

面向真实地形的水下航行器三维航路规划方法

耿正霖¹,程兴华²,包长春^{2*}

(1. 国防科技大学 电子科学学院, 湖南 长沙 410073; 2. 国防科技大学 气象海洋学院, 湖南 长沙 410073)

摘要:针对目前水下航行器航路规划结果适用性不强的问题,提出一种基于改进粒子群算法的水下航行器三维航路规划算法。基于海区真实地形数据构建维诺图,利用 Dijkstra 算法生成满足航行安全要求的初始航路集,作为粒子群算法的初始粒子,提高了初始粒子生成效率,同时改进粒子位置更新方法,根据航路上节点的毗邻节点的位置引导粒子位置变化,使得规划航路更加平滑,更适于水下航行器的航行。仿真结果表明,该算法较传统粒子群算法,规划航路适用性更好,且优化结果更稳健。

关键词:水下航行器;航路规划;平滑;粒子群优化算法

中图分类号:TP301.6 文献标志码:A 文章编号:1001-2486(2024)06-088-08



论文
拓展

3D path planning method for underwater vehicles based on real terrain

GENG Zhenglin¹, CHENG Xinghua², BAO Changchun^{2*}

(1. College of Electronic Science and Technology, National University of Defense Technology, Changsha 410073, China;

2. College of Meteorology and Oceanography, National University of Defense Technology, Changsha 410073, China)

Abstract: Aiming at the problem of poor applicability of current path planning results for underwater vehicles, a three-dimensional path planning algorithm for underwater vehicles based on improved particle swarm optimization algorithm was proposed. A Voronoi map based on the real terrain data of sea area was built, and the Dijkstra algorithm was used to generate the initial path set that meets the requirements of navigation safety as the initial particles of particle swarm optimization algorithm, so the generation efficiency of initial particles was improved. At the same time, the particles' position update method was improved to guide the particle position change according to the position of adjacent nodes on the route, which makes the planned route smoother and more suitable for the navigation of underwater vehicles. Simulation results show that the proposed algorithm is more robust and has better applicability than the traditional particle swarm optimization algorithm in path planning.

Keywords: underwater vehicle; path planning; smooth; particle swarm optimization algorithm

水下航行器在军事和民用领域都发挥着重要作用^[1]。对于执行水下探测搜索任务的水下航行器,通常具有明确的目的地,需要在出发位置和到达位置间选择较优的航行路线。而水下环境复杂,对航行器的航行造成一定的安全威胁,同时航行器的航行受其机动性能的限制,需要考虑水平转向角度、爬升/下降角度、续航能力等诸多因素。如何为水下航行器规划出安全、可行、经济的航行路径一直是水下航行器研究的热点。

针对水下航行器航路规划问题,不同的研究

者提出了不同的环境模型及对应的建模方法^[2],例如栅格法、维诺(Voronoi)图^[3-4]法等;针对不同模型,采用不同的优化算法实现路径的规划,主要算法包括 Dijkstra 算法^[5-6]、A* 算法^[7-9]、人工势场法^[10-11]等传统算法和遗传算法^[12]、粒子群算法^[13-14]、蚁群算法^[15-16]、神经网络^[17-18]、强化学习^[19]等智能算法及其变形改进算法^[20-21]。其中,传统算法通常具有较快的计算速度,但是大部分基于栅格模型,规划航路平滑度较差,且难以满足多约束条件下的优化需求。而智能优化算法通

收稿日期:2023-01-04

基金项目:国家自然科学基金资助项目(61702531)

第一作者:耿正霖(1988—),男,云南昆明人,助理研究员,博士,E-mail:oliver8812@163.com

*通信作者:包长春(1979—),男,吉林镇赓人,副教授,博士,E-mail:baochangchun1979@163.com

引用格式:耿正霖,程兴华,包长春.面向真实地形的水下航行器三维航路规划方法[J].国防科技大学学报,2024,46(6):88-95.

Citation:GENG Z L, CHENG X H, BAO C C. 3D path planning method for underwater vehicles based on real terrain[J]. Journal of National University of Defense Technology, 2024, 46(6): 88-95.

常具有较好的优化效果,但计算相对复杂。

在众多算法中,粒子群优化算法简单易于实现,广泛应用于模糊控制、函数优化等领域。但在与航路规划结合的过程中,存在规划路径曲折,可执行性较差;障碍物模型较为理想,与真实海底地形存在差异;大多用于二维航路的规划,难以满足三维航行需求等问题。因此,针对上述问题,本文面向真实海底地形,建立水下航行器三维航路模型及对应的评价函数,提出一种基于改进粒子群算法的水下航行器三维路径规划方法。

1 航路模型

1.1 航路表达方式

对于水下航行器,其可在水中进行水平方向和垂直方向上的运动,实现在三维空间内的位置变化。常用的二维路径规划结果限制了水下航行器在垂直方向的运动自由度,难以满足实际需求。采用三维坐标来描述水下航行器的航路,在确定起点和终点的情况下,可以通过一系列节点的三维坐标即水下航行器航行方向发生变化的坐标点来表示。如对于一条 N 个节点的路径,可表示为:

$$\mathbf{P} = \begin{bmatrix} x_1 & x_2 & \cdots & x_N \\ y_1 & y_2 & \cdots & y_N \\ z_1 & z_2 & \cdots & z_N \end{bmatrix} \quad (1)$$

