文所提 DRLCBS + ReJPS 方法的影响, 并通过蒙特卡洛仿真实验的方式将本文方法与 CBS + A\* 方法、CBS + MSA\* 方法与 CBS + ReJPS 方法进行了对比。得出主要结论如下:

1) 128 机规模的无人机集群规划耗时仅为 11.41 s, 顶层规划耗时约占总耗时的 48.92%, 具有良好的实时性。

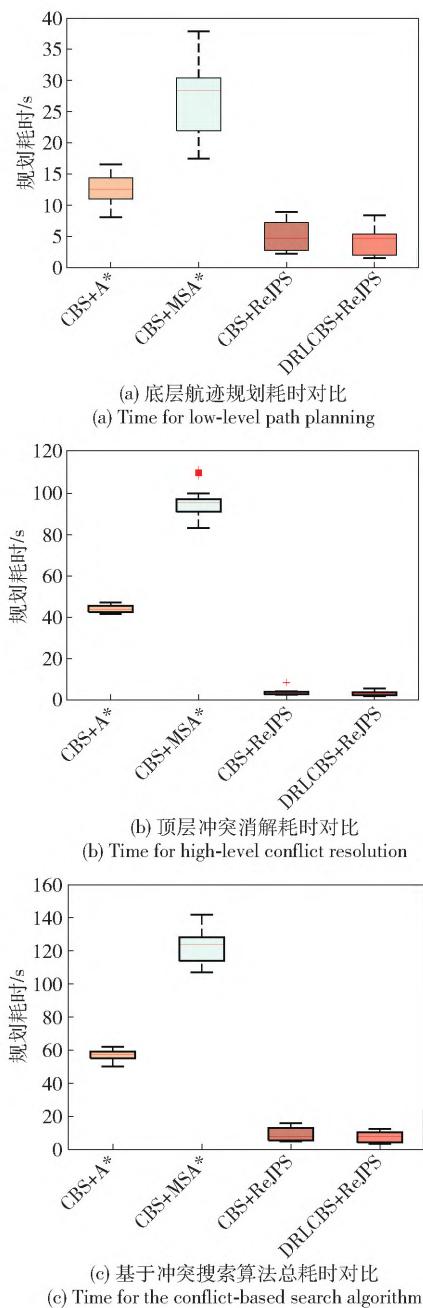


图 14 规划耗时对比

Fig. 14 Comparison of planning times

2) 基于 ReJPS 的底层航迹规划算法相比 A\* 算法规划耗时平均降低 56.95%, 经过 DRL 增强的 DRLCBS 方法冲突消解能力相比于 CBS 平均提升 20.84%。

3) 相较于 CBS + A\* 方法、CBS + MSA\* 与 CBS + ReJPS 等冲突搜索方法, 本文所提冲突搜索方法 DRLCBS + ReJPS 在最优性相当的情况下, 平均规划耗时分别降低了 86.64%、93.83% 和 19.65%。

尽管本文提出的 DRLCBS + ReJPS 方法在大规模无人机集群航迹规划中表现出较高的实时性与安

全性, 但当前方法主要针对已知或可预测的障碍物分布, 对完全未知动态障碍物(例如突发移动障碍)的实时避碰能力仍需提升; 算法的仿真验证与实际物理平台的传感器噪声、通信延迟存在差异。后续需通过虚实融合的半物理仿真平台进行验证, 并引入鲁棒性优化机制。

此外, 随着空域资源的日益紧张与无人机集群应用场景的复杂化(例如城市物流、灾害救援), 本文提出的“DRL + 分层冲突消解”框架为大规模集群高效规划提供了新的技术路径。未来的研究可以进一步优化算法的计算效率和对环境的适应性, 以应对更复杂的飞行工况和场景。通过进一步融合在线学习、分布式计算与高精度环境建模, 实现“超大规模-高动态-强安全”的下一代集群航迹自主规划系统, 推动无人机集群从理论验证走向工程落地。

## 参考文献(References)

- [1] GHOMMAM J, SAAD M, WRIGHT S, et al. Relay manoeuvre based fixed-time synchronized tracking control for UAV transport system [J]. Aerospace Science and Technology, 2020, 103: 105887.
- [2] SHAHI T B, XU C Y, NEUPANE A, et al. Machine learning methods for precision agriculture with UAV imagery: a review[J]. Electronic Research Archive, 2022, 30(12):4277–4317.
- [3] KHAN A, GUPTA S, GUPTA S K. Emerging UAV technology for disaster detection, mitigation, response, and preparedness [J]. Journal of Field Robotics, 2022, 39(6):905–955.
- [4] 陈亚萍, 王楠, 洪华杰, 等. 面向多无人平台区域监视任务的信息素正向激励栅格方法 [J]. 兵工学报, 2023, 44(9): 2859–2870.  
CHEN Y P, WANG N, HONG H J, et al. Pheromone positive incentive grid method for multi-unmanned platform regional surveillance task[J]. Acta Armamentarii, 2023, 44(9):2859 – 2870. (in Chinese)
- [5] CHANG G N, FU W X, ZHAO J M, et al. Overview of research on intelligent swarm munitions [J/OL]. Defence Technology, 2024, DOI: <https://doi.org/10.1016/j.dt.2024.08.017>.
- [6] 李军, 陈士超. 无人机蜂群关键技术发展综述 [J]. 兵工学报, 2023, 44(9):2533–2545.  
LI J, CHEN S C. Overview of key technology and its development of drone swarm [J]. Acta Armamentarii, 2023, 44(9):2533 – 2545. (in Chinese)
- [7] 赵军民, 何浩哲, 王少奇, 等. 复杂环境下多无人机目标跟踪与避障联合航迹规划 [J]. 兵工学报, 2023, 44(9):2685 – 2696.  
ZHAO J M, HE H Z, WANG S Q, et al. Joint trajectory planning for multiple UAVs target tracking and obstacle avoidance in a complicated environment[J]. Acta Armamentarii, 2023, 44(9): 2685 – 2696. (in Chinese)

