文章编号:1000 - 4653(2026)04 - 0001 - 11

基于 UUV 模型的三维混合 A* 定高路径规划

严浙平*, 董 岳
(哈尔滨工程大学 青岛创新发展基地, 山东 青岛 266000)

摘 要:针对无人水下航行器(UUV)在复杂地形环境下的定高路径规划问题,提出了一种基于 UUV 运动特性改进的混合 A* 路径规划算法(UUVA*)。首先,结合 UUV 的非完整运动约束,对传统混合 A* 算法进行状态空间扩展。其次,设计了适应 UUV 航行特性的运动基元,同时对代价函数进行优化,引导生成平滑且适航的路径。最后,为提高搜索效率,本文引入了多分辨率分层搜索与分支修剪机制,显著减少了节点扩展数量。通过在海底山丘、海底峡谷两种典型复杂地形下的仿真测试,验证了算法的定高规划能力,并与经典 A*、混合 A* 算法和 RRT* 算法进行对比仿真试验。结果表明,所提算法生成的路径更加平滑,UUV 跟踪误差更小,更适合实际应用,验证了本文方法在轨迹跟踪误差和路径可航行性方面的优越性。

关键词:智能航运;路径规划;改进混合 A* 算法;无人水下航行器;地形跟踪;动力学约束
中图分类号:U674.941;TP242 **文献标志码**:A **DOI**:10.3969/j.issn.1000 - 4653.2026.04.001

Fixed-altitude 3D path planning for unmanned underwater vehicle using improved hybrid A*

*Yan Zheping**, *Dong Yue*
(Qingdao Innovation Development Base, Harbin Engineering University, Qingdao 266000, China)

Abstract: To address the altitude-constrained path planning problem for Unmanned Underwater Vehicles (UUVs) in complex terrain environments, this paper proposes an improved hybrid A* path planning algorithm based on UUV motion characteristics, termed UUVA*. Firstly, the traditional hybrid A* algorithm is extended by expanding the state space and incorporating the nonholonomic motion constraints of UUVs. Secondly, motion primitives tailored to UUV navigation characteristics are designed, and the cost function is optimized to generate smoother and more navigable paths. Finally, to enhance search efficiency, a multi-resolution hierarchical search strategy and branch pruning mechanism are introduced, significantly reducing the number of node expansions. Simulation tests conducted in two representative complex terrains—submarine hills, and submarine canyons—verify the altitude-constrained planning capability of the proposed algorithm. Furthermore, comparative experiments against classical A* algorithm, hybrid A*, and RRT* algorithms are conducted. The result shows that the generated paths are smoother and yield smaller tracking errors, making the proposed algorithm more suitable for practical UUV applications, which verifies the superior performance of the proposed method in terms of trajectory tracking error and path navigability.

Key words: intelligent shipping; path planning; the improved hybrid A* algorithm; unmanned underwater vehicle; terrain following; dynamic constraints

无人水下航行器(Unmanned Underwater Vehicle, UUV)因其高效、安全与经济等优势,广泛应用于海洋勘探、军事侦察及环境监测等任务中[1-3]。在海底测绘、管道巡检和科学考察等作业场景中,UUV

收稿日期:2025-11-28
基金项目:国家自然科学基金项目(52571334)资助
第一作者(* 通信作者):严浙平(1972—),男,教授,博士,研究方向为水下无人系统及潜航器自主与智能控制技术,(E-mail)yanzheping@hrbeu.edu.cn
引用格式:严浙平,董岳. 基于 UUV 模型的三维混合 A* 定高路径规划[J]. 中国航海,2026,49(4):1 - 11.
Yan Z P, Dong Y. Fixed-altitude 3D path planning for unmanned underwater vehicle using hybrid A* [J]. Navigation of China, 2026, 49(4):1 - 11. (in Chinese)

需沿等高面稳定航行,以确保任务执行的连续性与安全性。目前,多数研究将定高航行视为控制问题,即先规划路径再执行轨迹跟踪。其通常忽视 UUV 的动力学特性及非完整性运动约束,导致规划路径与实际可行轨迹之间存在偏差。此外,传统规划算法多将 UUV 简化为质点模型,未充分考虑水动力阻力等动态因素,致使所生成路径需依赖大量后处理与轨迹拟合。因此,亟须在路径规划阶段直接融合 UUV 的动力学与运动学特性,以提升航行性能与轨迹可行性。

近年来,UUV 路径规划技术取得了显著进展,涵盖全局路径规划、局部路径规划及运动约束建模等多个研究方向。

在全局路径规划方面,Zeng 等[4]引入帝国竞争算法(Imperialist Competitive Algorithm, ICA),优化 UUV 在复杂海洋环境中的航迹选择,显著提升了轨迹的优越性;张岳星等[5]基于电子海图与改进粒子群优化算法(Particle Swarm Optimization, PSO),提出了一种适用于三维静态环境的高效路径规划方法;Pan 等[6]结合改进蚁群算法与遗传算法,构建了双层路径规划架构,增强了三维复杂环境下的搜索效率。这类方法的优点是全局性强,能有效跳出局部最优,但也存在计算复杂度高、实时性不足的问题,因此更适用于离线预规划场景。

在局部路径规划方面,Petres 等[7]提出了一种基于快速行进算法(Fast Marching, FM)的方法,能够在考虑水流影响及车辆转弯半径约束的前提下,为 UUV 规划连续的二维可行路径;Zhang 等[8]引入深度强化学习框架,使 UUV 在未知且动态变化的环境中具备自主路径决策能力。这类方法的优势在于实时性和环境适应性,但通常缺乏全局最优保证,并且强化学习类方法存在训练开销大、泛化能力有限的问题。

针对动力学与运动学约束的建模问题,Hou 等[9]借助强化学习引入运动学限制,从而提升了路径规划的可行性与执行效果;Scharff Willners 等[10]扩展混合 A*算法,将 UUV 的非完整运动特性纳入建模框架中,有效实现了避障与轨迹可行性之间的平衡;Liu 等[11]通过引入任务特定的动力学模型,改进了干扰流体动力系统(Improved Interfered Fluid Dynamic System, IIFDS),显著增强了路径的可执行性;Zhang 等[12]提出结合改进 A*人工势场法的混合路径规划方法,在搜索过程中直接嵌入动力学约束,提升了算法的实用性与鲁棒性。这类方法优点在于生成的路径更贴合实际运动能力,但存在搜索空间增大、效率下降的不足。

在定高航行相关研究方面,Yan 等[13]提出基于积分终端滑模控制(Integral Terminal Sliding Mode Control, ITSMC)的地形跟随策略,在干扰环境下提升了 UUV 航行的稳定性;Yan 等[14]针对配置单波束传感器的 UUV,设计了基于反步法的深度控制,实现了稳定可靠的地形跟随控制;Ru 等[15]将 A*算法与近底巡航策略相结合,在大规模海底复杂环境中实现了高效且安全的路径规划。这些方法主要着重于控制与导航层面的地形适应性,对路径规划的全局最优与非完整约束考虑不足。