矩阵的第 n 列就表示路径上第 n 个节点的三维坐标, x, y, z 分别为三个方向的坐标值。那么整条路径可分为 $N-1$ 段,第 n 和 $n+1$ 个节点间航路为第 n 段航路, $1 \leq n \leq N-1$ 。根据节点坐标,可以确定各段航行路径水平方向和垂直方向的航行角度及其变化情况。第 n 段航路水平方向和垂直方向角度可由节点坐标进行计算,即:

$$\alpha_n = \arctan\left(\frac{y_{n+1} - y_n}{x_{n+1} - x_n}\right) \quad (2)$$

$$\beta_n = \arctan\left(\frac{z_{n+1} - z_n}{\sqrt{(y_{n+1} - y_n)^2 + (x_{n+1} - x_n)^2}}\right) \quad (3)$$

1.2 航路评价指标

水下航行器运行环境复杂,规划航路必须能有效避开水下障碍物,且要符合水下航行器的航行特性和水下航行器的续航能力要求等。据此,建立航路评价指标,其包含的评价因素主要有:航路的可执行性、经济性和安全性。再根据不同因素的权重建立航路的综合评价指标。

1.2.1 可执行性

航路可执行性主要指水下航行器沿该航路航行是否符合水下航行器的机动性能限制,即航行过程中不能有急剧的方向变化,包括水平方向和垂直方向,航路上角度变化越小,变化越平缓,可执行性越好。

设水平方向角度变化门限为 Th_α ,第 n 段和第 $n+1$ 段之间的水平转角可以用 $|\alpha_{n+1} - \alpha_n|$ 表示,水平方向转角应小于相应门限, $|\alpha_{n+1} - \alpha_n| < Th_\alpha$ 。一条路径水平方向可执行性用水平转角变化的最大值来衡量:

$$f_\alpha = \max(|\alpha_{n+1} - \alpha_n|) / (2\pi) \quad (4)$$

同时,航行器垂直方向上的角度及其变化量也有一定的限制。首先,垂直方向爬升或下降不能超过一定门限,用 Th_{β_1} 表示;其次,角度变化量也不能超过一定限度,用 Th_{β_2} 表示。 β_n 表示第 n 段航路垂直方向的角度,则第 n 段和第 $n+1$ 段之间的垂直方向角度变化可以用 $|\beta_{n+1} - \beta_n|$ 表示,则用式(5)描述垂直方向角度大小对可执行性的评价方法,用式(6)描述垂直方向角度变化大小对可执行性的评价方法。

$$f_{\beta_1} = \begin{cases} \max(|\beta_n| / Th_{\beta_1}) & |\beta_n| < Th_{\beta_1} \\ 1 & |\beta_n| \geq Th_{\beta_1} \end{cases} \quad (5)$$

$$f_{\beta_2} = \max(|\beta_{n+1} - \beta_n|) / (2\pi) \quad (6)$$

那么,垂直方向可执行即可通过两方面进行综合评价:

$$f_\beta = \mu_1 \cdot f_{\beta_1} + \mu_2 \cdot f_{\beta_2} \quad (7)$$

其中, μ_1 和 μ_2 表示两个指标的权重。

1.2.2 经济性

水下航行器供能有限,在确定起始位置和终点位置的情况下,路径越短,通常消耗能量越少,故以路径长短进行航路经济性的评价,航路越短,经济性越好。整条航路的航行距离,可以通过如下计算:

$$L = \sum_{n=1}^{N-1} \sqrt{(x_{n+1} - x_n)^2 + (y_{n+1} - y_n)^2 + (z_{n+1} - z_n)^2} \quad (8)$$

用整条航路的距离与起点终点的直线距离比作为评价标准,即:

$$f_L = \frac{\sum_{n=1}^{N-1} \sqrt{(x_{n+1} - x_n)^2 + (y_{n+1} - y_n)^2 + (z_{n+1} - z_n)^2}}{\sqrt{(x_N - x_1)^2 + (y_N - y_1)^2 + (z_N - z_1)^2}} \quad (9)$$

路径距离越短, f_L 越接近 1,航路经济性越好。

1.2.3 安全性

水下地形复杂,起伏不定,水下航行器必须避

开海山等障碍物,才能保证其能安全到达目的地,执行相关任务。故航路的安全性是最重要的评价指标之一,航路必须绕开障碍物,避免发生触底、碰撞事故。

航路的安全性主要考虑三个方面:航路离海底的距离;航路离水面的距离;航路侧向离障碍物的距离。距海底过近可能导致触底事故,距水面较近则容易暴露自身位置,同时需要设定航道的安全距离,否则也易导致航行绕过障碍物时触壁。为此,构建航路模型如图 1 所示。

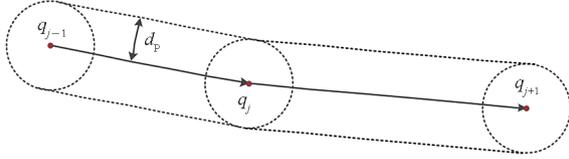


图 1 航路模型
Fig. 1 Path model

图 1 中: q_{j-1} 、 q_j 、 q_{j+1} 表示路径上的节点;黑色实线表示路径; d_p 表示航道宽度,通过其限定航线距两侧障碍物的距离;虚线覆盖区域即为航道区域,通过比较航路高程和对应航道区域内海底高程,确定是否满足航行安全性需求。

