

引用格式: 郭凯, 姚明启, 乔磊. 基于仿射变换的多UUV自适应队形变换路径规划方法[J]. 中国舰船研究, 2026, 21(2): 101-111.
GUO K, YAO M Q, QIAO L. A path planning method for adaptive formation reshaping of multi-UUVs based on affine transformation[J]. Chinese Journal of Ship Research, 2026, 21(2): 101-111 (in both Chinese and English).

基于仿射变换的多UUV自适应 队形变换路径规划方法



扫码阅读全文

郭凯, 姚明启, 乔磊*

上海交通大学 船舶海洋与建筑工程学院, 上海 200240

摘要: [目的] 针对多无人潜航器(UUV)编队在复杂水下环境中面临的编队保持与灵活避障难以同时实现的难题, 提出一种能够自适应队形变换的全局路径规划方法。 [方法] 基于仿射变换框架, 将多UUV系统的协同路径规划问题映射至二维平面仿射参数空间进行求解。首先, 采用一种改进的快速搜索随机树*(RRT*)算法进行前端路径搜索, 该算法通过整合快速探索与迭代优化阶段、加权KD(k-dimensional)树、混合采样机制以及自适应采样参数调整, 实现初始仿射状态序列的高效生成; 然后, 利用基于B样条曲线的后端优化器, 采用梯度下降法最小化一个综合考虑轨迹平滑性、UUV运动学可行性、环境碰撞安全性以及自适应编队缩放代价的目标函数, 优化得到连续光滑且满足多重约束的仿射参数轨迹。 [结果] 湖上试验结果表明, 所提规划方法能够生成安全、可行的编队路径, 能成功引导多UUV编队穿越模拟的狭窄障碍区域, 且UUV的实际速度和加速度均满足设定的可行性约束。 [结论] 所提的基于仿射变换的全局规划方法能够通过自适应队形变换为航行于复杂障碍场景下的多UUV编队提供安全可行的编队路径, 实现多UUV编队的安全航行, 同时, 也能显著提升海洋无人潜航器的自主性与环境适应性, 对推动海洋无人装备技术的发展和实际应用具有积极意义。

关键词: 无人潜航器; 编队路径规划; 仿射变换; 快速搜索随机树*; 轨迹优化

中图分类号: U674.941

文献标志码: A

DOI: 10.19693/j.issn.1673-3185.04560

0 引言

随着海洋探索与开发任务的日益复杂, 单个无人潜航器(unmanned underwater vehicle, UUV)在感知范围、续航能力、任务适应性以及应急处理方面开始显现出其局限性^[1-2]。多UUV通过分布式协作与协同控制, 能够显著提升任务的执行效率, 扩大作业范围, 增强系统鲁棒性与容错能力, 在海洋环境监测、资源勘探、水下搜救和安全巡逻等领域具有重要的应用价值^[3-4]。然而, 复杂多变的水下环境, 特别是障碍物密集的区域, 给多UUV编队协同作业提出了严峻的挑战。因此, 如何在保持编队稳定性的同时实现灵活、自主、高效的队形变换以规避障碍物, 是提升多UUV编队系统环境适应性和任务安全性的关键。

理想的多智能体编队系统需要能在规避障碍

的同时, 还能维持必要的编队结构或是根据环境约束灵活调整队形^[5]。针对多智能体编队避障问题, 一种主流方法是基于个体轨迹优化的分布式策略, 即通过为编队内每个个体独立规划避障轨迹, 并在优化目标中加入队形约束项来协调避障与编队保持^[6-7], 试图在个体避障与整体队形保持间取得平衡。但在密集障碍或通信受限的环境下, 易引发编队变形, 如个体避障绕行导致的编队解列。对于水下通信带宽有限且存在延迟、遮挡的UUV系统, 这种依赖个体频繁信息交互和独立决策的分布式方法稳定性较差。另一类方法侧重于将编队作为一个整体进行避障规划, 即将编队视为一个虚拟的刚性结构或凸包区域^[8-9], 通过规划该整体区域的路径或优化其中心位置来避开障碍物。这种方法简化了问题并能严格维持预设队形, 但主要缺陷在于缺乏局部调整能力, 即

收稿日期: 2025-06-16 修回日期: 2025-10-13 网络首发时间: 2025-11-11 17:58

基金项目: 国家自然科学基金资助项目(52101365); 中国科协青年人才托举工程项目(2021QNRC001)

作者简介: 郭凯, 男, 2001年生, 硕士生。研究方向: 海洋机器人协同编队规划。E-mail: guokai2023@sjtu.edu.cn

乔磊, 男, 1989年生, 博士, 副教授, 博士生导师。研究方向: 海洋智能机器人和无人系统。

E-mail: qiaolei@sjtu.edu.cn

*通信作者: 乔磊

使编队中仅有少数个体面临碰撞风险,也需要整个编队进行统一的避让机动,这不仅会导致路径过度保守,也使其难以适应需要精细局部调整的复杂障碍环境。还有一些学者针对通过调整编队队形本身实现避障,在触发特定环境条件时切换至适配队形以通过障碍进行了探索^[10]。然而,该方法受限于预设队形库的完备性以及切换算法的实时性,因队形类别不足难以应对复杂障碍布局,而库容过大又会导致计算效率下降,难以满足动态环境响应需求。

为了实现更平滑、灵活、连续的队形调整以适应避障需求,将仿射变换(affine transformation, AT)理论引入了编队机动控制领域^[11-14]。仿射变换是一种几何变换,能够保持几何形状的平行性,由该变换定义的编队构型可以统一进行缩放、旋转、平移和剪切等几何变换操作,从而改变编队的整体形态和朝向,这为编队适应性避障提供了新的可能性。Pang等^[11]和Zhao^[12]利用仿射编队结构实现了UUV编队的整体缩放与旋转,并结合人工势场法引导编队绕过了障碍物。但变换参数依赖于预设规则或领导者路径,缺乏对环境信息的主动响应。因此,本文拟针对现有方法在自主适应环境进行队形变换避障方面的问题,提出一种基于仿射变换的多UUV全局规划方法。首先,将编队轨迹规划问题映射到仿射参数空间,通过改进的快速搜索随机树*(rapidly-exploring random tree*, RRT*)算法为多UUV编队生成考虑队形变换的全局初始路径点;然后,在仿射参数空间进行参数轨迹优化,以生成既满足避障、避碰约束,又保证队形变换平滑连续的编队路径。

1 问题描述

1.1 UUV运动学及动力学模型

本文将研究UUV在恒定深度下的水平面运动,并只考虑纵荡、横荡和艏摇三个自由度。以第 j 个UUV为例,其坐标系定义和UUV的状态如图1所示。图中: O_E 表示惯性坐标系,所有UUV共用一个惯性坐标系; O_B^j 表示第 j 个UUV的机体坐标系; x_j, y_j, ψ_j 分别为UUV在固定坐标系下的纵、横向位置和艏向角; u_j, v_j 分别为UUV在载体坐标系下的纵荡和横荡速度; $V_j = [u_j, v_j]^T$,为二维速度矢量。根据文献[15]中建立的UUV数学模型,本文的三自由度运动学方程定义如下:

$$\dot{\eta}_j = \begin{bmatrix} \mathbf{R}_j(\psi_j) & \mathbf{0}_{2 \times 1} \\ \mathbf{0}_{1 \times 2} & 1 \end{bmatrix} \mathbf{v}_j \quad (1)$$

式中: $\eta_j = [x_j, y_j, \psi_j]^T$, $\mathbf{v}_j = [u_j, v_j, r_j]^T$,其中 r_j 为艏向角速度; $\mathbf{R}_j(\psi_j)$ 为坐标旋转矩阵。

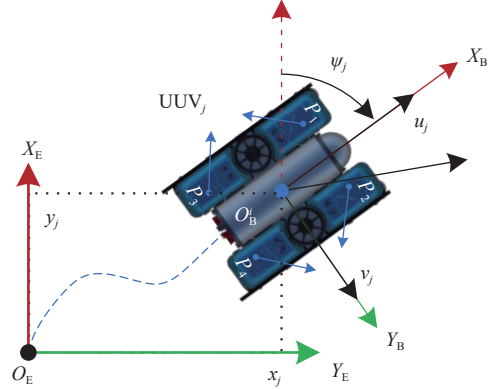


图1 第 j 个UUV的坐标系和状态

Fig. 1 Coordinate system and state of the j -th UUV

第 j 个UUV的三自由度动力学方程定义如下:

$$\begin{cases} \mathbf{M}_j \dot{\mathbf{v}}_j + \mathbf{C}_j(\mathbf{v}_j) \mathbf{v}_j + \mathbf{D}_j(\mathbf{v}_j) \mathbf{v}_j = \boldsymbol{\tau}_j + \boldsymbol{\tau}_d^j \\ \boldsymbol{\tau}_j = \mathbf{T}_j \mathbf{u}_j \end{cases} \quad (2)$$