针对以上不足,本文提出了 UUVA*算法,其创新点在于:1)结合非完整约束与定高需求,设计符合实际航行能力的运动基元,使路径更符合 UUV 动态特性;2)代价函数优化,提升路径可行性与平滑性;3)引入多分辨率搜索与分支修剪机制,在保证路径质量的同时满足实时性需求。因此,本文方法兼顾了全局可行性、路径平滑性和搜索效率,在静态复杂地形环境下为 UUV 提供了更具适航性和可执行性的路径规划方案。

本文部分研究成果基于本团队前期在 UUV 导航与路径规划领域的研究基础。例如,Yan 等[16]提出一种改进的 A*法,有效优化了复杂环境中 UUV 的路径规划问题,显著改善了障碍物规避与路径平滑性;Zhang 等[17]进一步引入动力学约束,扩展该算法为混合 A*式,提升了算法在不规则水下环境中的精度与效率。基于上述成果,本文针对 UUV 的非完整性运动特性与定高航行任务需求,提出了一种改进的混合 A*路径规划算法,从而进一步提升路径的适用性、可行性与任务稳定性。

1 UUV 的模型

本文将坐标系分为固定坐标系 $E\text{-}\xi\eta\zeta$ 和运动坐标系 $G\text{-}xyz$ 进行建模。固定坐标系原点 E 设置为海中一点,$E\xi$ 轴保持水平,$E\eta$ 置于水平面内,与 $E\xi$ 轴垂直,$E\zeta$ 垂直于 $E\xi\eta$ 坐标面,正方向指向地心。运动坐标系坐标原点 G 设置在 UUV 重心,Gx 从纵中剖面选取,指向艏向,Gy 垂直于纵中剖面,指向 UUV 右舷,Gz 处于纵中剖面,指向航行器底部。如图 1 所示。

UUV 的位置与姿态向量表示为

$$\boldsymbol{\eta} = [x, y, z, \varphi, \theta, \psi] \tag{1}$$

式(1)中:x、y、z 为 UUV 在固定坐标系下的位置坐标;φ、θ、ψ 分别为横倾角、纵倾角与艏向角。

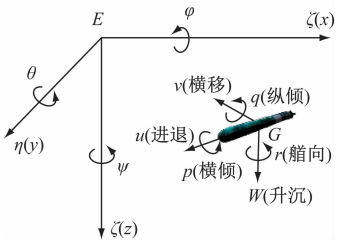


图 1 UUV 坐标系

Fig. 1 UUV coordinate system

UUV 的速度和角速度表示为

$$\boldsymbol{v} = [u, v, w, p, q, r] \tag{2}$$

式(2)中:u、v、w 为沿运动坐标系 x、y、z 轴方向的线速度;p、q、r 为绕相应轴的角速度。

UUV 的姿态变化和位移关系式为

$$\begin{cases} \dot{x} = u\cos\varphi\cos\theta + v(\cos\psi\sin\theta\sin\varphi - \sin\psi\cos\varphi) + \\ \quad w(\cos\psi\sin\theta\cos\varphi + \sin\psi\sin\varphi) \\ \dot{y} = u\sin\varphi\cos\theta + v(\sin\psi\sin\theta\sin\varphi + \cos\psi\cos\varphi) + \\ \quad w(\sin\psi\sin\theta\cos\varphi - \cos\psi\sin\varphi) \\ \dot{z} = -u\sin\theta + v\cos\theta\sin\varphi + w\cos\theta\cos\varphi \end{cases} \tag{3}$$

UUV 的姿态角变化满足:

$$\begin{cases} \dot{\varphi} = p + q\sin\varphi\tan\theta + r\cos\varphi\tan\theta \\ \dot{\theta} = q\cos\varphi - r\sin\varphi \\ \dot{\psi} = q\sin\varphi/\cos\theta + r\cos\varphi/\cos\theta \end{cases} \tag{4}$$

结合式(3)和式(4)可得

$$\dot{\boldsymbol{\eta}} = J(\boldsymbol{\eta}) \cdot v \tag{5}$$

由严浙平等[18]的工作可得,UUV 的平移运动方程为:

$$m\left[\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{\Omega} \times \boldsymbol{U} + \frac{\partial \boldsymbol{\Omega}}{\partial t} \times \boldsymbol{R}_G + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{R}_G)\right] = \boldsymbol{F}_\Sigma \tag{6}$$

其中:

$$\begin{cases} \dfrac{\partial \boldsymbol{U}}{\partial t} = [\dot{u} \quad \dot{v} \quad \dot{w}]^{\mathrm{T}} \\ \boldsymbol{\Omega} \times \boldsymbol{U} = [wq - vr \quad ur - wp \quad vp - uq]^{\mathrm{T}} \\ \dfrac{\partial \boldsymbol{\Omega}}{\partial t} \times \boldsymbol{R}_G = [z_G\dot{q} - y_G\dot{r} \quad x_G\dot{r} - z_G\dot{p} \quad y_G\dot{p} - x_G\dot{q}]^{\mathrm{T}} \\ \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{R}_G) = \begin{bmatrix} (y_Gp - x_Gq)q + (z_Gp - x_Gr)r \\ (z_Gq - y_Gr)r + (x_Gq - y_Gp)p \\ (x_Gr - z_Gp)p + (y_Gr - z_Gq)q \end{bmatrix} \end{cases} \tag{7}$$

式(7)中:m 为航行器运动时的质量(kg);$\boldsymbol{U}$ 为航行器的运动坐标系原点的移动速度(m/s);$\boldsymbol{\Omega}$ 为航行器的转动角速度(rad/s);$\boldsymbol{R}_G$ 为航行器的重心 $\boldsymbol{G}$ 在运动坐标系下的位置向量(m);$\boldsymbol{F}_\Sigma$ 为航行器所受合外力(N)。

UUV 的旋转运动方程为:

$$\begin{aligned} &\boldsymbol{J}_G\frac{\partial \boldsymbol{\Omega}}{\partial t} + \boldsymbol{R}_G \times m\left(\frac{\partial \boldsymbol{\Omega}}{\partial t} \times \boldsymbol{R}_G\right) + \boldsymbol{\Omega} \times \boldsymbol{J}_G\boldsymbol{\Omega} + \\ &\boldsymbol{\Omega} \times [\boldsymbol{R}_G \times m(\boldsymbol{\Omega} \times \boldsymbol{R}_G)] + \\ &\boldsymbol{R}_G \times m\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{R}_G \times (\boldsymbol{\Omega} \times m\boldsymbol{U}) = \boldsymbol{T}_\Sigma \end{aligned} \tag{8}$$