对于某一段航路,用 D_{pt} 表示航道覆盖区域地形的最大高程,这里认为海平面的高程为 0,海平面以下高程为负值,海平面以上高程为正值。用 Th_{DptL} ($Th_{DptL} \geq 0$) 表示距海底深度余量,即水下航行器距海底的最小距离;用 Th_{DptH} ($Th_{DptH} \geq 0$) 表示距水面的深度余量,即水下航行器距海面的最小距离。用 $Path_L$ 和 $Path_H$ 分别表示该段航路的最小高程和最大高程,各参数含义如图 2 所示。

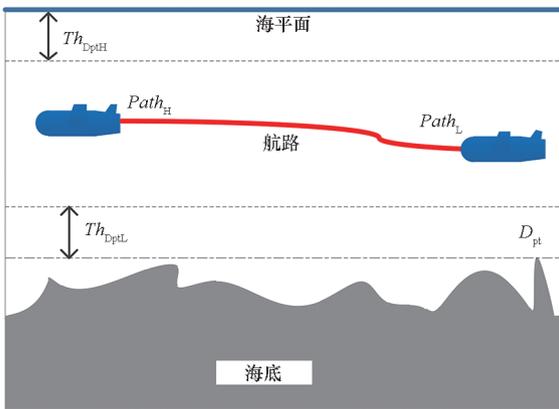


图 2 航路安全性参数示意图
Fig. 2 Security parameters of a path

用 c_{Dpt1} 和 c_{Dpt2} 表示该段航路距离海底和海面是否在安全范围内,“0”表示安全,“1”表示不

安全。

$$c_{Dpt1} = \begin{cases} 0 & Path_L > D_{pt} + Th_{DptL} \\ 1 & Path_L \leq D_{pt} + Th_{DptL} \end{cases} \quad (10)$$

$$c_{Dpt2} = \begin{cases} 0 & Path_H \leq -Th_{DptH} \\ 1 & Path_H > -Th_{DptH} \end{cases} \quad (11)$$

对于 N 个节点的航路, $f_{Dpt1} = \frac{1}{N-1} \sum_{n=1}^{N-1} c_{Dpt1,n}$, $f_{Dpt2} = \frac{1}{N-1} \sum_{n=1}^{N-1} c_{Dpt2,n}$, 下标 n 表示航路段。航路安全性的评价指标可表示为:

$$f_\gamma = \mu_3 \cdot f_{Dpt1} + \mu_4 \cdot f_{Dpt2} \quad (12)$$

其中, μ_3 和 μ_4 表示两个指标的权重。通过分析,即可对水下航行器航路实现综合评价,可用下式计算:

$$f = \omega_1 f_\alpha + \omega_2 f_\beta + \omega_3 f_L + \omega_4 f_\gamma \quad (13)$$

其中, $\omega_1, \omega_2, \omega_3, \omega_4$ 分别为不同因素的权重。

2 优化算法

2.1 粒子群算法及粒子表示方式

粒子群优化 (particle swarm optimization, PSO) 算法于 1995 年提出,目前已广泛应用于路径规划、函数优化等领域。在粒子群算法中,每个个体称为一个粒子。粒子通过在搜索空间内以一定的速度运行,寻找最优位置。搜索过程中,根据单个粒子自身搜索到的历史最优位置和种群内其他粒子的历史最优位置进行速度和位置的调整。

其中,用 $p_i = [p_{i,1}, p_{i,2}, \dots, p_{i,N}]$ 表示粒子 i 的位置矢量, $v_i = [v_{i,1}, v_{i,2}, \dots, v_{i,N}]$ 表示粒子 i 的运行速度, $p_i^b = [p_{i,1}^b, p_{i,2}^b, \dots, p_{i,N}^b]$ 表示粒子 i 搜索到的最优位置, $g^b = [g_1^b, g_2^b, \dots, g_N^b]$ 表示整个粒子种群搜索到的最优位置。

粒子速度和位置迭代方程如下:

$$v_i^{k+1} = wv_i^k + c_1 r_1 (p_i^b - p_i^k) + c_2 r_2 (g^b - p_i^k) \quad (14)$$

$$p_i^{k+1} = p_i^k + v_i^{k+1} \quad (15)$$

其中: i 为粒子序号; k 为迭代次数; r_1 和 r_2 为随机数,取值范围 $0 \sim 1$; w 为惯性因子,控制粒子当前速度对后续速度的影响程度,起平衡全局搜索和局部搜索的作用; c_1 和 c_2 称为学习因子,通常为正的常数,控制粒子自我总结能力和向种群中最优个体学习的能力。粒子的优劣则通过适应度函数进行评价,上一节对航路的综合评价结果即为对粒子评价的适应度函数,该值越小,对应粒子越优。