式中: $\mathbf{M}_j \in \mathbb{R}^{3 \times 3}$,为惯性矩阵,包括刚体惯性和水动力附加质量; $\mathbf{C}_j(\mathbf{v}_j)$, $\mathbf{D}_j(\mathbf{v}_j)$ 分别为科里奥利和向心力矩阵以及水动力阻尼矩阵; $\boldsymbol{\tau}_j = [X_j, Y_j, N_j]^T$,为由 $P_1 \sim P_4$ 推进器推力在纵向、横向和垂向产生的合力向量; $\boldsymbol{\tau}_d^j$ 为环境扰动; $\mathbf{T}_j \in \mathbb{R}^{3 \times 4}$,为推进器的推力分配矩阵; $\mathbf{u}_j = [P_1, P_2, P_3, P_4]^T$,为4个推进器上的推力向量。

1.2 仿射编队与规划目标

为了对多UUV目标编队进行统一的几何变换,将采用基于齐次坐标的二维仿射变换^[11]描述目标编队构型的动态变化。定义一组包含 N_U ($N_U \geq 3$)个UUV的编队系统,仿射编队结构允许目标编队在标称编队构型的基础上进行平移、旋转、缩放和剪切等操作,也即任意时刻 t 的目标编队构型 $\tilde{\mathbf{P}}^*(t)$ 是标称编队构型 $\tilde{\mathbf{P}}^0$ 的一种仿射变换,计算方式如下:

$$\tilde{\mathbf{P}}^*(t) = \mathbf{A}(t) \tilde{\mathbf{P}}^0 \quad (3)$$

$$\tilde{\mathbf{p}}_j^*(t) = \begin{bmatrix} x_j^*(t) \\ y_j^*(t) \\ 1 \end{bmatrix} = \begin{bmatrix} A_1 & A_3 & A_5 \\ A_2 & A_4 & A_6 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_j^0 \\ y_j^0 \\ 1 \end{bmatrix} = \mathbf{A}(t) \tilde{\mathbf{p}}_j^0 \quad (4)$$

式中: $\tilde{\mathbf{P}}^0 = [\tilde{\mathbf{p}}_1^0(t), \tilde{\mathbf{p}}_2^0(t), \dots, \tilde{\mathbf{p}}_{N_U}^0(t)] \in \mathbb{R}^{3 \times N_U}$; $\tilde{\mathbf{P}}^*(t) = [\tilde{\mathbf{p}}_1^*(t), \tilde{\mathbf{p}}_2^*(t), \dots, \tilde{\mathbf{p}}_{N_U}^*(t)] \in \mathbb{R}^{3 \times N_U}$,其中 $\tilde{\mathbf{p}}_j^m(t) = [x_j^m(t), y_j^m(t), 1]^T$, $m = \{0, *\}$, $j \in [1, N_U]$,为第 j 个UUV在 t 时刻的二

维齐次坐标; $A_1 \sim A_6$ 为 $\mathbf{A}(t)$ 中的时变参数; $\mathbf{A}(t) \in \mathbb{R}^{3 \times 3}$, 为二维齐次仿射变换矩阵, 其定义如下^[1]:

$$\mathbf{A}(t) = \lambda(t)\mathbf{A}_T(t)\mathbf{A}_R(t)\mathbf{A}_{Sh}(t)\mathbf{A}_S(t) \quad (5)$$

式中: λ 为整体缩放因子; $\mathbf{A}_T(t)$ 由平移参数 $t_x(t)$ 和 $t_y(t)$ 定义, 表示编队整体在二维平面的位移; $\mathbf{A}_R(t) = \mathbf{A}_R(\theta(t))$, 由角度 $\theta(t)$ 定义, 表示编队相对参考编队方向的旋转角度; $\mathbf{A}_{Sh}(t)$ 由 x 与 y 方向的倾斜参数 $sh_{xy}(t)$ 和 $sh_{yx}(t)$ 定义, 为编队产生的倾斜和变形; $\mathbf{A}_S(t)$ 由沿 x 与 y 轴的缩放参数 $s_x(t)$ 和 $s_y(t)$ 定义, 使编队分别沿 x 轴和 y 轴进行独立的缩放。

为简化规划问题的维度和复杂度, 将仿射变换简化为旋转、缩放和平移。定义规划的仿射状态为

$$\mathbf{q}(t) = [t_x(t), t_y(t), \theta(t), s_x(t), s_y(t)]^T \in \mathbb{R}^5 \quad (6)$$

规划算法的目标是找到一条从初始规划状态 $\mathbf{q}(0)$ 到目标规划状态 $\mathbf{q}(T)$ 的连续、可行轨迹 $\mathcal{T}_A = \{\mathbf{q}(t)|t \in [0, T]\}$, 其他仿射参数(整体缩放因子和剪切量)由任务定义并在规划阶段保持固定。因此, 式(3)和式(4)可以进一步表述成 $\mathbf{q}(t)$ 和 $\tilde{\mathbf{p}}^0$ 的仿射映射关系函数 $f_{af}(\cdot)$:

$$\tilde{\mathbf{P}}^*(t) = \mathbf{A}(\mathbf{q}(t))\tilde{\mathbf{P}}^0 = f_{af}(\mathbf{q}(t), \tilde{\mathbf{P}}^0) \quad (7)$$

$$\tilde{\mathbf{p}}_j^*(t) = \mathbf{A}(\mathbf{q}(t))\tilde{\mathbf{p}}_j^0 = f_{af}(\mathbf{q}(t), \tilde{\mathbf{p}}_j^0) \quad (8)$$

为确保生成的轨迹在复杂水域环境中安全、可行且高效, 这条仿射参数轨迹必须满足以下多重约束:

1) UUV 运动学约束, 即每个 UUV 的速度、加速度必须满足其动力学特性限制 ($\|\mathbf{V}_j\|_2 \leq V_{\max}$, $\|\mathbf{A}_j\|_2 \leq A_{\max}$);

2) 编队中任意 UUV 与环境障碍物的距离 $\{q_1, q_2, \dots, q_k\}$ 必须大于安全阈值 d_{safe} , 且编队内部个体间需保持最小安全间距, 以避免碰撞;

3) 仿射参数轨迹必须保证速度、加速度的连续性, 避免产生不可执行的突变。

所考虑的避障环境需符合实际岛屿的复杂非结构化环境, 因不同区域的可通行空间差异显著, 因此, 编队需根据当前位置环境特征动态调整编队形态, 以达到与环境特征的全局最优匹配。

综上, 多 UUV 编队规划任务由原本复杂的 N_U 个 UUV 规划路径的多体问题转换为求解仿射状态的演化轨迹问题。

2 编队全局规划方法设计

针对问题描述中的规划目标, 本文提出一种

包含前端路径搜索和后端轨迹优化的全局规划框架, 如图2所示。其中, 前端规划器采用一种改进的 RRT*算法, 该算法采用分阶段搜索方式, 在三维仿射参数空间(平移、旋转、独立轴缩放)快速找到一条连接起始与目标编队状态、并初步规避障碍物的离散仿射状态序列。针对前端输出的离散状态点, 后端采用基于 B 样条曲线的轨迹优化器, 通过最小化一个综合考虑轨迹平滑性、运动学可行性、碰撞安全性以及编队形态自适应性的目标函数, 生成一条连续、光滑且满足多重约束的最优仿射参数轨迹, 最终指导多 UUV 编队完成复杂的避障任务。因本文侧重于解决静态场景下的全局编队轨迹规划问题, 故采用占据栅格地图来表示无障碍区域。