其中:

$$\begin{cases} \boldsymbol{J}\dfrac{\partial \boldsymbol{\Omega}}{\partial t} = \begin{bmatrix} J_x & J_{xy} & J_{xz} \\ J_{yx} & J_y & J_{yz} \\ J_{zx} & J_{zy} & J_z \end{bmatrix}\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} \\ \boldsymbol{\Omega} \times (\boldsymbol{J}\boldsymbol{\Omega}) = \\ \quad \begin{bmatrix} (J_{zx}p + J_{zy}q + J_zr)q - (J_{yx}p + J_yq + J_{yz}r)r \\ (J_xp + J_{xy}q + J_{xz}r)r - (J_{zx}p + J_{zy}q + J_zr)p \\ (J_{yx}p + J_yq + J_{yz}r)p - (J_xp + J_{xy}q + J_{xz}r)q \end{bmatrix} \\ \boldsymbol{R}_G \times m\dfrac{\partial \boldsymbol{U}}{\partial t} = m\begin{bmatrix} y_G\dot{w} - z_G\dot{v} \\ z_G\dot{u} - x_G\dot{w} \\ x_G\dot{v} - y_G\dot{u} \end{bmatrix} \\ \boldsymbol{R}_G \times (\boldsymbol{\Omega} \times m\boldsymbol{U}) = \\ \quad m\begin{bmatrix} y_G(vp - uq) + z_G(wp - ur) \\ z_G(wq - vr) + x_G(uq - vp) \\ x_G(ur - wp) + y_G(vr - wq) \end{bmatrix} \end{cases} \tag{9}$$

式(8)中:$\boldsymbol{J}_G$ 为航行器对原点在重心的坐标系的惯性矩阵;$\boldsymbol{T}_\Sigma$ 为航行器所受合外力对运动坐标系原点的力矩(N·m)。

2 UUV 的混合 A* 算法

2.1 混合 A* 算法

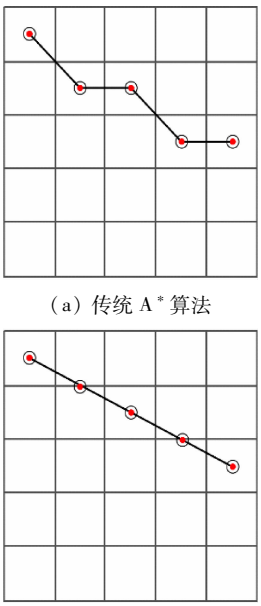
混合 A*(Hybrid A*)算法由斯坦福大学的 Dolgov 等人提出,并在 2007 年 DARPA 城市挑战赛中成功应用。该方法结合了传统 A* 算法的启发式搜索机制与移动体动力学约束,能够生成连续、曲率可行的轨迹,特别适用于存在非完整约束的自主系统。

如图2 所示,相较传统 A* 仅在离散栅格中心扩展节点,混合 A* 可在栅格内部的连续空间内扩展状态,这种方法可以抹平不必要的转折,显著提升路径的平滑性与可执行性。

混合 A* 算法使用开放列表(Open List)维护待扩展状态,使用关闭列表(Closed List)避免重复扩展。在每次迭代中,算法都会选取开放列表中总代价最小的状态点进行扩展,总代价函数为:

$$F(x) = g(x) + h(x) \tag{10}$$

式(10)中:$g(x)$ 和 $h(x)$ 分别为状态点的路径成本和启发式成本。



(a) 传统 A* 算法

(b) 混合 A* 算法

图 2 算法扩展方式

Fig. 2 Algorithm extension method

2.2 基于 UUV 的改进混合 A* 算法

2.2.1 状态空间扩展

由于 UUV 主要在三维水下空间工作,传统二维平面的混合 A* 算法不足以处理其自由度。虽然 Scharff Willners 提出了适用于 3D 空间的混合 A*,但其未考虑 UUV 的非完整约束。

为了更完备地刻画系统动态特性并提升状态估计性能,本文在原本如式(1)所示的六维状态空间的基础上,引入式(2)所示的速度和角速度,构建了十二维状态向量:

$$\boldsymbol{S} = [x, y, z, \varphi, \theta, \phi, u, v, w, p, q, r] \tag{11}$$

式(11)中:横倾角 φ 和角速度 p 自由变化,仅参与动力学建模,不施加运动约束。

参考式(6)和式(8)描述的 UUV 动力学模型,并结合相关 UUV 的水动力特性报告[19-20],可以建立 UUV 的牛顿-欧拉方程:

$$\boldsymbol{F} = \boldsymbol{M} \cdot \dot{\boldsymbol{v}} \tag{12}$$

式(12)中:$\boldsymbol{F}$ 为 UUV 所受合外力和合力矩;$\boldsymbol{M}$ 为惯性矩阵;$\dot{\boldsymbol{v}}$ 为 UUV 的线加速度和角加速度。

其中,

$$\boldsymbol{M} = \begin{bmatrix} m - \frac{1}{2}\rho L^3 X_u & 0 & 0 & 0 & mz_G & -mz_G \\ 0 & m - \frac{1}{2}\rho L^3 Y_v & 0 & 0 & 0 & mx_G - \frac{1}{2}\rho L^4 Y_r \\ 0 & 0 & m - \frac{1}{2}\rho L^3 Z_w & my_G & -mx_G - \frac{1}{2}\rho L^4 Z_q & 0 \\ 0 & -mz_G - \frac{1}{2}\rho L^4 K_v & my_G & I_x - \frac{1}{2}\rho L^5 K_p & 0 & 0 \\ mz_G & 0 & -mx_G - \frac{1}{2}\rho L^4 M_w & 0 & I_y - \frac{1}{2}\rho L^5 M_q & 0 \\ -my_G & mx_G - \frac{1}{2}\rho L^4 N_v & 0 & 0 & 0 & I_z - \frac{1}{2}\rho L^5 N_r \end{bmatrix} \tag{13}$$

式(13)中:ρ 为 UUV 所处的流体密度(kg/m^3);L 为 UUV 的总长(m);X_u、Y_v、Y_r、Z_w、Z_q、K_v、K_p、M_w、M_q、N_v、N_r 分别表示 UUV 在不同方向上的阻尼(kg/s);x_G、y_G、z_G 分别表示 UUV 质心相对于状态点的位置(m);I_x、I_y、I_z 分别表示 UUV 绕 x、y、z 轴的转动惯量(kg·m^2)。

将式(12)的惯性矩阵求逆,即可得到

$$[\dot{u} \quad \dot{v} \quad \dot{w} \quad \dot{p} \quad \dot{q} \quad \dot{r}]^{\mathrm{T}} = \boldsymbol{M}^{-1} \cdot \boldsymbol{F} \tag{14}$$