对于三维路径,每个节点采用其三维坐标描述,根据航路模型,第*i*个粒子,即第*i*条路径可表示为:

$$\mathbf{p}_i = \begin{bmatrix} x_{i,1} & x_{i,2} & \cdots & x_{i,N} \\ y_{i,1} & y_{i,2} & \cdots & y_{i,N} \\ z_{i,1} & z_{i,2} & \cdots & z_{i,N} \end{bmatrix} \quad (16)$$

对应地,速度矢量 \mathbf{v}_i 也有相同的维度。

$$\mathbf{v}_i = \begin{bmatrix} v_{i,1}^x & v_{i,2}^x & \cdots & v_{i,N}^x \\ v_{i,1}^y & v_{i,2}^y & \cdots & v_{i,N}^y \\ v_{i,1}^z & v_{i,2}^z & \cdots & v_{i,N}^z \end{bmatrix} \quad (17)$$

其中, $\mathbf{p}_{i,j} = [x_{i,j}, y_{i,j}, z_{i,j}]^T$ 表示第*i*条路径的第*j*个节点的三维坐标, $\mathbf{v}_{i,j} = [v_{i,j}^x, v_{i,j}^y, v_{i,j}^z]^T$ 表示第*i*条路径的第*j*个节点的三维速度, v^x 、 v^y 、 v^z 表示速度在三个方向上的分量。

2.2 初始航路生成

对于粒子群优化算法,需要生成一定数量的初始粒子,即初始路径集。初始路径集的合理性会直接影响最终的优化效果。而在复杂海洋地形条件下,若无指导性信息,初始路径极大可能需要经过参数的遍历或随机选择产生,耗时较长且可能产生不合理的结果,影响路径规划算法的时效性和适应性。为此,文中提出一种基于 Voronoi 图的航路生成方法,根据地形约束生成对应区域的 Voronoi 图,一定程度上减小了航路选择空间,并利用 Dijkstra 算法生成可行的初始路径,为后续优化提供初值。

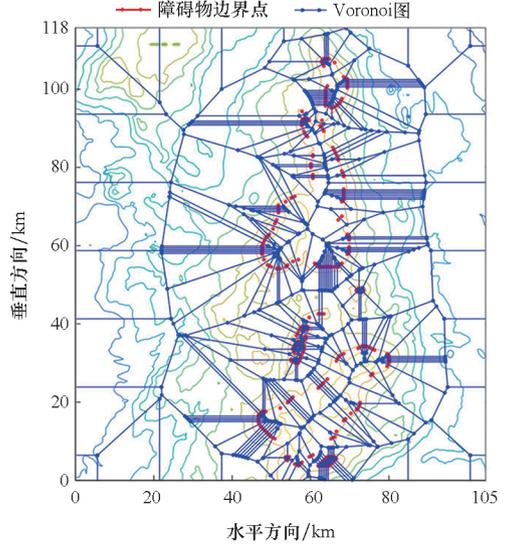
利用 Voronoi 图,可以产生避开障碍物的可行节点集以及对应的连通关系,基于 Voronoi 图产生的路径通常能远离障碍物,但路径不平滑,而作为后续优化的初始路径,平滑问题将在优化过程中得到解决,其可行性很好地满足了对初始路径的需求。同时,基于 Dijkstra 算法,可以得到 Voronoi 图下的最短路径,很好地满足了其对初始路径的需求。

生成方式如下:

- 1) 根据海区的地形数据,生成一定深度下的二维地图,得到障碍边界点。
- 2) 根据障碍物的边界点,生成对应区域的 Voronoi 图。
- 3) 基于 Voronoi 图,利用 Dijkstra 算法规划出可行航路,通过改变 Voronoi 图中节点间距离权重,得到不同的航路,作为后续优化的可选航路集。
- 4) 根据设定起点和终点以及上述得到的可

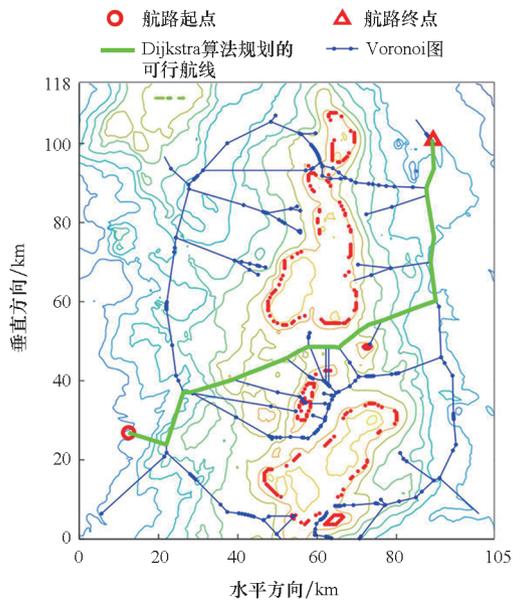
选航路,生成初始航路集。

图3即为根据实际海区地形生成的 Voronoi 图和根据上述方法得到的初始路径。所得到的航路为二维航路,此时根据起点和终点的深度坐标,设置航路中间节点的深度即可作为三维航路的初始值。