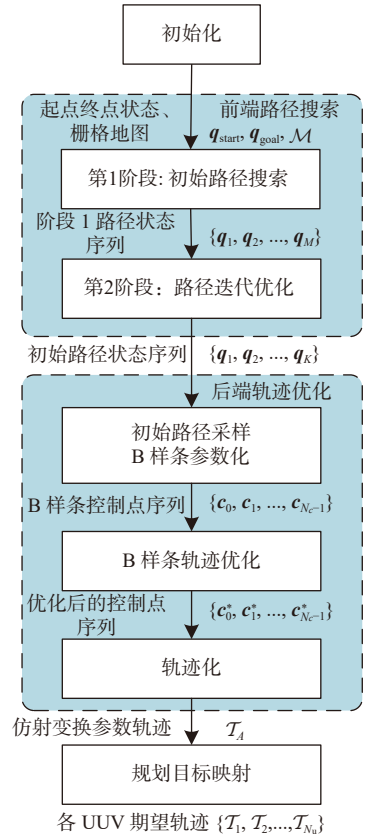


图2 全局规划方法流程图

Fig. 2 Flowchart of the global formation planning method

2.1 AT-RRT*算法

本节将在 RRT*算法的基础上进行改进, 设计一种基于仿射变换的 RRT*算法。该算法的核心思想是在仿射参数空间中进行搜索, 直接生成一系列描述编队平移、旋转和缩放变化的仿射状态点 $\{q_1, q_2, \dots, q_k\}$, 从而构成初始的编队路径。

2.1.1 扩展采样空间

传统的 RRT*在 \mathbb{R}^2 或者 $SE(2)$ 空间中通常为单机器人搜索路径点, 对于具有仿射编队结构的多

UVU 编队, 本文将采样空间扩展至包含式(6)的二维仿射参数空间:

$$Q = \mathbb{R}^2 \times SO(2) \times S \in \mathbb{R}^5 \quad (9)$$

式中: \mathbb{R}^2 部分对应仿射参数空间的平移参数 $t = (t_x, t_y)$, 其范围受限于环境空间边界 $t \in [t_{x,\min}, t_{x,\max}] \times [t_{y,\min}, t_{y,\max}]$; $SO(2)$ 对应旋转参数 θ , 通常参数化为 $\theta \in [-\pi, \pi]$; S 部分对应编队横向和纵向缩放因子 $s = (s_x, s_y)$, 其范围由允许的最小和最大缩放值界定, $s = [s_{x,\min}, s_{x,\max}] \times [s_{y,\min}, s_{y,\max}]$ 。

算法采用KD树数据结构在搜索过程(剪枝、重布线)中高效地查找邻近状态节点。然而, KD树是直接在原始的五维状态空间 Q 中使用标准欧氏距离进行邻近搜索, 并未考虑各维度的尺度差异。在本文中, 旋转和缩放量的搜索范围远小于位移量, 导致KD树在选择邻近状态节点时主要按照位移维度进行划分, 在旋转和缩放维度几乎考虑了搜索范围内的所有值, 因此在这几个维度上退化成了普通的数据结构, 导致搜索效率降低。针对这个问题, 本算法对KD树操作进行了加权表示, 即定义一组维度权重 $w = [w_x, w_y, w_\theta, w_{s_x}, w_{s_y}]$, 对于原始状态 $q = [t_x, t_y, \theta, s_x, s_y]^T$, 加权后的搜索状态表示为 $q_w = [w_x t_x, w_y t_y, w_\theta \theta, w_{s_x} s_x, w_{s_y} s_y]^T$ 。通过这种加权表示, 可以更加均衡地考虑不同维度在邻近搜索中的贡献, 避免因尺度差异导致搜索偏向某些维度, 从而提高KD树在多尺度状态空间中的划分效率和搜索精度。

2.1.2 两阶段搜索策略

为了在保证搜索效率的同时获得高质量的编队路径, AT-RRT*算法采用了分阶段搜索策略, 并分别设计不同的多目标的代价函数来引导搜索过程。

第1阶段: 初始路径搜索。此阶段的首要任务是尽快在复杂的五维状态空间 Q 中找到一条连接起始仿射状态 q_{start} 到目标仿射状态 q_{goal} 且全程无碰撞的路径。

为了加速这个过程, 在该阶段, 节点状态扩展类似于传统的RRT*算法, 只在位置分量 $t = (t_x, t_y)$ 维度进行探索。对于其他维度, 旋转分量 θ 设置为前进方向, $\theta_{\text{new}} = \text{dir}(t_{\text{cur}}, t_{\text{new}})$, 缩放分量固定设置为最小安全缩放比例量进行碰撞检测以提高搜索效率, $s_{\text{new}} = (s_{x,\min}, s_{y,\min})$ 。该阶段的KD树操作与传统算法相同, 只在位置分量维度进行查询、插入和删除。一旦找到一条连接起点和终点状态的节点通路 $\{q_1, q_2, \dots, q_M\}$, 算法就会切换至第2阶段, 并将这条通路上的所有节点进行加权操作插入到全状态的KD树中, 以供第2阶段查询, 具

体的算法流程见算法1。

算法1 AT-RRT*算法

Algorithm 1 AT-RRT* algorithm

Input: $q_{\text{start}}, q_{\text{goal}}$

Result: $\{q_1, q_2, \dots, q_K\}$

KDTree_{cur} ← KDTree_{2D};
reach_goal ← false

while ($i < i_{\text{max}}$ and $t < t_{\text{max}}$) **do**

if CurPhase == Phase1 **then**

$q_{\text{rand}} \leftarrow \text{Sample2DState}();$

else

$q_{\text{rand}} \leftarrow \text{SampleFullState}();$

$q_{\text{near}} \leftarrow \text{FindNearest}(\text{KDTree}, q_{\text{rand}});$

$q_{\text{new}} \leftarrow \text{Step}(q_{\text{near}}, q_{\text{rand}});$

if IsStateValid(q_{new}) **then**

$q_{\text{cur}} \leftarrow \text{ChooseParentAndRewire}(q_{\text{new}});$

KDTree_{cur}.Insert(q_{cur});

if IsNearGoal(q_{cur}) && IsLineFree($q_{\text{cur}}, q_{\text{goal}}$) **then**

$C \leftarrow q_{\text{cur}}.\text{Cost} + \text{Cost}(q_{\text{cur}}, q_{\text{goal}});$

if reach_goal == false **then**

reach_goal ← true;

changePhaseTo2();

KDTree_{cur} ← KDTree_{5D};

else

CheckandUpdateCost(C);

return tracebackFormGoal(q_{goal})

第2阶段: 路径迭代优化。在该阶段, 会在完整的五维状态空间 Q 进行节点搜索和KD树操作, 其主要目的是在已找到的第1阶段路径的基础上进行迭代优化, 得到高质量的初始全局路径。

RRT*算法通过比较不同连接路径的累积代价来选择父节点和执行重布线, 从而驱动路径向低成本优化。在该算法中, 2个阶段的代价函数相同, 通过综合考虑多个评价因素, 定义如下:

$$C = w_p c_p + w_{\Delta\theta} c_{\Delta\theta} + w_{\theta_i} c_{\theta_i} + w_{\Delta s} c_{\Delta s} + w_{s_{\text{std}}} c_{s_{\text{std}}} \quad (10)$$

式中: $w_p, w_{\Delta\theta}, w_{\theta_i}, w_{\Delta s}, w_{s_{\text{std}}}$ 为各项对应的权重; c_p 为位置移动代价, 该项在整条通路的累计代价近似为路径长度; $c_{\Delta\theta}$ 为旋转分量变化成本, 用于惩罚剧烈的旋转; c_{θ_i} 为前向偏差成本, 以鼓励编队旋转角度与移动方向一致; $c_{\Delta s}$ 为缩放变化成本, 用于惩罚剧烈的缩放变化; $c_{s_{\text{std}}}$ 为与标准编队缩放偏差的成本, 该项鼓励目标编队缩放量接近标称编队构型的缩放量, 也即标准缩放量 $s_{\text{std}} = (1, 1)$ 。

在优化阶段, 算法利用该代价函数, 通过不断添加新节点并寻找更优的连接(重布线), 迭代降低从 q_{start} 到 q_{goal} 的累积路径成本, 直至满足最大迭

次数、时间和成本收敛等终止条件。

2.1.3 混合采样策略

为了加速算法收敛,并根据不同阶段的搜索目标,在有效引导搜索树向目标区域扩展的同时保持对整个状态空间的探索能力,AT-RRT*采用4种采样模式的混合采样策略,如图3所示。