设 t 时刻 UUV 的状态变量为 S_t,结合式(5)和式(12),可得到完整的状态向量导数 $\dot{S}$。

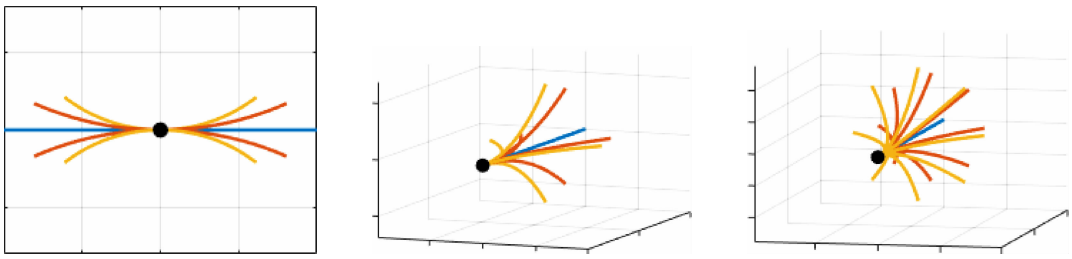
$$\dot{S} = [J(\eta) \quad \boldsymbol{M}^{-1}] \begin{bmatrix} S_{7:12} \\ \boldsymbol{F} \end{bmatrix} \tag{15}$$

因此,UUV 的状态转移方程可表示为:

$$S_{t+T} = S_t + \int_t^{t+T} \dot{S} \mathrm{d}t \tag{16}$$

2.2.2 使用运动基元扩展状态点

混合 A* 算法通常通过设定速度与转向角进行状态扩展,适用于允许后退的地面车辆系统,其运动基元如图 3a 所示。Scharff Willners 将运动基元扩展至三维空间,引入 z 轴并取消后退,形成适用于部分飞行与水下系统的三维混合 A* 算法。该方法采用固定转向角进行状态扩展,且每次仅沿 y 轴与 z 轴分别扩展,形成如图 3b 所示的十字型扩展结构,这种结构存在搜索效率低与路径质量低的问题,且未引入完整的动力学约束。



(a) 传统 A* 算法 (b) 三维混合 A* 算法 (c) UUV 的混合 A* 算法

图 3 算法运动基元

Fig. 3 Algorithmic motion primitives

针对以上不足,本文基于 UUV 动力学特性重新设计运动基元。

通过对 UUV 建模,可以得知控制力的矩阵:

$$\begin{bmatrix} F_0 \\ F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{\rho}{2}L^2 Y_{dr}|u|u \\ 0 & 0 & \frac{\rho}{2}L^2 Z_{ds}|u|u + \frac{\rho}{2}L^3 Z_{1q1ds}u|q| & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\rho}{2}L^3 M_{ds}|u|u + \frac{\rho}{2}L^4 Z_{1q1ds}u|q| & 0 \\ 0.25 & -0.25 & 0 & \frac{\rho}{2}L^3 N_{dt}|u|u \end{bmatrix} \cdot \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \end{bmatrix} \tag{18}$$

式(18)中:$F_0 \sim F_5$ 为 UUV 在 x、y、z 三个方向上受到的控制力(N)和控制力矩(kg · m^2/s^2);u_0、u_1、u_2、u_3 为 UUV 的两个主推进器的推力(N)及升降舵和水平舵的舵角(rad);u、q 分别为 UUV 运动坐标系中沿 x 轴的速度(m/s)和绕 y 轴的角速度(rad/s);Y_{dr}、Z_{ds}、Z_{1q1ds}、M_{ds}、M_{1q1ds}、N_{dr}为 UUV 水动力系数。

通过设定推进器推力与舵角,可计算 UUV 在特定时间段内的受力情况,进而推导状态变化率。该方式取代传统速度-转向角运动基元,可以使路径扩展过程显式考虑动力学约束,提升路径的物理可行性。同时,过程中的推力与舵角亦可直接作为控制器参考输入,减轻轨迹跟踪负担。

此外,本文的算法允许升降舵与水平舵独立控制舵角,使 UUV 可在 y、z 上同步运动,形成如图 3c 所示的多向扩展结构,显著提升扩展轨迹的多样性与搜索效率。

2.2.3 代价函数优化

由式(10)可知,在路径规划算法中,启发式成本常简化为当前状态点到目标点的欧几里得距离:

$$h(s) = W_r \cdot C_{hr}(s) \tag{19}$$

式(19)中:W_r 为位置权重;C_{hr}为位置估计成本。

当前节点路径成本则为上一状态的路径成本加上当前位置变化成本与角度变化成本:

$$g(s) = g_{last} + W_r \cdot C_{gr}(s) + W_{so} \cdot C_{gso}(s) \tag{20}$$

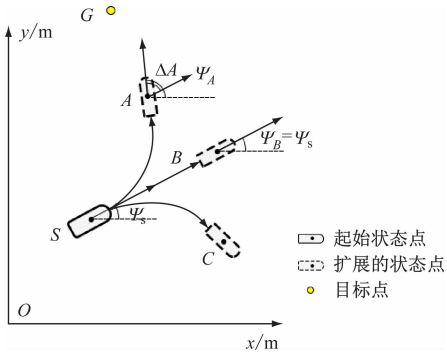
式(20)中:g_{last}为上一节点路径成本;W_r 和 W_{so}为位置权重和角度权重;C_{gr} 和 C_{gso} 为位置变化和角度变化。

尽管角度成本的引入可抑制不必要的频繁转向,但在某些场景中会导致搜索路径偏离最优。图 4 为一典型案例,如图 4 所示,当从起点 S 向目标点 G 搜索路径时,状态 A 虽更接近目标,但由于其转向成本 ΔA 较高,可能被具有更小角度变化成本但路径劣于 A 的状态 B 优先扩展。这种扩展方式降低了搜索效率,且会在无法避免角度成本的复杂环境中使算法趋于保守。

为解决此问题,本文在启发式成本中引入角度估计成本项,以平衡路径成本中的转向惩罚,更新后的启发式函数为:

$$h(s) = W_r \cdot C_{hr}(s) + W_{so} \cdot C_{hso}(s) \tag{21}$$

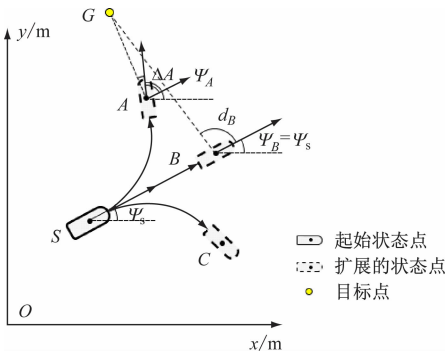
式(21)中:C_{hso}为 UUV 当前艏向与当前位置至目标点方向之间的夹角,衡量 UUV 未来必要的转向角大小。图 5 展示了新算法中状态 A,B,C 的扩展情况,虽然 B 的角度变化成本 ΔB 仍小于 A 的,但其角度估计成本 dB 更高,因此总代价函数中 A 更优,从而被优先扩展。