(a) 根据地形生成的 Voronoi 图

(a) Voronoi diagram based on terrain



(b) 初始路径

(b) Initial path

图3 海区地形 Voronoi 图及初始路径

Fig.3 Voronoi diagram of the sea area terrain and initial path

2.3 改进的粒子群优化算法

传统的粒子群算法在粒子位置的调整上只受最优个体的影响,而没有有效的调整策略。本算法鉴于航行器水平和俯仰方向上机动角度尽可能

小的需求,提出一种基于毗邻节点位置的粒子位置调整方法。以连续的 3 个节点的坐标关系计算中间节点的优化方向,改进粒子位置的调整方式,图 4 为路径的矢量模型。

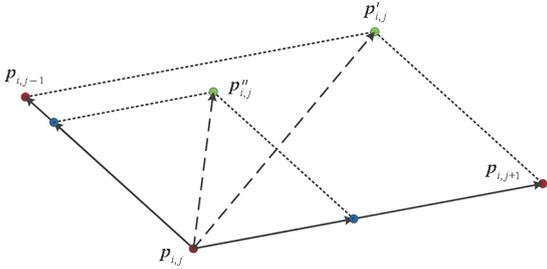


图 4 路径矢量模型

Fig. 4 Path vector model

图中, $p_{i,j-1}$ 、 $p_{i,j}$ 和 $p_{i,j+1}$ 表示路径 p_i 上的 3 个连续节点,为了减小整条路线水平方向和垂直方向的角度变化,通过 3 个连续节点之间的位置关系确定中间节点的调整方向,即粒子的位置调整情况。以中间节点为起点,做其指向前后节点的矢量,即 $(p_{i,j-1} - p_{i,j})$ 和 $(p_{i,j+1} - p_{i,j})$,以两矢量加权的方向为该点的位置调整方向,设置加权因子分别为 c_3 和 c_4 , c_3 和 c_4 为 0~1 之间的随机数,控制调整矢量的方向,调整比例因子为 h ,控制调整矢量的大小。图中,矢量 $(p''_{i,j} - p_{i,j})$ 和 $(p'_{i,j} - p_{i,j})$ 表示节点 $p_{i,j}$ 位置可能的调整矢量。则该点的位置变化值可表示为:

$$u_{i,j} = \begin{cases} h[c_3(p_{i,j-1} - p_{i,j}) + c_4(p_{i,j+1} - p_{i,j})] & 2 \leq j \leq N-1 \\ [0, 0, 0]^T & \text{其他} \end{cases} \quad (18)$$

则 $u_i = [u_{i,1}, u_{i,2}, \dots, u_{i,N}]$ 。

基于上述粒子位置调整方法,提出新的粒子位置更新方法:

$$p_i^{k+1} = p_i^k + v_i^{k+1} + u_i^k \quad (19)$$

u_i^k 更新只与当前粒子位置有关,即:

$$u_{i,j}^k = \begin{cases} h[c_3(p_{i,j-1}^k - p_{i,j}^k) + c_4(p_{i,j+1}^k - p_{i,j}^k)] & 2 \leq j \leq N-1 \\ [0, 0, 0]^T & \text{其他} \end{cases} \quad (20)$$

粒子速度更新方式不变,如式(14)所示。

3 仿真结果

以海区地形数据为基础数据,对算法进行仿真验证。设定水下航行器的出发位置和目标位置,采用上述改进算法进行路径的优化,并与采用传统粒子位置更新方法(如式(15)所示)、传统遗传算法以及基于模拟退火算法^[22]改进的遗传算法(后文简称为遗传退火算法)的优化结

果进行比较,仿真中采用相同的评价函数,只是优化方法存在差异。下文仿真了 2 个不同的场景,每个场景进行 500 次仿真。具体场景参数如表 1 所示。

表 1 不同场景航路参数

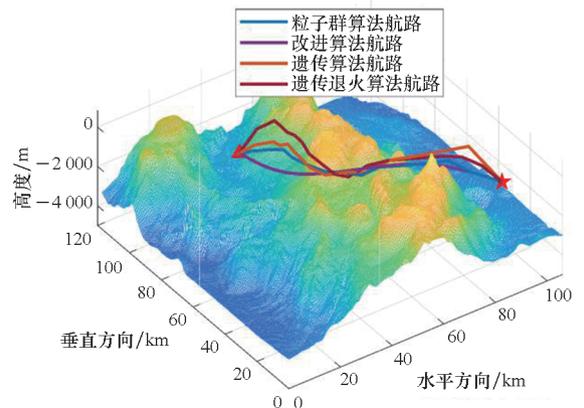
Tab. 1 Path parameters in different scenarios

序号	起点坐标/km	终点坐标/km	路径节点数
1	(37.42, 91.58, -0.12)	(95.69, 21.32, -0.14)	16
2	(12.44, 26.87, -0.20)	(89.25, 100.80, -0.24)	16

粒子群相关算法中,惯性因子 $w = 0.4$ 、学习因子 $c_1 = 1$ 、 $c_2 = 0.2$,设置粒子数量为 40,迭代次数为 200。整条路径包含 16 个节点。 μ_1 、 μ_2 、 μ_3 、 μ_4 分别取 0.5、0.5、0.6 和 0.4, ω_1 、 ω_2 、 ω_3 、 ω_4 分别取 0.25、0.25、0.1 和 0.4。遗传算法采用保留、交叉和变异 3 种方式对个体进行更新,最优的 20% 个体直接保留进入下一代,70% 子代个体通过父代个体交叉产生,10% 通过父代个体随机选择产生,在非保留产生的个体中,以一定比例发生变异。在遗传退火算法中,子代个体的产生方法与遗传算法相同,但在个体变异时,引入退火算法原理,以一定概率选择非更优个体进入下一代。