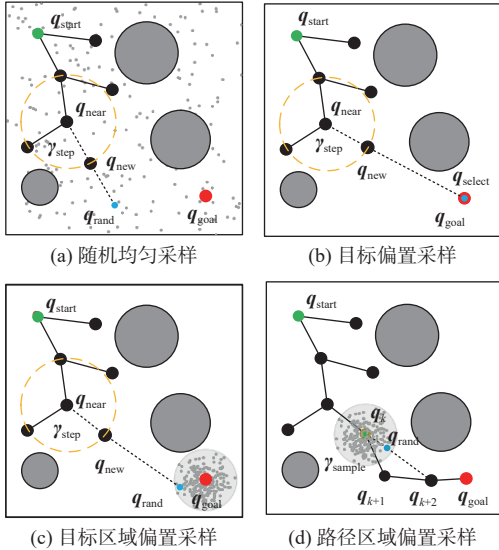


图3 4种采样模式
Fig. 3 Four sampling modes

由图3(a)可知,随机均匀采样允许在状态空间 Q 内均匀随机地生成采样点;目标偏置采样直接选择目标状态节点 q_{goal} 作为采样点;目标区域偏置采样在目标状态 q_{goal} 周围的一个邻域进行高斯分布采样;路径区域偏置采样只会第2阶段中启用,是该阶段的主要采样方式,会在已有路径点中随机选择一个状态点 q_k ,并在周围的一个邻域使用高斯采样得到 q_{rand} ,在通过碰撞检测后直接作为 q_{new} 。在这种情况下,如果 q_{rand} 得到的成本更少,例如如图3(d)中 q_{rand} 和 q_k, q_{k+2} 构成的路径长度比之前的路径更短,则后续重布线操作将使用 q_{rand} 代替 q_{k+1} 。

总的来说,采样方式(a)保证算法有对整个状态空间的探索能力,方式(b)和方式(c)引导搜索树向目标区域扩展,方式(d)则加速了第2阶段的路径成本收敛。在实际算法中,4种采样方式会按照预设的概率进行选择,其中第1阶段侧重于探索,方式(a)的概率更大,阶段2侧重于迭代已有路径成本,方式(d)的概率更大。综上所述,采用4种采样模式的混合采样策略能够有效平衡全局探索和局部优化,从而在不同搜索阶段实现高效且有针对性的路径搜索,最终提升算法的整体性能与收敛速度。

2.1.4 自适应采样和扩展参数

图3中,前3种采样方式在得到 q_{rand} 后, q_{near}

会朝着 q_{rand} 的方向生成 q_{new} ,扩展的步长为 γ_{step} 。为了鼓励算法在空旷的区域进行激进的步进以加速扩展速度,需在障碍物复杂、狭窄的区域进行保守的步进以提高采样点碰撞检测的成功率,加快算法整体的运行时间。其中,步进长度会根据障碍物密度 d_q 进行调整,并且因为不同维度的尺度不同,步长也需各自定义, $\gamma_{step} = [\gamma_t, \gamma_\theta, \gamma_s]$ 和 d_q 的定义如下所示。

$$d_q = d(q, R) = \begin{cases} 0, & E(q) \geq R \\ 1 - E(q)/R, & 0 \leq E(q) \leq R \end{cases} \quad (11)$$

式中: $E(q)$ 为采样点 q 位置分量 (t_x, t_y) 的欧几里得距离场(Euclidean signed distance field, ESDF)值,表示该点距离最近障碍物的距离; R 为设定的查询范围。 $d_q \in [0, 1]$,其中0表示查询范围内无障碍,1表示 (t_x, t_y) 点有障碍物。

$$\gamma_i(d_q) = \gamma_{i,max}^{step} \times (\beta_i^{step} + (1 - \beta_i^{step}) \times (1 - d_q)) \quad (12)$$

式中: $i \in \{t, \theta, s\}$,分别为位移、旋转和缩放的3种分量; $\gamma_{i,max}^{step}$ 为分量 i 的最大步长距离; β_i^{step} 为分量的调制因子; $d_q = d(q_k, \gamma_{i,max}^{step})$ 。 $\gamma_i(d_q)$ 会随着障碍物密度的增大而减小。

同样,对于路径区域采样方式,其效果很大程度上依赖于如何在选定的路径点 q_k 周围生成新的采样点。因此,算法对采样半径 γ_{sample} 也进行了自适应调整,在环境拥挤的区域,采样范围较小,以进行更精细、更安全的局部探索;反之,采样范围会相对增大,以鼓励更大胆的探索,增加采样的成功率。在算法中,采样半径是通过调节高斯采样的标准差来实现的,即 $\gamma_{sample} = [\sigma_t, \sigma_\theta, \sigma_s]$,定义如下:

$$\sigma_i(d_q) = \sigma_{i,max} \times (\beta_i^{sample} + (1 - \beta_i^{sample}) \times (1 - d_q)) \quad (13)$$

式中: $d_q = d(q_k, \sigma_{t,max})$; $\sigma_{i,max}$ 为分量 i 的高斯采样最大标准差;参数 β_i^{sample} 的定义与式(12)同,但具体的取值不同。

2.2 仿射参数优化器

本节将在AT-RRT*得到离散仿射状态序列的基础上进行优化,旨在寻找一个时序变化的仿射参数序列,以使整个编队在满足运动学约束、避免碰撞的同时,尽可能保持标称编队构型并使轨迹平滑。

2.2.1 基于B样条的轨迹表示

为了将离散的仿射状态序列转化为适合优化的初始连续轨迹 \mathcal{T}_A^{init} ,本节使用均匀B样条曲线 Φ 进行轨迹参数化表示,该曲线是一个分段多项

式,由次数 p_b 、结点间距 Δt 以及 N_c 个控制点 $\{\mathbf{c}_k\}_{k=0}^{N_c-1}$ 唯一确定。本文选择 $p_b = 3$,即三次均匀B样条,且 $\mathbf{c}_k \in \mathbb{R}^5$ 也在仿射参数空间,因此整个轨迹的时间跨度为 $(N_c - 3)\Delta t$ 。

因由RRT类算法生成的路径点分布可能不均匀,故首先对离散的仿射状态序列进行重采样,得到 $N_c - 1$ 个五维仿射状态点序列,并作为插值点结合边界条件(起始和终止时刻的速度为0)求解确定一组B样条曲线的控制点序列 $\mathbf{C} = \{\mathbf{c}_0, \mathbf{c}_1, \dots, \mathbf{c}_{N_c-1}\}$ 。对于均匀三次B样条,第 k 个时间步长的轨迹约束点 \mathbf{q}_k 可以由其相邻的3个控制点通过下式计算得到^[6]:

$$\mathbf{q}_k = \mathbf{q}(t_k) = \frac{1}{6}(\mathbf{c}_k + 4\mathbf{c}_{k+1} + \mathbf{c}_{k+2}) \quad (14)$$

式中, $k = 1, 2, \dots, N_c - 2$ 。在轨迹约束点 \mathbf{q}_k 处,第 j 个UUV的目标位置 $\mathbf{p}_{k,j}^* = [x_{k,j}^*, y_{k,j}^*]$ 由式(8)中的仿射映射关系 $f_{\text{af}}(\cdot)$ 计算得到:

$$\tilde{\mathbf{p}}_{k,j}^* = \begin{bmatrix} \mathbf{p}_{k,j}^* \\ 1 \end{bmatrix} = f_{\text{af}}(\mathbf{q}_{k,j}, \tilde{\mathbf{p}}_j^0) \quad (15)$$

2.2.2 优化问题和目标函数

优化变量为 $N_c - 2p_b + 1$ 个控制点的子集 $\{\mathbf{c}_{p_b}, \mathbf{c}_{p_b+1}, \dots, \mathbf{c}_{N_c-p_b}\}$,由于边界约束点 \mathbf{q}_0 和 \mathbf{q}_N 由前 p_b 个和最后 p_b 个控制点确定^[7],因此无需修改。优化问题表述如下:

$$\min \mathcal{J} = \lambda_s \mathcal{J}_s + \lambda_f \mathcal{J}_f + \lambda_c \mathcal{J}_c + \lambda_{sc} \mathcal{J}_{sc} \quad (16)$$

式中: \mathcal{J}_s 表示平滑性代价; \mathcal{J}_f 表示可行性代价; \mathcal{J}_c 表示碰撞代价; \mathcal{J}_{sc} 表示自适应编队缩放代价; $\lambda_s, \lambda_f, \lambda_c, \lambda_{sc}$ 分别为对应的权重。