Ψ_A – 状态 A 艏向角；Ψ_B – 状态 B 艏向角；
Ψ_S – 状态 S 艏向角；ΔA – 状态 A 角度变化成本。

图 4 扩展的角度成本

Fig. 4 Expansion angle cost



Ψ_A – 状态 A 艏向角；Ψ_B – 状态 B 艏向角；
Ψ_S – 状态 S 艏向角；ΔA – 状态 A 角度变化成本；
$\mathrm{d}A$ – 状态 A 角度估计成本；$\mathrm{d}B$ – 状态 B 角度估计成本。

图 5 扩展的角度估计成本

Fig. 5 Expansion angle cost estimate

此外，为适应 UUV 的定高航行需求，本文在启发式成本中进一步引入高度代价项，使路径趋于参考高度航行。最终启发式成本表达如下：

$$h(s) = W_{\mathrm{r}} \cdot C_{hr}(s) + W_{\mathrm{so}} \cdot C_{hso}(s) + W_{hei}(s) \cdot C_{hei}(s) \tag{22}$$

式(22)中：W_{r}、W_{so}、W_{hei} 分别为位置、角度、高度权重；C_{hr}、C_{hso}、C_{hei} 分别为位置、角度、高度的估计成本。

2.2.4 多分辨率搜索和分支修剪

在混合 A* 算法中，角度分辨率决定了每次状态扩展时可采样的转向角间隔。分辨率越高，状态扩展的方向越精细，路径平滑性和精度也随之提升；但与此同时，状态空间规模将呈指数增长，尤其在 UUV 需同时在水平与垂直两个自由度上进行扩展的情况下，计算开销显著增加，严重影响搜索效率。

为解决这一问题，本文设计了一种多分辨率搜索的方法，即首先以较低分辨率快速扩展，筛选潜力状态扇区，再在局部以高分辨率细化搜索。

如图 6 所示，在从当前点 S 到目标点 G 的规划过程中，算法会初步扩展出 1 ~ 5 个状态，并根据式(10)的总代价函数在内部优选出代价更低的状态 2、3，将它们之间的区域设定为潜力扇区，在潜力扇区内部以更高分辨率扩展出 6 ~ 8 个状态，在保证搜索质量的前提下减少计算开销。

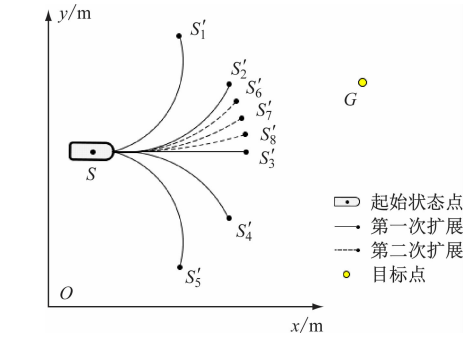


图 6 多分辨率搜索

Fig. 6 Multi-resolution search

为防止高分辨率扩展可能导致状态空间密集冗余的问题，本文使用一种分支修剪策略，新扩展的状态会被映射至其对应的栅格中心，若已有状态占据该栅格，系统将根据总代价进行竞争，仅保留代价更优者。图 7 展示了该策略的效果，状态 3 和状态 4 映射至同一栅格中心，因状态 3 总代价更低，状态 4 被舍弃。

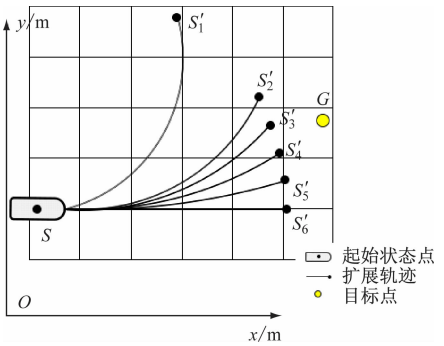


图 7 分支修剪

Fig. 7 Pruning branches

以方向舵与升降舵角度范围设为[– 1，1]、分辨率为 0.1 为例，单次扩展将产生超过 400 个状态点；而结合多分辨率扩展与分支修剪后，扩展状态数可降低至少 50 个。如图 8 所示，经实测，状态空间冗余减少近 80%，搜索时间缩短约 50%，在大幅提升效率的同时，仅对路径质量产生极小影响。

2.3 算法流程

基于以上改进，本文将基于 UUV 模型的混合 A* 算法称为 UUVA* 算法，其根据 UUV 的运动学和动力学方程扩展状态树，在规划路径的同时完成定高任务，并在规划完成后向控制器输出路径导航点

及预输入控制量队列,算法流程图如图 9 所示。

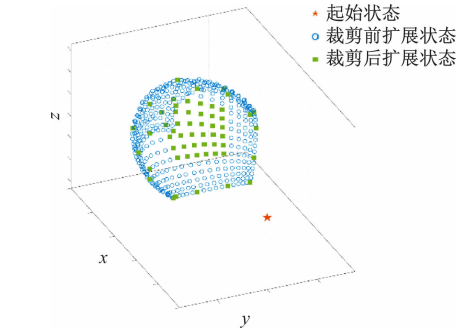


图 8 裁剪前后数量对比

Fig. 8 Comparison before and after pruning

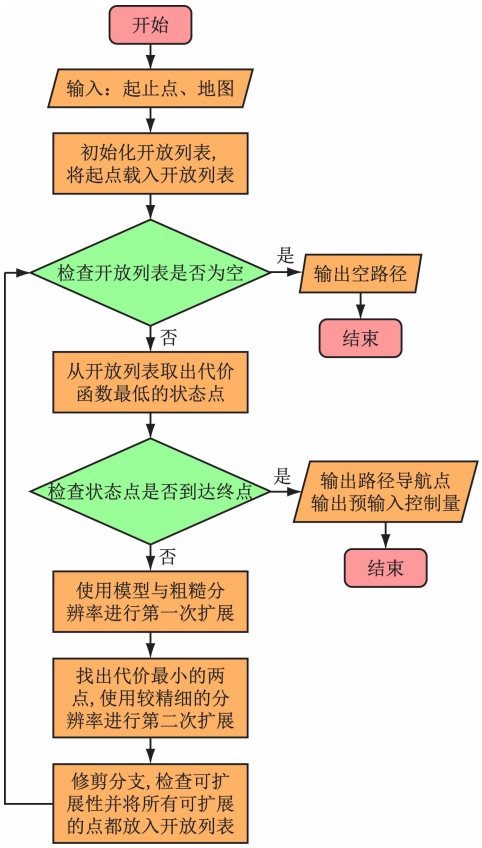


图 9 UUVA* 算法流程图

Fig. 9 UUVA* algorithm flowchart

3 仿真试验验证

为验证所提出的 UUVA* 算法的有效性,本文在海底山丘与海底峡谷两种典型地形环境下,分别与三种常用路径规划算法(A*、混合 A*、RRT*)进行了对比仿真试验。两种环境的俯视图如图 10 所示,其中绿色三角形表示起点,红色五角星表示终点。试验区域均为 100 m×100 m 的正方形区域,地形高差约为 15 m。参考航行高度设定为距地形表面 5 m。