- [8] 于连波,曹品钊,石亮,等.基于改进冲突搜索的多智能体路径规划算法[J].航空学报,2023,44(增刊1):727648.
- YU L B, CAO P Z, SHI L, et al. An improved conflict-based search algorithm for multi-agent path planning [ J ]. Acta Aeronautica et Astronautica Sinica, 2023, 44 (S1):727648. (in Chinese)
- [9] HONG Y K, KIM Y D. Two-stage multicriteria decision-making framework for aircraft conflict resolution[J]. Journal of Aerospace Information Systems, 2023, 20 (10):596 – 604.
- [10] LIU Z X, CAI K Q, XIE J D, et al. A network-based conflict resolution approach for unmanned aerial vehicle operations in dense nonsegregated airspace[J]. IEEE Intelligent Transportation Systems Magazine, 2022, 14 (3):212 – 232.
- [11] VAN DEN BERG J, SNOEYINK J, LIN M C, et al. Centralized path planning for multiple robots: optimal decoupling into sequential plans [ C ] // Proceedings of Robotics: Science and Systems. Seattle, WA, US: MIT Press, 2009.
- [12] YANG J, YIN D, NIU Y F, et al. Distributed cooperative onboard planning for the conflict resolution of unmanned aerial vehicles [ J ]. Journal of Guidance, Control, and Dynamics, 2019, 42 (2):272 – 283.
- [13] 徐广通,王祝,曹严,等.动态优先级解耦的无人机集群轨迹分布式序列凸规划[J].航空学报,2022,43(2):325059.  
XU G T, WANG Z, CAO Y, et al. Dynamic-priority-decoupled UAV swarm trajectory planning using distributed sequential convex programming [ J ]. Acta Aeronautica et Astronautica Sinica, 2022, 43 (2):325059. (in Chinese)
- [14] REN Z Q, RATHINAM S, CHOSET H. CBSS: a new approach for multiagent combinatorial path finding[J]. IEEE Transactions on Robotics, 2023, 39 (4):2669 – 2683.
- [15] SHARON G, STERN R, FELNER A, et al. Conflict-based search for optimal multi-agent pathfinding [ J ]. Artificial Intelligence, 2015, 219:40 – 66.
- [16] 王子晗,童向荣.基于冲突搜索的多智能体路径规划研究进展[J].计算机科学,2023,50(6):358 – 368.  
WANG Z H, TONG X R. Research progress of multi-agent path finding based on conflict-based search algorithms[ J ]. Computer Science, 2023, 50 (6):358 – 368. (in Chinese)
- [17] REN Z Q, RATHINAM S, CHOSET H. A conflict-based search framework for multiobjective multiagent path finding[ J ]. IEEE Transactions on Automation Science and Engineering, 2023, 20 (2):1262 – 1274.
- [18] BOYARSKI E, FELNER A, STERN R, et al. ICBS: the improved conflict-based search algorithm for multi-agent pathfinding; extended abstract [ C ] // Proceedings of the 8th Annual International Symposium on Combinatorial Search. Ein Gedi, the Dead Sea, Israel: Israel Science Foundation, 2015: 223 – 225.
- [19] SHARON G, STERN R, FELNER A, et al. Meta-agent conflict-based search for optimal multi-agent path finding [ C ] // Proceedings of the 5th Annual Symposium on Combinatorial Search. Niagara Falls, Ontario, Canada: Association for the Advancement of Artificial Intelligence, 2012:97 – 104.
- [20] HUANG T A, KOENIG S, DILKINA B. Learning to resolve conflicts for multi-agent path finding with conflict-based search [ C ] // Proceedings of the AAAI Conference on Artificial Intelligence. Reston, VA, US:AAAI, 2021:11246 – 11253.
- [21] 周熙栋,张辉,陈波.非结构化场景下基于改进JPS算法的移动机器人路径规划[J].控制与决策,2024,39(2):474 – 482.  
ZHOU X D, ZHANG H, CHEN B. Mobile robot path planning based on improved JPS algorithm in unstructured scenarios[ J ]. Control and Decision, 2024, 39 (2):474 – 482. (in Chinese)
- [22] ZHANG J C, AN Y Q, CAO J N, et al. UAV trajectory planning for complex open storage environments based on an improved RRT algorithm[ J ]. IEEE Access, 2023, 11:23189 – 23204.
- [23] MAO T X, DENG H. Path planning of slender tensegrities based on the artificial potential field method[ J ]. AIAA Journal, 2023, 61 (5):2255 – 2265.
- [24] HESSEL M, MODAYIL J, VAN HASSELT H, et al. Rainbow: combining improvements in deep reinforcement learning [ C ] // Proceedings of the AAAI Conference on Artificial Intelligence. New Orleans, LA, US:AAAI, 2018:3215 – 3222.
- [25] JI S W, XU W, YANG M, et al. 3D convolutional neural networks for human action recognition [ J ]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35 (1): 221 – 231.
- [26] XIANG J, CHEN J, LIU Y C. Hybrid multiscale search for dynamic planning of multi-agent drone traffic [ J ]. Journal of Guidance, Control, and Dynamics, 2023, 46 (10): 1963 – 1974.