为了简化导数的计算,UUV的速度 \mathbf{V}_{k_j} 、加速度 \mathbf{A}_{k_j} 、加加速度 \mathbf{J}_{k_j} 分别使用第1,2,3阶有限差分进行近似:

$$\mathbf{V}_{k_j} \approx \frac{\mathbf{p}_{k+1,j}^* - \mathbf{p}_{k,j}^*}{\Delta t} \quad (17)$$

$$\mathbf{A}_{k_j} \approx \frac{\mathbf{V}_{k+1,j} - \mathbf{V}_{k_j}}{\Delta t} \approx \frac{\mathbf{p}_{k+2,j}^* - 2\mathbf{p}_{k+1,j}^* + \mathbf{p}_{k,j}^*}{\Delta t^2} \quad (18)$$

$$\mathbf{J}_{k_j} \approx \frac{\mathbf{A}_{k+1,j} - \mathbf{A}_{k_j}}{\Delta t} \approx \frac{\mathbf{p}_{k+3,j}^* - 3\mathbf{p}_{k+2,j}^* + 3\mathbf{p}_{k+1,j}^* - \mathbf{p}_{k,j}^*}{(\Delta t)^3} \quad (19)$$

平滑性代价 \mathcal{J}_s 惩罚编队中每个UUV在其轨迹上的加速度和加加速度的大小,使得轨迹变化更加连续和平滑,定义如下:

$$\mathcal{J}_s = \sum_{k=1}^{N_c-4} \sum_{j=1}^{N_U} \|\mathbf{A}_{k_j}\|_2^2 + \sum_{k=1}^{N_c-5} \sum_{j=1}^{N_U} \|\mathbf{J}_{k_j}\|_2^2 \quad (20)$$

可行性代价 \mathcal{J}_f 确保生成的轨迹对于每个UUV

都是可执行的,本文对每个约束点 \mathbf{q}_k 上的每个UUV的速度 \mathbf{V}_{k_j} 和加速度 \mathbf{A}_{k_j} 进行约束,惩罚超过预设的 V_{\max} 和 A_{\max} 的部分:

$$\begin{aligned} \mathcal{J}_f &= \sum_{k=1}^{N_c-3} \sum_{j=1}^{N_U} F_f(\mathbf{V}_{k_j}) + \sum_{k=1}^{N_c-4} \sum_{j=1}^{N_U} F_f(\mathbf{A}_{k_j}) \\ F_f(\mathbf{V}_{k_j}) &= \max(0, \|\mathbf{V}_{k_j}\|_2^2 - V_{\max}^2)^2 \\ F_f(\mathbf{A}_{k_j}) &= \max(0, \|\mathbf{A}_{k_j}\|_2^2 - A_{\max}^2)^2 \end{aligned} \quad (21)$$

碰撞代价 \mathcal{J}_c 为了避免编队中的每个UUV与环境中的障碍物发生碰撞,会在每个约束点 \mathbf{q}_k 上惩罚每个与障碍物的距离小于安全阈值 d_{safe} 的UUV:

$$\begin{aligned} \mathcal{J}_c &= \sum_{k=1}^{N_c-2} \sum_{j=1}^{N_U} F_c(\mathbf{p}_{k_j}^*) \\ F_c(\mathbf{p}_{k_j}^*) &= \max(0, d_{\text{safe}} - E(\mathbf{p}_{k_j}^*))^2 \end{aligned} \quad (22)$$

式中, $E(\mathbf{p}_{k_j}^*)$ 为 $(x_{k_j}^*, y_{k_j}^*)$ 点处的ESDF值。

设计自适应编队缩放代价 \mathcal{J}_{sc} 的目标是平衡编队构型保持与环境适应性需求,即在较为开阔、无障碍的空间,希望编队尽可能保持其预设的标称构型的缩放量($s_{x,\text{std}} = s_{y,\text{std}} = 1.0$);而在障碍物密集或需要穿越狭窄通道时,编队构型可以缩小以适应环境。此外,为了防止个体间的碰撞, \mathcal{J}_{sc} 会惩罚小于缩放量的最小安全阈值($s_{x,\text{min}}, s_{y,\text{min}}$),即对于约束点 \mathbf{q}_k ,其 x 方向缩放量的分段惩罚项如式(23)所示, y 方向的同理。

$$C_x(s_{x,k}, \alpha_k) = \begin{cases} \alpha_{\min}(s_{x,\text{min}} - s_{x,k})^2, & s_{x,k} < s_{x,\text{min}} \\ \alpha_k(1.0 - s_{x,k})^2, & s_{x,k} \geq s_{x,\text{min}} \end{cases} \quad (23)$$

式中: α_{\min} 为缩放量低于最小安全阈值的固定惩罚系数,数值较大; $\alpha_k = \alpha(d_{q_k})$,为与当前轨迹约束点的障碍物环境有关的惩罚系数,定义如下:

$$\alpha(d_{q_k}) = \alpha_{\max} \times (\beta_s + (1 - \beta_s) \times (1 - d_{q_k})) \quad (24)$$

式中: $d_{q_k} = d(\mathbf{q}_k, \mathbf{R}_{s,\text{max}})$,为 \mathbf{q}_k 约束点的位置点 $(t_{x,k}, t_{y,k})$ 的障碍物密度,其中 $\mathbf{R}_{s,\text{max}}$ 为 x 和 y 轴缩放为最大缩放时($s_{x,\text{max}}, s_{y,\text{max}}$)的外接圆半径,如图4所示; β_s 为调制因子, $\alpha_{\min} = \alpha_{\max} \times \beta_s$ 。

自适应编队缩放代价 \mathcal{J}_s 由所有轨迹点在 x, y 方向惩罚的加权求和得到:

$$\mathcal{J}_{sc} = \sum_{k=1}^{N_c-2} (w_x C_x(s_{x,k}, \alpha_k) + w_y C_y(s_{y,k}, \alpha_k)) \quad (25)$$

式中, w_x 和 w_y 为对应的权重系数。

2.2.3 梯度计算与优化求解

为了能够高效地求解上述无约束优化问题,本文使用L-BFGS算法,该算法是基于梯度的数

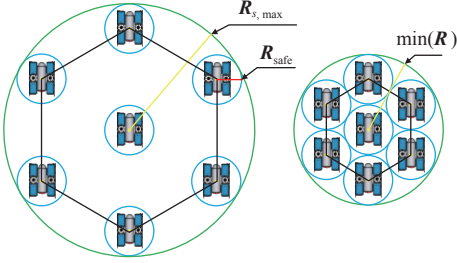


图4 最大缩放和最小缩放尺度时的外接圆

Fig. 4 Circumscribed circles at maximum and minimum scaling factors

值优化方法,通过存储有限的历史梯度信息来近似海森矩阵的逆,从而有效确定搜索方向。因此,要使用该算法就需计算目标函数 \mathcal{J} 相对于优化变量 \mathbf{c}_m 的梯度 $\partial\mathcal{J}/\partial\mathbf{c}_m$ 。梯度可以通过轨迹约束点 \mathbf{q}_k 反向传播至控制点,可见,总代价 \mathcal{J} 对单个控制点 \mathbf{c}_m 的梯度,为通过所有受 \mathbf{c}_m 影响的轨迹点 \mathbf{q}_k 对 \mathbf{c}_m 求导的贡献之和:

$$\frac{\partial\mathcal{J}}{\partial\mathbf{c}_m} = \sum_k \left(\frac{\partial\mathcal{J}}{\partial\mathbf{q}_k} \right)^T \frac{\partial\mathbf{q}_k}{\partial\mathbf{c}_m} \quad (26)$$

由式(14)可知,上式只有当 $k=m, m-1, m-2$ 时 \mathbf{q}_k 才会对 \mathbf{c}_m 的梯度有贡献,因此,只有当上述3项为非零项时,将 $k \in \{m, m-1, m-2\}$ 代入式(14)推导出的偏导数才为非零项。得到代价关于轨迹约束点的梯度关系如下:

$$\frac{\partial\mathcal{J}}{\partial\mathbf{c}_m} = \left(\frac{1}{6} \frac{\partial\mathcal{J}}{\partial\mathbf{q}_{m-2}} + \frac{4}{6} \frac{\partial\mathcal{J}}{\partial\mathbf{q}_{m-1}} + \frac{1}{6} \frac{\partial\mathcal{J}}{\partial\mathbf{q}_m} \right)^T \quad (27)$$