试验采用四种路径规划算法分别生成从起点到终点的路径,并在相同的 UUV 动力学模型下进行轨迹跟踪仿真。部分 UUV 模型参数如表 1 所示。

表 1 UUV 模型参数

Tab. 1 UUV model parameters

参数	含义	数值
L/m	船体总长	4
m/kg	总质量	1 000
I_x/kg·m^2	绕 x 轴转动惯量	298
I_y/kg·m^2	绕 y 轴转动惯量	3 298
I_z/kg·m^2	绕 z 轴转动惯量	3 400

轨迹跟踪控制采用 PID 控制器,其中推进器控制器比例增益 K_p、积分增益 K_i、微分增益 K_d 分别设为 16、1.5 和 3.0;方向舵与水平舵控制器比例增益、积分增益和微分增益分别设为 3.0、0.02 和 0.2。首先通过齐格勒-尼科尔斯整定方法获得上述参数的初始值,再根据实际轨迹跟踪效果进行手动微调。所有试验均在 Intel(R) Core(TM) i7-10750H CPU @ 2.60GHz 平台上,基于 C++ 与 ROS(Robot Operating System)实现[21]。

3.1 海底山丘仿真试验

该区域由位于地图中心的两座山丘及其周围缓坡构成,如图 10a 所示。UUV 需从山丘侧面绕行以抵达终点。起点坐标设置为(25.0,25.0),终点坐标设置为(95.0,45.0)。四种算法的规划结果如图 11 所示。

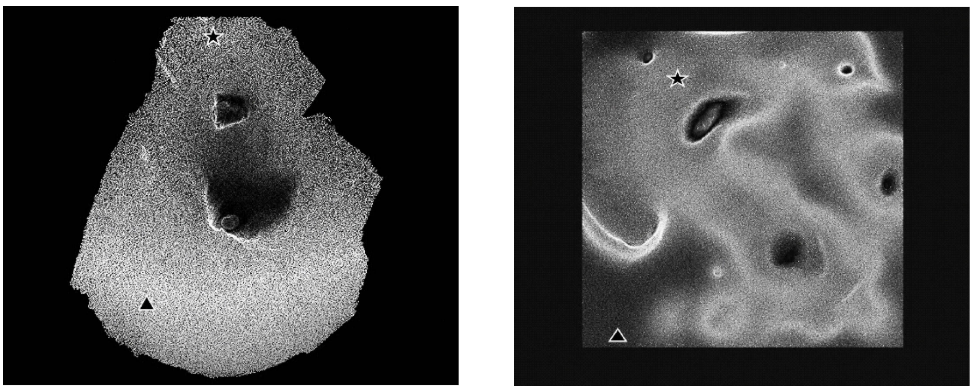
由图 11 可见,A* 算法和 RRT* 算法生成的路径转折较多,导致轨迹跟踪时的负担较大;而混合 A* 与 UUVA* 算法生成的路径更为平滑,符合 UUV 的运动特性。

基于 UUV 模型的轨迹跟踪试验进一步验证了这一差异。图 12 为 UUV 的定高跟踪误差对比结果,如图 12 所示,UUVA* 生成的轨迹误差较其他算法更小,这表明 UUVA* 算法在地形跟随能力方面具有优势。

海底山丘的轨迹跟踪数据对比如表 2 所示。可以观察到,UUVA* 的误差峰值远低于其他算法,且误差曲线收敛速度更快,振荡幅度更小。例如,在部分山丘地形边缘,A* 与 RRT* 的误差峰值接近 1.0 m,而 UUVA* 的误差峰值为约 0.3 m。这说明 UUVA* 生成的路径更易于被 UUV 实际跟踪。

3.2 海底峡谷仿真试验

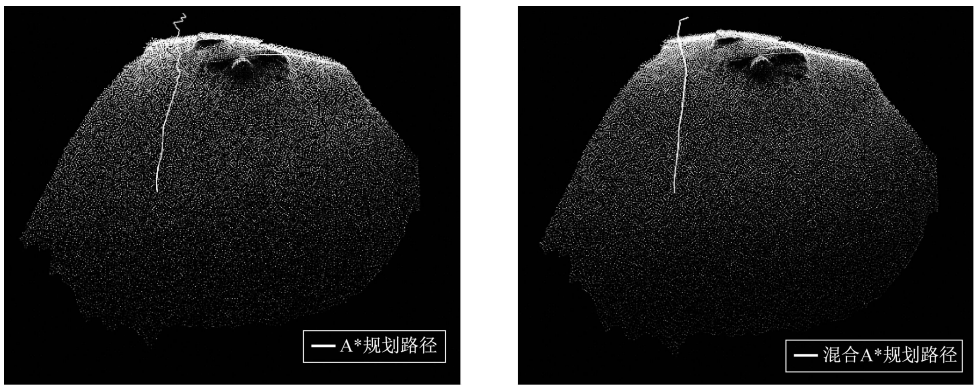
与海底山丘区域相比,海底峡谷地形更加复杂,



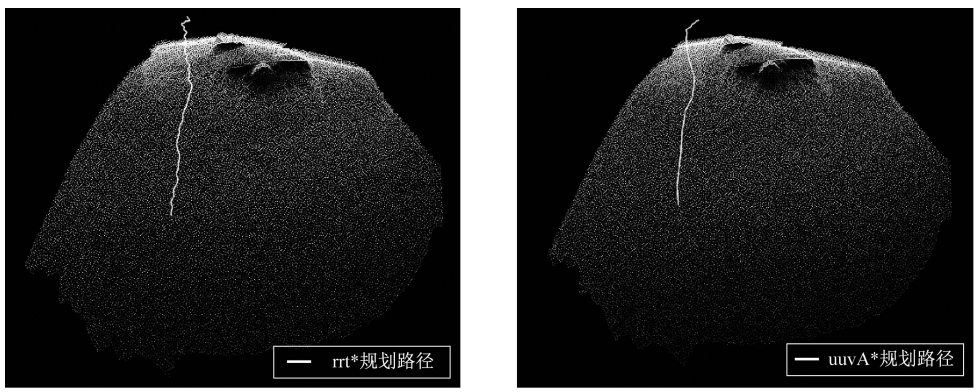
(a) 海底山丘 (b) 海底峡谷

图 10 地形俯视图

Fig. 10 Topographic top view



(a) 经典 A* 算法 (b) 混合 A* 算法



(c) RRT* 算法 (d) UUVA* 算法

图 11 海底山丘规划路径

Fig. 11 Planning path of submarine hills

其包含多个较高的突起物及起伏不定的缓坡,如图 10b 所示。UUV 需绕过不可逾越的障碍物,并经相对平缓的峡谷通道到达终点。起点坐标设置为(1.0,10.0),终点坐标设置为(85.0,30.0)。四种算法的规划结果如图 13 所示。