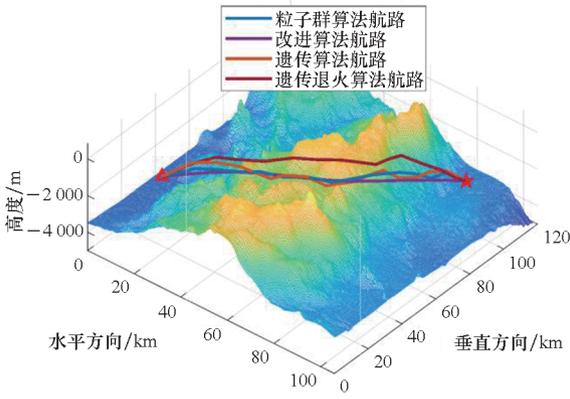
仿真了 2 种不同场景下得到的最优路径,如图 5 所示。图 5(a)为场景 1 下改进算法和其他几种算法规划出的航路,图 5(b)为场景 2 下改进算法和其他几种算法规划出的航路。可以直观看出,在不同场景下,改进算法规划出的结果比其他几种方法规划航路更平滑,航路变化幅度更小。

同时对不同场景进行 500 次仿真,得到适应度均值和标准差随迭代次数的变化情况以及最终



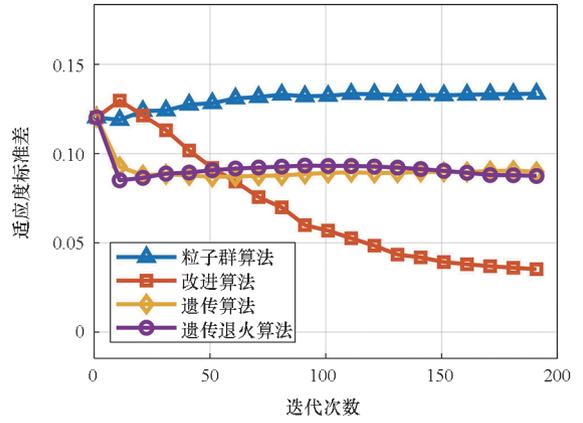
(a) 场景 1 航路仿真结果

(a) Path simulation results in scenario 1



(b) 场景2 航路仿真结果

(b) Path simulation results in scenario 2



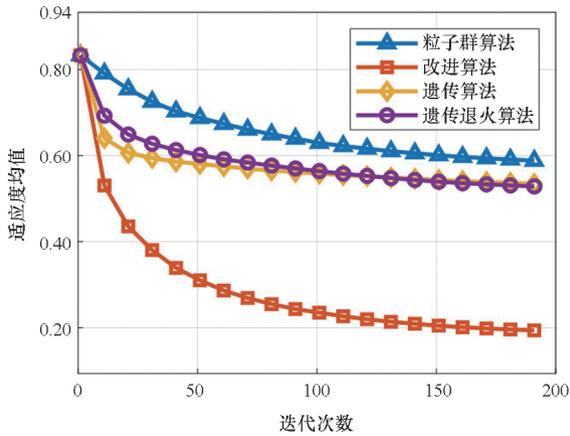
(c) 场景1 适应度标准差随迭代次数变化情况

(c) The average fitness standard deviation changes with the number of iterations in scenario 1

图5 不同场景航路仿真结果

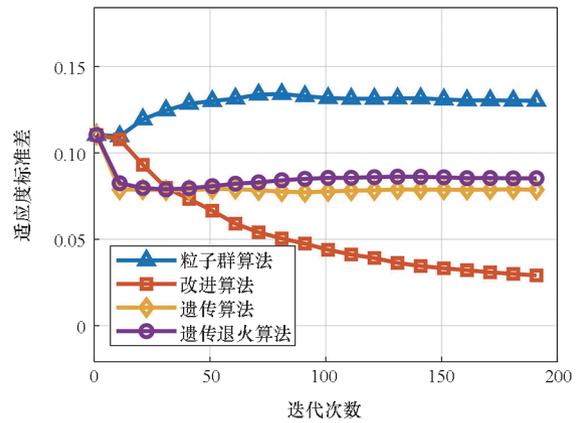
Fig.5 Path simulation results in different scenarios

适应度值的统计直方图,如图6和图7所示。并对优化结果进行统计,比较本文改进算法与其他几种方法优化结果的均值和标准差。



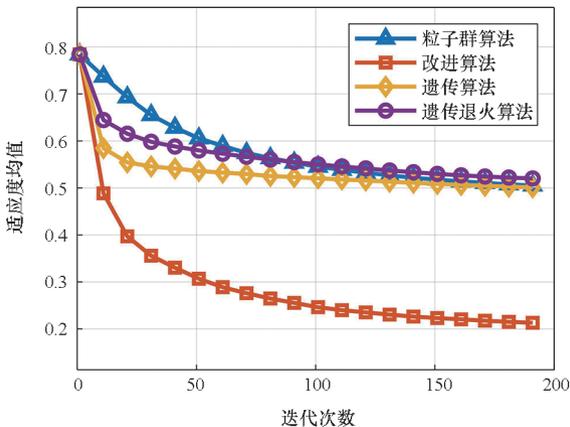
(a) 场景1 适应度均值随迭代次数变化情况

(a) The average fitness value changes with the number of iterations in scenario 1