此外,由于平滑、可行和避碰代价中涉及第 j 个UUV在时间步 k 的期望位置点 $\mathbf{p}_{k,j}^*$,因此,还需要计算 $\mathbf{p}_{k,j}^*$ 关于 \mathbf{q}_m 的梯度 $\partial\mathbf{p}_{k,j}^*/\partial\mathbf{q}_m$,根据链式求导法则。当 $m=k$ 时,该项不为0,结合式(3)、式(6)和式(7),偏导数的雅可比矩阵表示如下:

$$\frac{\partial\mathbf{p}_{k,j}^*}{\partial\mathbf{q}_m} = \begin{bmatrix} \frac{\partial x_{k,j}^*}{\partial t_m} & \frac{\partial x_{k,j}^*}{\partial t_m} & \frac{\partial x_{k,j}^*}{\partial \theta_m} & \frac{\partial x_{k,j}^*}{\partial s_{x,m}} & \frac{\partial x_{k,j}^*}{\partial s_{y,m}} \\ \frac{\partial y_{k,j}^*}{\partial t_m} & \frac{\partial y_{k,j}^*}{\partial t_m} & \frac{\partial y_{k,j}^*}{\partial \theta_m} & \frac{\partial y_{k,j}^*}{\partial s_{x,m}} & \frac{\partial y_{k,j}^*}{\partial s_{y,m}} \end{bmatrix} \quad (28)$$

下面,开始推导前3项代价关于 $\mathbf{p}_{k,j}^*$ 的梯度计算方式。平滑性代价为加速度项 \mathcal{J}_A^s 与加加速度项 \mathcal{J}_j^s 的代价之和, $\mathcal{J}_s = \sum_k \sum_j \mathcal{J}_{A,k,j}^s + \sum_k \sum_j \mathcal{J}_{j,k,j}^s$ 。以加速度项为例,其对轨迹约束点 \mathbf{q}_m 的梯度根据加速度的定义式(18)和式(20),可以使用链式法则计算:

$$\frac{\partial\mathcal{J}_{A,k,j}^s}{\partial\mathbf{q}_m} = \frac{\partial\mathcal{J}_{A,k,j}^s}{\partial\mathbf{A}_{k,j}} \cdot \frac{\partial\mathbf{A}_{k,j}}{\partial\mathbf{q}_m} = \frac{2\mathbf{A}_{k,j}^T}{\Delta t^2} \left(\frac{\partial\mathbf{p}_{k+2,j}^*}{\partial\mathbf{q}_m} - 2\frac{\partial\mathbf{p}_{k+1,j}^*}{\partial\mathbf{q}_m} + \frac{\partial\mathbf{p}_{k,j}^*}{\partial\mathbf{q}_m} \right) \quad (29)$$

式中,当且仅当 $m \in \{k, k+1, k+2\}$ 时梯度非零。加加速度项的梯度 $\partial\mathcal{J}_{j,k,j}^s/\partial\mathbf{q}_m$ 与 $\partial\mathcal{J}_{A,k,j}^s/\partial\mathbf{q}_m$ 的形式相同。

可行性代价的梯度与平滑性代价的形式相似,以加速度项的梯度 $\partial\mathcal{J}_{v,k,j}^r/\partial\mathbf{q}_m$ 为例,

$$\frac{\partial\mathcal{J}_{v,k,j}^r}{\partial\mathbf{q}_m} = \frac{\partial\mathcal{J}_{v,k,j}^r}{\partial\mathbf{V}_{k,j}} \cdot \frac{\partial\mathbf{V}_{k,j}}{\partial\mathbf{q}_m} = \frac{\partial F_t(\mathbf{V}_{k,j})}{\partial\mathbf{V}_{k,j}} \cdot \left(\frac{\partial\mathbf{p}_{k+1,j}^*}{\partial\mathbf{q}_m} - \frac{\partial\mathbf{p}_{k,j}^*}{\partial\mathbf{q}_m} \right) \quad (30)$$

式中,当且仅当 $m \in \{k, k+1\}$ 时梯度非零。加加速度项 $\partial\mathcal{J}_{v,k,j}^r/\partial\mathbf{q}_m$ 与 $\partial\mathcal{J}_{A,k,j}^s/\partial\mathbf{q}_m$ 的形式相同。

碰撞代价的梯度 $\partial\mathcal{J}_{k,j}^c/\partial\mathbf{q}_m$ 与 $\mathbf{p}_{k,j}^*$ 处的ESDF距离值有关:

$$\frac{\partial\mathcal{J}_{k,j}^c}{\partial\mathbf{q}_m} = \frac{\partial F_c(\mathbf{p}_{k,j}^*)}{\partial E} \cdot \tilde{N}E(\mathbf{p}_{k,j}^*)^T \cdot \frac{\partial\mathbf{p}_{k,j}^*}{\partial\mathbf{q}_m} \quad (31)$$

式中:当 $m=k$ 且 $d_{\text{safe}} - E(\mathbf{p}_{k,j}^*) > 0$ 时梯度非零; $\tilde{N}E(\mathbf{p}_{k,j}^*)$ 为ESDF距离场在 $\mathbf{p}_{k,j}^*$ 点的梯度信息,通过双线性插值计算得到。

自适应编队缩放梯度只与状态约束点中对应方向的缩放量有关,等于 x 和 y 方向的加权梯度之和: $\partial\mathcal{J}_k^{\text{sc}}/\partial\mathbf{q}_m = w_x \partial C_{x,k}/\partial\mathbf{q}_m + w_y \partial C_{y,k}/\partial\mathbf{q}_m$ 。以 x 方向为例,其对第 m 个状态约束点 \mathbf{q}_m 的梯度如下:

$$\frac{\partial C_x(s_{x,k}, \alpha_k)}{\partial\mathbf{q}_m} = \frac{\partial C_x}{\partial s_{x,k}} \frac{\partial s_{x,k}}{\partial\mathbf{q}_m} + \frac{\partial C_x}{\partial \alpha_{k,j}} \frac{\partial \alpha_{k,j}}{\partial\mathbf{q}_m} \quad (32)$$

式中,当且仅当 $m=k$ 时梯度不为0,各项的具体公式如下:

$$\frac{\partial C_x}{\partial s_{x,k}} = \begin{cases} -2\alpha_{\min}(s_{x,\min} - s_{x,k}), & s_{x,k} < s_{x,\min} \\ -2\alpha_k(1.0 - s_{x,k}), & s_{x,k} \geq s_{x,\min} \end{cases} \quad (33)$$

$$\frac{\partial s_{x,k}}{\partial\mathbf{q}_m} = (0 \ 0 \ 0 \ 1 \ 0) \quad (34)$$

$$\frac{\partial C_x}{\partial \alpha_k} = \begin{cases} 0, & s_{x,k} < s_{x,\min} \\ (1.0 - s_{x,k})^2, & s_{x,k} \geq s_{x,\min} \end{cases} \quad (35)$$

$$\frac{\partial \alpha_k}{\partial\mathbf{q}_m} = -\alpha_{\max}(1 - \beta_{\text{sc}}) \nabla E((t_{m,x}, t_{m,y})) \quad (36)$$

式(32)中,障碍物密度相对于约束点的梯度与 $E(\mathbf{q}_k) = E((t_x, t_y))$ 有关, $\partial C_y/\partial\mathbf{q}_m$ 与式(32)~式(36)的推导过程相同,不同的是 $\partial s_{y,k}/\partial\mathbf{q}_m = (0 \ 0 \ 0 \ 0 \ 1)$ 。

3 湖上试验

3.1 试验设置

为了验证所提方法的有效性,在上海交通大学致远湖进行了湖上试验,试验设备如图5所示。由于试验条件有限,采用3台BlueROV2作为编队中的部分UUV,剩余的UUV则作为虚拟UUV在地面站电脑上进行仿真实验。实际UUV的位

姿由多普勒速度仪(DVL)、惯性测量单元(IMU)和全球导航卫星系统(GNSS)获得,其中GNSS用于获取地面站和3个真实UUV初始位置的经纬度,地面站为NED坐标系原点。任务开始后,通过DVL进行航位推演,虚拟UUV的位姿由仿真平台确定。标称编队如图6所示,其中绿色编号为实际UUV,黄色编号为虚拟UUV。实际和仿真UUV的轨迹跟踪控制器全部使用串级PID控制器。算法中,可行性约束中的最大速度和最大加速度项分别设置为 $V_{max} = 0.3 \text{ m/s}$, $A_{max} = 1.0 \text{ m/s}^2$ 。