由图 13 可见,A* 和 RRT* 算法的路径同样存在大量急转折,而混合 A* 与 UUVA* 生成的路径则更加平滑和连续,显著降低了跟踪控制的难度。

UUV 的定高跟踪误差对比结果如图 14 所示。海底峡谷的轨迹跟踪数据如表 3 所示。结果表明 UUVA* 的误差始终维持在较低水平,最大误差通常不超过 0.2 m,而其他算法的误差峰值在 0.7 ~ 1.0 m 之间。此外,UUVA* 的误差曲线波动幅度最小,体现出更好的鲁棒性与可跟随性。

3.3 结果分析

综合两组仿真试验结果可知,UUVA* 算法在路径规划与轨迹跟踪方面均表现出明显优势。首先,在路径平滑性方面,UUVA* 显著减少路径转折,使

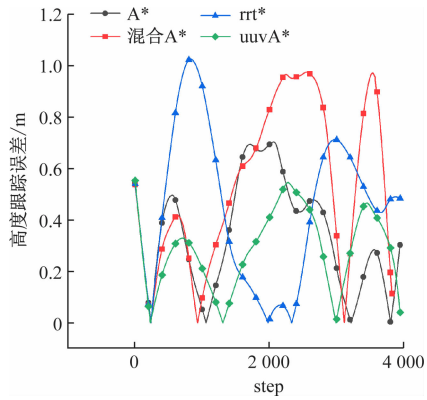


图 12 海底山丘定高跟踪误差

Fig. 12 Altitude tracking error of submarine hills

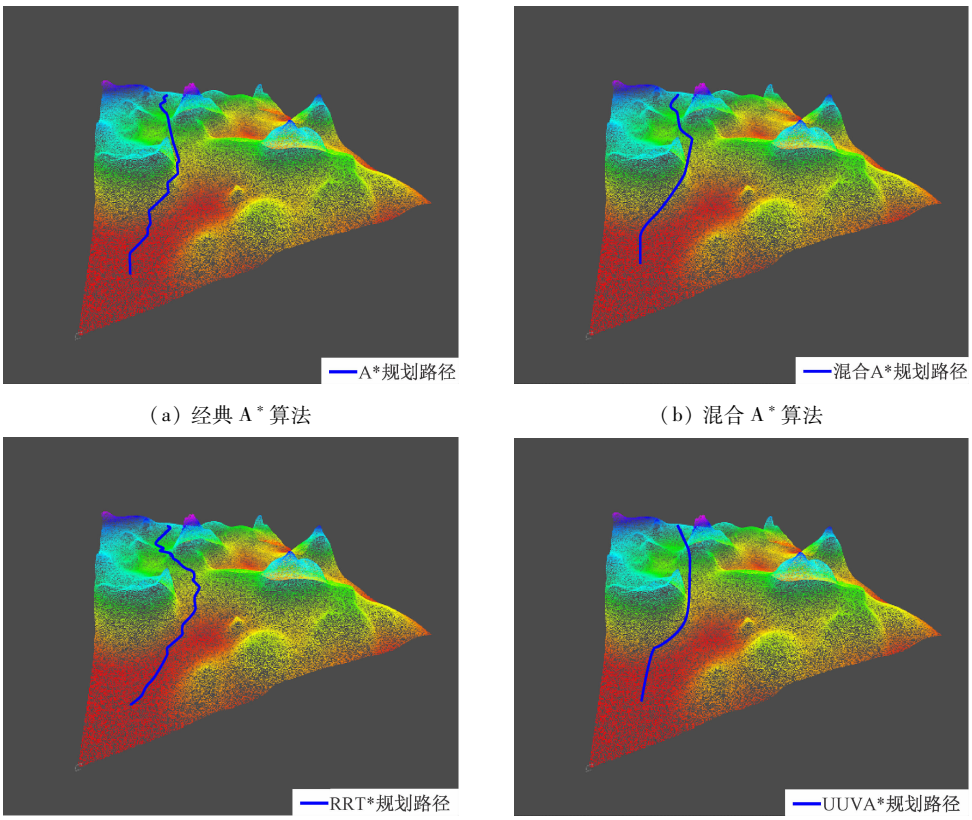
规划结果更加连续,避免急转弯带来的控制负担。其次,从高度跟随性来看,UUVA* 在地形起伏区域能够保持与参考高度曲线高度一致,变化连续且突变幅度小,体现出优异的地形适应性。最后,在轨迹跟踪性能上,UUVA* 的跟踪误差低于 A*、混合 A* 和 RRT* 算法,其峰值误差仅为其他算法的 10% ~ 30%,最高不超过 50%。整体曲线收敛迅速、振荡幅度小,显示出优良的可跟随性。

根据两组试验的数据对比可以看出,误差的产生不仅与算法本身相关,还受到地形复杂度与参数设置的影响。在地形起伏剧烈或障碍物密集的区域,UUV需要频繁调整姿态以适应环境,导致跟踪

表 2 海底山丘轨迹跟踪数据

Tab. 2 Trajectory tracking data of submarine hills

算法种类	规划时间/ms	平均误差/m	误差标准差/m	最大误差/m	平均误差置信区间(95%)/m
经典 A*	670.54	0.491	0.208	1.009	0.483 ~ 0.498
混合 A*	216.70	0.568	0.317	0.970	0.556 ~ 0.579
RRT*	149.82	0.367	0.204	1.002	0.359 ~ 0.374
UUVA*	318.08	0.168	0.111	0.337	0.164 ~ 0.172



(a) 经典 A* 算法 (b) 混合 A* 算法

(c) RRT* 算法 (d) UUVA* 算法

图 13 海底峡谷规划路径

Fig. 13 Planning path of submarine canyon

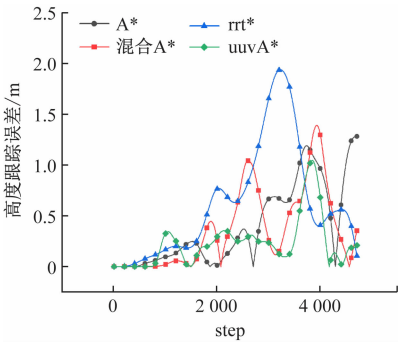


图 14 海底峡谷定高跟踪误差

Fig. 14 Altitude tracking error of submarine canyon

控制器的瞬时负担增加,从而引入一定误差。此外,UUVA* 算法中路径平滑权重、搜索分辨率等参数的选取对误差大小也具有直接影响。因此,虽然 UUVA* 整体表现优越,但在复杂环境中仍需结合地形特征与参数优化策略,才能最大程度降低跟踪误差。