(d) 场景2 适应度标准差随迭代次数变化情况

(d) The average fitness standard deviation changes with the number of iterations in scenario 2



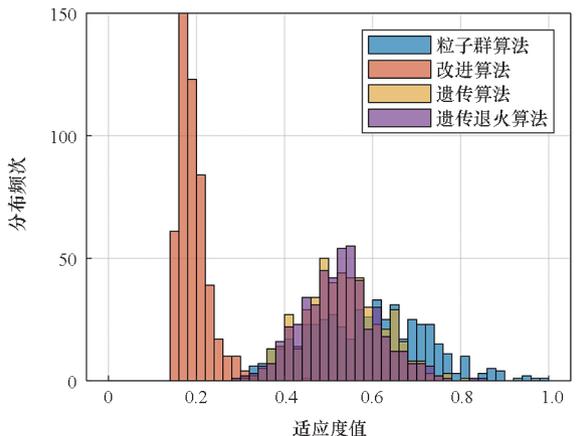
(b) 场景2 适应度均值随迭代次数变化情况

(b) The average fitness value changes with the number of iterations in scenario 2

图6 优化结果的收敛情况

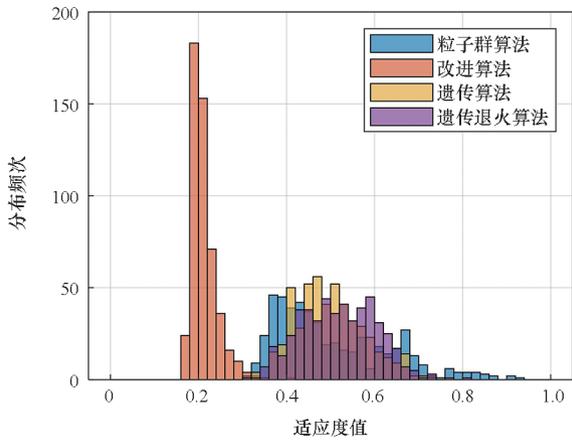
Fig.6 Convergence of optimization results

由图6可以看出,不同场景下不同算法适应度均值均随迭代次数的增加而降低,说明航路在不断优化,但明显看出,改进算法优化结果的适应度下降最快,且最终结果远小于其他几种方法,



(a) 场景1 适应度分布

(a) Fitness distribution in scenario 1



(b) 场景 2 适应度分布

(b) Fitness distribution in scenario 2

图 7 不同场景下适应度分布

Fig. 7 Fitness distribution in different scenarios

说明其优化效果最好。同时可以看到,不同场景下 4 种算法适应度的标准差随迭代次数的增加呈现不同的变化趋势。传统粒子群算法结果的适应度标准差随迭代次数的增加缓慢增加,说明该方法得到的结果分散,收敛性差;2 种遗传算法结果的标准差随迭代次数逐渐下降,说明其结果分布较为集中,收敛性较好;而改进算法结果的适应度标准差随迭代次数的增加略微上升后迅速下降,最终远低于其他 3 种算法,说明改进算法得到的结果分布最集中,收敛性最好。从图 7 中可以明显看出,在仿真的 4 种方法中,改进算法优化结果比其他几种方法更好,且分布更集中,即结果最稳定,算法稳健性最好。对应的优化统计结果如表 2 所示,改进算法在两种场景下优化结果的均值和方差最小,说明其优化效果最好,波动最小。

表 2 不同场景优化统计结果

Tab. 2 Optimization statistics results of different scenarios

算法	场景 1	场景 2
粒子群算法	0.586 3 ± 0.133 6	0.503 6 ± 0.129 7
改进算法	0.192 6 ± 0.035 0	0.211 8 ± 0.028 7
遗传算法	0.532 9 ± 0.090 2	0.499 4 ± 0.078 7
遗传退火算法	0.526 6 ± 0.087 4	0.518 8 ± 0.085 0

4 结论

本文提出一种基于改进粒子群算法的水下航行器航路规划算法,该算法引入方向调整因子,提出一种基于毗邻节点位置的粒子群位置调整方法,算法显著改善了规划路径平滑度,在相同的优

化迭代次数下,本文算法得到的航路更加平滑,更适于水下航行器的航行。基于地形数据建立 Voronoi 图,利用 Dijkstra 算法进行初始航路的快速生成,提升算法规划速率。从仿真结果看,航路的适应度普遍优于传统方法,且优化结果的分布更加集中,即优化结果更为稳健。

参考文献 (References)

[1] 吕重阳. 水下航行器路径规划关键技术研究[D]. 哈尔滨: 哈尔滨工程大学, 2019.
 LYU C Y. Research on key technologies of path planning for underwater vehicle [D]. Harbin: Harbin Engineering University, 2019. (in Chinese)

[2] 郭银景, 孟庆良, 孔芳, 等. AUV 路径规划算法研究现状与展望[J]. 计算机科学与探索, 2020, 14(12): 1981 - 1994.
 GUO Y J, MENG Q L, KONG F, et al. Research status and prospect of AUV path planning algorithms [J]. Journal of Frontiers of Computer Science and Technology, 2020, 14(12): 1981 - 1994. (in Chinese)

[3] AYAWLI B B K, MEI X, SHEN M Q, et al. Mobile robot path planning in dynamic environment using Voronoi diagram and computation geometry technique [J]. IEEE Access, 2019, 7: 86026 - 86040.