标系、障碍物区域、起始和目标状态的信息。其中,MAP坐标系下 q_{start} 和 q_{goal} 的具体值如下:

$$q_{start} = [0.5 \quad -11.23 \quad 1.57 \quad 1.0 \quad 1.0]^T$$

$$q_{goal} = [3.5 \quad 9.25 \quad 1.57 \quad 1.0 \quad 1.0]^T$$

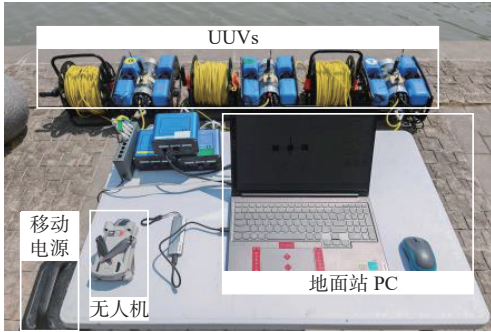


图5 试验设备
Fig. 5 Experimental equipment

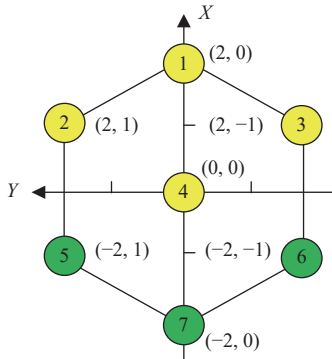


图6 标称编队构型
Fig. 6 Nominal formation configuration

图7所示为栅格化的试验地图和ESDF距离场的可视化。图8给出了地图坐标系和NED坐

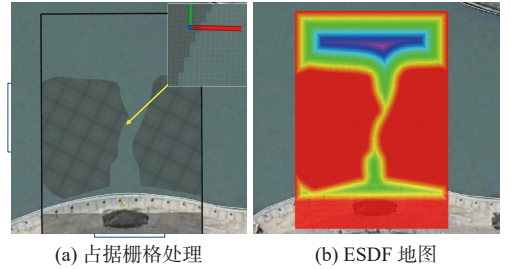


图7 栅格地图与ESDF地图
Fig. 7 Grid map and ESDF map

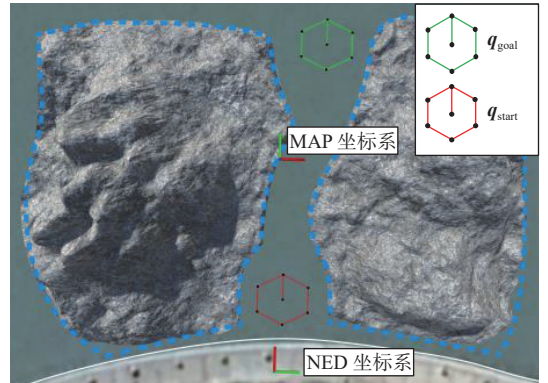


图8 地图和规划任务设置
Fig. 8 Map and planning mission setup

3.2 试验结果

试验过程如图9所示,图中展示了湖面和软件平台在 $t = 0, 33, 72, 115 \text{ s}$ 共4个时刻时UUV的运动位置。从图中可以看出,编队UUV通过缩放、旋转以及调整整体的位置,通过了狭窄的水道,并在较为宽阔的区域保持了标称编队的缩放尺度。

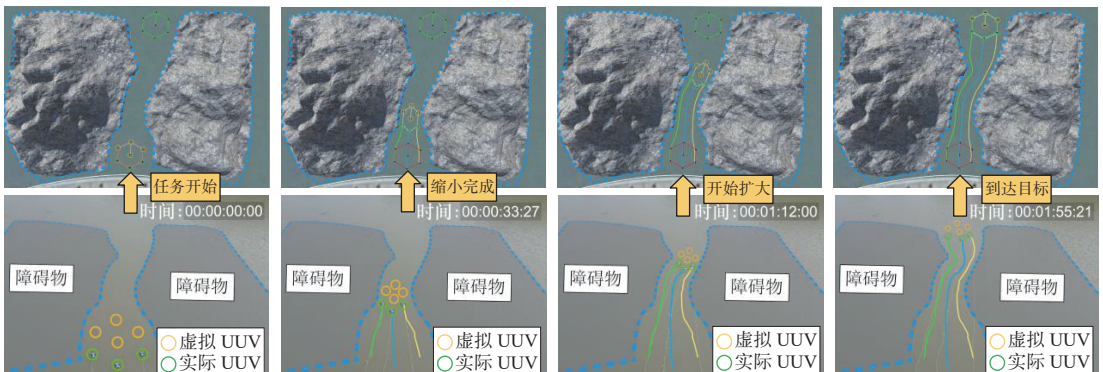


图9 试验过程关键帧
Fig. 9 Keyframes of the experimental process

图10和图11给出了实际UUV的轨迹跟踪误差。由图可见,其整体跟随误差较小,位置误

差小于 0.25 m ,艏向角误差小于 10° ,这说明算法规划的轨迹具有较高的可执行性、良好的平滑性

和连续性。此外,图 12 和表 1 还给出了实际 UUV 在试验过程中的速度 $\|V\|_2$ 和加速度 $\|A\|_2$ 曲线与状态量(平均值、标准差、绝对值最大值),从中可以看出,速度和加速度均满足任务设置的可行性约束。湖上试验的视频见文献 [18]。

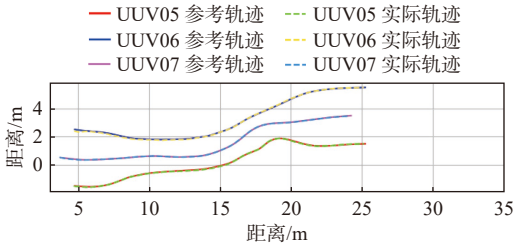


图 10 MAP 坐标系下实际 UUV 的期望轨迹与实际轨迹
Fig. 10 Desired trajectories and actual trajectories of the actual UUVs in the MAP coordinate system

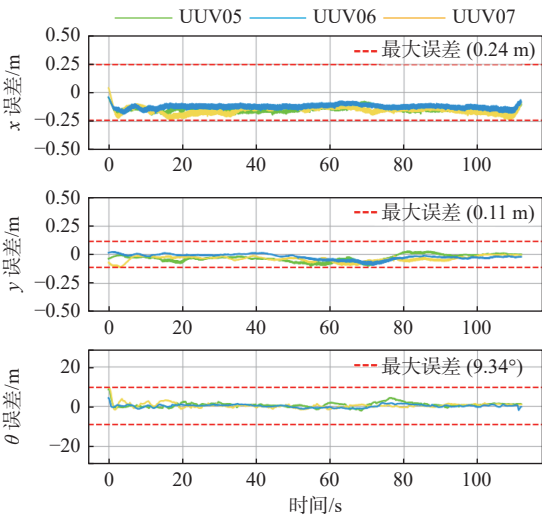


图 11 实际 UUV 的位姿跟踪误差
Fig. 11 Pose tracking errors of the actual UUVs

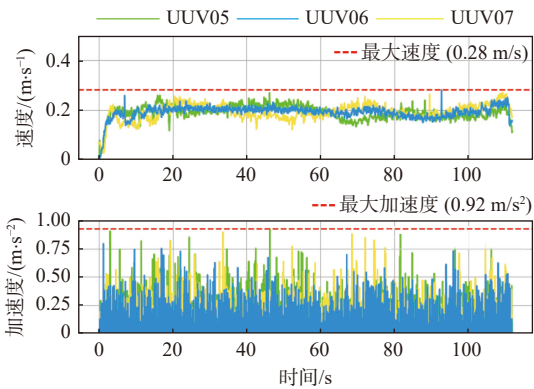


图 12 实际 UUV 的速度和加速度曲线
Fig. 12 Speed and acceleration curves of the actual UUVs

4 结 语

本文针对多 UUV 编队系统在复杂水下环境中执行协同任务时面临的编队保持与灵活避障难以同时实现的问题,提出了一种基于仿射变换的编队全局路径规划方法。该方法的核心思想是将编队整体的几何形态控制与路径规划问题映射到