综上,UUVA* 在复杂海底地形中能够生成平滑、连续且可有效跟随的路径,显著优于传统算法,从而提升了 UUV 的整体航行性能。

表 3 海底峡谷轨迹跟踪数据

Tab. 3 Trajectory tracking data of submarine canyon

算法种类	规划时间/ms	平均误差/m	误差标准差/m	最大误差/m	平均误差置信区间(95%)/m
A*	464.94	0.266	0.170	0.874	0.260 ~ 0.272
混合 A*	216.06	0.347	0.252	0.932	0.338 ~ 0.356
RRT*	154.07	0.331	0.202	0.780	0.324 ~ 0.338
UUVA*	234.77	0.067	0.040	0.171	0.065 ~ 0.068

4 结 论

本文针对 UUV 在复杂地形下的定高路径规划问题,提出了 UUVA*。该方法通过状态空间扩展和运动基元设计,有效结合了 UUV 的非完整约束;同时在代价函数中引入适航性优化,使生成的路径更加平滑且便于跟踪;再辅以多分辨率分层搜索与分支修剪机制,显著提高了搜索效率。仿真结果表明,本文方法在海底峡谷、海底山丘两类典型复杂地形中均能生成适航路径,并在轨迹跟踪精度与路径可行性方面优于经典 A* 与混合 A* 算法。

试验验证了所提算法在多种复杂环境下的稳定性能,体现出良好的普适性。由于所采用的运动基元设计与分层搜索机制具有一定的通用性,该方法不仅适用于 UUV 的定高规划,也为其他受非完整约束的自主体路径规划提供了参考。

需要指出的是,本文的研究仍存在一定局限:试验规模相对有限,环境主要假设为静态场景,且尚未在实际海试条件下进行验证。未来工作将围绕更大规模与动态环境下的三维路径规划展开,并进一步考虑环境不确定性和传感器噪声的影响,同时结合实际海试验证算法的工程适用性和鲁棒性,以推动其在真实任务中的应用。

参 考 文 献

[1] Yuan S Y, Li Y, Bao F W, et al. Marine environmental monitoring with unmanned vehicle platforms: present applications and future prospects[J]. Science of The Total Environment, 2023, 858(Part 1): 159741.

[2] Wynn R B, Huvenne V A I, Le Bas T P, et al. Autonomous Underwater Vehicles (AUVs): Their past, present and future contributions to the advancement of marine geoscience[J]. Marine Geology, 2014, 352: 451-468.

[3] Kasaya T. Usage of autonomous underwater vehicle and autonomous surface vehicle for marine geophysical exploration[J]. Geophysical Exploration, 2021, 74: 131-141.

[4] Zeng Z, Sammut K, Lammas A, et al. Imperialist competitive algorithm for AUV path planning in a variable ocean[J]. Applied Artificial Intelligence, 2015, 29(4):402-420.

[5] 张岳星,王轶群,李硕,等. 基于海图和改进粒子群优化算法的 AUV 全局路径规划[J]. 机器人,2020,42(1):120-128.
Zhang Y X, Wang Y Q, Li S, et al. Global path planning for AUV based on charts and the improved particle swarm optimization algorithm[J]. Robot, 2020, 42(1): 120-128. (in Chinese)

[6] Pan X, Feng G L, Huang L, et al. A new bilevel algorithm for UUV global path planning[J]. AIP Advances, 2024, 14(8):085311.

[7] Petres C, Pailhas Y, Patron P, et al. Path planning for autonomous underwater vehicles[J]. IEEE Transactions on Robotics, 2007, 23(2): 331-341.

[8] Zhang A, Wang W X, Bi W H, et al. A path planning method based on deep reinforcement learning for AUV in complex marine environment[J]. Ocean Engineering, 2024, 313(Part 1): 119354.

[9] Hou X Y, Du J, Wang J J, et al. AUV path planning with kinematic constraints in unknown environment using reinforcement learning[C]//Proceedings of the 2020 4th International Conference on Digital Signal Processing. 2020: 274-278.

[10] Scharff Willners J, Gonzalez-Adell D, Hernández J D, et al. Online 3-dimensional path planning with kinematic constraints in unknown environments using hybrid A* with tree pruning[J]. Sensors, 2021, 21(4): 1152.

[11] Liu L Q, Sun M, Zhao E J, et al. Three-dimensional dynamic trajectory planning for autonomous underwater robots under the PPO-IIFDS framework[J]. Journal of Marine Science and Engineering, 2025, 13(3): 445.

[12] Zhang W, Wang N X, Wu W H. A hybrid path planning algorithm considering AUV dynamic constraints based on improved A* algorithm and APF algorithm[J]. Ocean Engineering, 2023, 285(Part1): 115333.

[13] Yan Z P, Yu H M, Li B Y. Bottom-following control for an underactuated unmanned undersea vehicle using integral-terminal sliding mode control[J]. Journal of Central South University, 2015, 22(11): 4193-4204.

[14] Yan Z P, Hao L C, Wang Y Z, et al. A terrain-following control method for autonomous underwater vehicles with single-beam sensor configuration[J]. Journal of Marine Science and Engineering, 2024, 12(3): 366.

[15] Ru J Y, Yu H, Liu H, et al. A bounded near-bottom cruise trajectory planning algorithm for underwater vehicles[J]. Journal of Marine Science and Engineering, 2023, 11(1): 7.

[16] Yan Z P, Li J Y, Wu Y, et al. A real-time path planning algorithm for AUV in unknown underwater environment based on combining PSO and waypoint guidance[J]. Sensors, 2019, 19(1): 20.

[17] Zhang X, Wang Z Q, Chen H J, et al. A trajectory tracking and local path planning control strategy for Unmanned Underwater Vehicles[J]. Journal of Marine Science and Engineering, 2023, 11(12): 2230.

[18] 严浙平,周佳加. 水下无人航行器控制技术[M]. 北京:国防工业出版社,2015.
Yan Z P, Zhou J J. Control technology of unmanned underwater vehicle (UUV)[M]. Beijing: National Defense Industry Press, 2015. (in Chinese)

[19] 施小成,周佳加,边信黔,等. 模糊滑模变结构控制在 AUV 纵倾控制中的应用[J]. 计算机仿真,2008,25(10):168-171.
Shi X C, Zhou J J, Bian X Q, et al. Simulation of fuzzy sliding-mode control for AUV pitch control[J]. Computer Simulation, 2008, 25(10): 168-171. (in Chinese)

[20] 严浙平,晏正新,牟春晖,等. 基于反步法的欠驱动 UUV 的 3 维路径跟踪控制[J]. 信息与控制,2012,41(2):180-184.
Yan Z P, Yan Z X, Mou C H, et al. 3D path following control of underactuated UUV based on backstepping[J]. Information and Control, 2012, 41(2): 180-184. (in Chinese)

[21] Quigley M, Conley K, Gerkey B, et al. ROS: an open-source robot operating system[J]. IEEE Robotics & Automation Magazine, 2009, 15(1): 19.