[4] HAR-PELED S, VARADHARAJAN R. Shortest secure path in a Voronoi diagram[EB/OL]. (2021 - 03 - 19) [2022 - 12 - 12]. <https://arxiv.org/pdf/2007.09773>.

[5] KONAR A, CHAKRABORTY I G, SINGH S J, et al. A deterministic improved Q-learning for path planning of a mobile robot[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2013, 43(5): 1141 - 1153.

[6] 姜辰凯, 李智, 盘书宝, 等. 基于改进 Dijkstra 算法的 AGVs 无碰撞路径规划[J]. 计算机科学, 2020, 47(8): 272 - 277.
 JIANG C K, LI Z, PAN S B, et al. Collision-free path planning of AGVs based on improved Dijkstra algorithm[J]. Computer Science, 2020, 47(8): 272 - 277. (in Chinese)

[7] KIADI M, GARCÍA E, VILLAR J R, et al. A* -based co-evolutionary approach for multi-robot path planning with collision avoidance [J]. Cybernetics and Systems, 2023, 54(3): 339 - 354.

[8] SANG T T, XIAO J C, XIONG J F, et al. Path planning method of unmanned surface vehicles formation based on improved A* algorithm[J]. Journal of Marine Science and Engineering, 2023, 11(1): 176.

[9] 张韬, 项祺, 郑婉文, 等. 基于改进 A* 算法的路径规划在海战兵棋推演中的应用[J]. 兵工学报, 2022, 43(4): 960 - 968.
 ZHANG T, XIANG Q, ZHENG W W, et al. Application of path planning based on improved A* algorithm in war gaming of naval warfare [J]. Acta Armamentarii, 2022, 43(4): 960 - 968. (in Chinese)

[10] JO H J, KIM S R, KIM J H, et al. Comparison of velocity obstacle and artificial potential field methods for collision avoidance in swarm operation of unmanned surface vehicles[J]. Journal of Marine Science and Engineering, 2022, 10(12): 2036.

[11] WU D H, WEI L S, WANG G L, et al. APF-IRRT* : an

- improved informed rapidly-exploring random trees-star algorithm by introducing artificial potential field method for mobile robot path planning [J]. *Applied Sciences*, 2022, 12(21): 10905.
- [12] LIN L, GOODRICH M A. UAV intelligent path planning for wilderness search and rescue [C]//Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009: 709 - 714.
- [13] WANG Y B, BAI P, LIANG X L, et al. Reconnaissance mission conducted by UAV swarms based on distributed PSO path planning algorithms [J]. *IEEE Access*, 2019, 7: 105086 - 105099.
- [14] LIM H S, FAN S S, CHIN C K H, et al. Particle swarm optimization algorithms with selective differential evolution for AUV path planning [J]. *IAES International Journal of Robotics and Automation (IJRA)*, 2020, 9(2): 94 - 112.
- [15] CHEN Y L, BAI G Q, ZHAN Y, et al. Path planning and obstacle avoiding of the USV based on improved ACO-APF hybrid algorithm with adaptive early-warning [J]. *IEEE Access*, 2021, 9: 40728 - 40742.
- [16] KONATOWSKI S. Application of the ACO algorithm for UAV path planning [J]. *Przełąd Elektrotechniczny*, 2019, 1(7): 117 - 121.
- [17] SHIRI H, PARK J, BENNIS M. Remote UAV online path planning via neural network-based opportunistic control [J]. *IEEE Wireless Communications Letters*, 2020, 9(6): 861 - 865.
- [18] LI S, GUO Y. Neural-network based AUV path planning in estuary environments [C]//Proceedings of the 10th World Congress on Intelligent Control and Automation, 2012.
- [19] WU J H, SONG C X, MA J, et al. Reinforcement learning and particle swarm optimization supporting real-time rescue assignments for multiple autonomous underwater vehicles [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2022, 23(7): 6807 - 6820.
- [20] LIANG Y, WANG L D. Applying genetic algorithm and ant colony optimization algorithm into marine investigation path planning model [J]. *Soft Computing*, 2020, 24(11): 8199 - 8210.
- [21] ZHU T T, ZHU D Q, YAN M Z. Multiple underwater target search path planning based on GBNN [C]//Proceedings of the International Conference on Intelligent Robotics and Applications, 2019: 225 - 232.
- [22] 赵炳巍, 贾峰, 曹岩, 等. 基于模拟退火算法的人工势场法路径规划研究 [J]. *计算机工程与科学*, 2022, 44(4): 746 - 752.
- ZHAO B W, JIA F, CAO Y, et al. Path planning of artificial potential field method based on simulated annealing algorithm [J]. *Computer Engineering & Science*, 2022, 44(4): 746 - 752. (in Chinese)