表 1 实际 UUV 的速度和加速度状态量

Table 1 State quantities of speed and acceleration for the actual UUVs

UUV	参数	属性	数值
UUV05	速度/(m·s ⁻¹)	平均值	0.188 9
		标准差	0.030 3
		最大值	0.267 8
	加速度/(m·s ⁻²)	平均值	0.092 0
		标准差	0.149 7
		最大值	0.923 8
UUV06	速度/(m·s ⁻¹)	平均值	0.190 3
		标准差	0.033 0
		最大值	0.269 5
	加速度/(m·s ⁻²)	平均值	0.092 0
		标准差	0.145 9
		最大值	0.895 5
UUV07	速度/(m·s ⁻¹)	平均值	0.189 9
		标准差	0.026 2
		最大值	0.279 7
	加速度/(m·s ⁻²)	平均值	0.089 0
		标准差	0.139 2
		最大值	0.073 9

仿射参数空间中进行求解,从而实现从高维多体协同到低维参数优化的有效转化。算法的前端是两阶段的路径搜索,同时兼顾高效率的初始路径搜索和高质量的路径迭代优化;后端利用 B 样条优化平滑仿射参数轨迹,并进行约束惩罚,兼顾编队保持与避障需求。通过虚实结合的 UUV 混合编队湖上试验,对所提方法的有效性和实用性进行了验证。试验结果表明,该方法能够引导多 UUV 编队在模拟的复杂障碍环境中实现平滑、连续的队形变换从而安全通过狭窄通道,同时在开阔水域恢复并保持期望队形。

本研究可为提升 UUV 编队系统在复杂环境下的自主导航能力、环境适应性以及任务成功率等提供可行的解决方案,为水下搜救、资源勘探、环境监测等实际应用场景中多 UUV 系统的部署和高效作业提供可能。但是,目前本文方法还未完全考虑实际多 UUV 应用场景中面临的问题,未来可以进一步研究并提高算法在海流干扰和声学通信延迟方面的影响。

参考文献:

[1] DAS B, SUBUDHI B, PATI B B. Cooperative formation control of autonomous underwater vehicles: an overview[J]. International Journal of Automation and Computing, 2016, 13(3): 199–225.
[2] MILLÁN P, ORIHUELA L, JURADO I, et al. Forma-

- tion control of autonomous underwater vehicles subject to communication delays[J]. *IEEE Transactions on Control Systems Technology*, 2014, 22(2): 770–777.
- [3] SAHU B K, SUBUDHI B. Flocking control of multiple AUVs based on fuzzy potential functions[J]. *IEEE Transactions on Fuzzy Systems*, 2018, 26(5): 2539–2551.
- [4] FANG Z, JIANG D, HUANG J, et al. Autonomous underwater vehicle formation control and obstacle avoidance using multi-agent generative adversarial imitation learning[J]. *Ocean Engineering*, 2022, 262: 112182.
- [5] YANG Y, XIAO Y, LI T S. A survey of autonomous underwater vehicle formation: performance, formation control, and communication capability[J]. *IEEE Communications Surveys & Tutorials*, 2021, 23(2): 815–841.
- [6] VAN PARYS R, PIPELEERS G. Distributed MPC for multi-vehicle systems moving in formation[J]. *Robotics and Autonomous Systems*, 2017, 97: 144–152.
- [7] QUAN L, YIN L J, ZHANG T R, et al. Robust and efficient trajectory planning for formation flight in dense environments[J]. *IEEE Transactions on Robotics*, 2023, 39(6): 4785–4804.
- [8] TANG Q R, ZHANG Y, YU F C, et al. An obstacle avoidance approach based on system outlined rectangle for cooperative transportation of multiple mobile manipulators[C]//2018 IEEE International Conference on Intelligence and Safety for Robotics (ISR). Shenyang: IEEE, 2018: 533–538.
- [9] ALONSO-MORA J, BAKER S, RUS D. Multi-robot formation control and object transport in dynamic environments via constrained optimization[J]. *The International Journal of Robotics Research*, 2017, 36(9): 1000–1021.
- [10] 施宇. 复杂环境 AUV 集群自组织编队协同及避障控制[D]. 长春: 吉林大学, 2022.
- SHI Y. Research on self-organized formation cooperation and obstacle avoidance control of multiple AUVs in complex environment[D]. Changchun: Jilin University, 2022 (in Chinese).
- [11] PANG W, ZHU D Q, SUN C Y. Multi-AUV formation reconfiguration obstacle avoidance algorithm based on affine transformation and improved artificial potential field under ocean currents disturbance[J]. *IEEE Transactions on Automation Science and Engineering*, 2024, 21(2): 1469–1487.
- [12] ZHAO S Y. Affine formation maneuver control of multi-agent systems[J]. *IEEE Transactions on Automatic Control*, 2018, 63(12): 4140–4155.
- [13] YAO M Q, YU L, GUO K, et al. Distributed two-layered leader–follower affine formation control for multiple AUVs in 3-D space[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2025, 55(1): 85–95.
- [14] 姚明启, 于林, 郭凯, 等. 多自主水下航行器分布式三维仿射编队机动控制[J]. *中国舰船研究*, 2025, 20(1): 360–368.
- YAO M Q, YU L, GUO K, et al. Distributed three-dimensional affine formation maneuver control for multiple autonomous underwater vehicles[J]. *Chinese Journal of Ship Research*, 2025, 20(1): 360–368 (in Chinese).
- [15] 陈盼召, 郭凯, 葛彤, 等. 外部扰动下水下无人航行器鲁棒快速轨迹跟踪控制[J]. *中国舰船研究*, 2025, 20(6): 294–303.
- CHEN P Z, GUO K, GE T, et al. Robust and fast trajectory tracking control of UUVs with external disturbances[J]. *Chinese Journal of Ship Research*, 2025, 20(6): 294–303 (in Chinese).
- [16] LIN Y, LIU Y, ZHANG P P, et al. Safety-first tracker: a trajectory planning framework for omnidirectional robot tracking[C]//2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Abu Dhabi: IEEE, 2024: 5416–5423.
- [17] ZHOU B Y, GAO F, WANG L Q, et al. Robust and efficient quadrotor trajectory generation for fast autonomous flight[J]. *IEEE Robotics and Automation Letters*, 2019, 4(4): 3529–3536.
- [18] SJTU-MIRUS. This video is supporting material for my recently submitted paper[EB/OL]. (2025-06-15)[2025-06-16]. <https://sjtu-mirus.github.io/MIRUS.github.io/research/kai>.

A path planning method for adaptive formation reshaping of multi-UUVs based on affine transformation

GUO Kai, YAO Mingqi, QIAO Lei*

School of Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

Abstract: [**Objective**] To address the challenge of simultaneously maintaining formation integrity and enabling flexible obstacle avoidance for multi-unmanned underwater vehicle (multi-UUV) formations in complex underwater environments, this paper proposes a global path planning method that supports adaptive formation reshaping. [**Method**] The proposed method is built upon an affine transformation framework that maps the cooperative path planning problem of the multi-UUV system into a two-dimensional affine parameter space. First, a front-end path search is conducted using an improved rapidly-exploring random tree* (RRT*) algorithm. By integrating fast exploration and iterative optimization phases, a weighted k-dimensional (KD) tree, a hybrid sampling mechanism, and adaptive tuning of sampling parameters, this algorithm efficiently generates an initial sequence of affine states. Subsequently, a B-spline-based back-end optimizer employs a gradient descent method to minimize a comprehensive objective function that accounts for trajectory smoothness, UUV kinematic feasibility, environmental collision safety, and the cost associated with adaptive formation scaling. The optimization process yields a continuous and smooth trajectory of affine parameters that satisfies multiple constraints. [**Results**] Lake experiments demonstrate that the proposed planning method can generate safe and feasible formation paths. It successfully guided the multi-UUV formation through a simulated narrow obstacle region, while the actual velocities and accelerations of the UUVs remained within the predefined feasibility constraints. [**Conclusion**] The proposed global planning method, based on affine transformation, effectively generates safe and feasible paths for multi-UUV formations navigating complex obstacle environments by enabling adaptive formation reshaping. This method significantly enhances the autonomy and environmental adaptability of marine unmanned vehicles, and holds great value for advancing the development and practical application of marine unmanned systems technology.

Key words: unmanned underwater vehicle; formation path planning; affine transformation; rapidly-exploring random tree*; trajectory optimization