Local Path Planning for Unmanned Surface Vehicles based on Hybrid A* and B-spline

Liang Zhao
College of Civil Engineering and Architecture, Zhejiang University
Hangzhou, China
liamzhao13@zju.edu.cn

Ruyin Mao
College of Civil Engineering and Architecture, Zhejiang University
Hangzhou, China
22012067@zju.edu.cn

Yong Bai
College of Civil Engineering and Architecture, Zhejiang University
Hangzhou, China
baiyong@zju.edu.cn

***Abstract*— Unmanned Surface Vehicles (USVs) have witnessed a vigorous growth in the past decades and have been applied in various applications in both commercial and military domains. Central to the control of USVs, local path planning is one of the crucial technologies in the process towards autonomy. This paper investigates the application of a novel path planning algorithm in combination with modified Hybrid A* and several path manipulators. First, the heuristic strategy and node expansion method of standard Hybrid A* have been adapted for the algorithm. Second, node cutting technique and B-spline path smoother are applied to enhance the path quality. Simulations have been carried out to illustrate the excellent performance of the proposed method comparing to several state-of-the-art methods. Furthermore, the new algorithm is tested in the USV model, and the results have demonstrated that it can perfectly coordinate with the USV control system. Therefore, the proposed algorithm can be considered as a reliable method dealing with local path planning problem for USV.**

Keywords—path planning, unmanned surface vehicles, hybrid A*, path smoother

I. Introduction

USVs can be defined as unmanned surface vessels which can perform missions in a variety of cluttered environments without any human intervention [1]. With the features of high-level autonomy and coordinated cooperation, they have been extensively applied in ocean research [2], ocean resources exploration [3], and maritime transport or rescue [4]. To perform tasks more effectively, path planning technology has played a crucial part in both industrial application and scientific research.

In general, path planning algorithms for USV can be sorted as global approaches and local approaches. The first class has prior knowledge about the environment modeled as a map. It is often expensive for implementation and not suitable for real application because global environment information is not always available a priori. However, the local approaches, where the path is generated by taking data from the sensors during the mission, enable the USV to respond to a new environment effectively. This method is more complicated in design but more applicable in practice.

Many of the previous studies on local path planning algorithm development for various applications have been widely conducted in the past years. The work in [5] used two modified ant colony optimization methods considering multiple objectives to solve path planning problems. The results have shown each of the proposed approaches can be considered for path planning depending the application needs. To obtain a smooth time-optimal path for USVs, [6] proposed a control-based method which is called rapid-Gauss pseudo-spectral method (RGPM). The simulation results have presented that RGPM is superior to the traditional GPM in terms of path quality and calculation speed. Moreover, [7] combining the improved A* with artificial potential field to address path planning problem for USV formation fleet. The algorithm can greatly reduce the probability of falling into the local optimum situation, and is validated to be effective in different environments. [8] proposed an improved BA* (IBA*) algorithm with the unit decomposition method and map update method to address the problem of insufficient continuity and high-resolution environment modeling. Experiments on a real unmanned surface mapping vehicle have been conducted to illustrate the superior performance of IBA*. Work presented in [9] used improved artificial fish swarm algorithm with B-spline to solve path planning. Simulations have been carried out to demonstrate the combination of the methods can plan routes effectively in the complex marine environments.

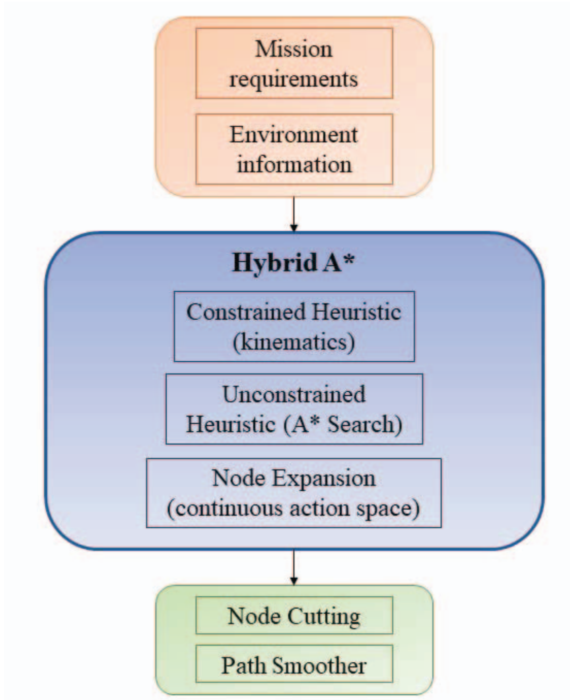

Fig. 1. Scheme of the proposed method

Most of the previous works to solve the local path planning problem are to find a collision-free and shortest path, but the solution provides no guarantee of feasibility that the surface vessel can execute in the real world, accompanied with bad path quality and unnecessary turn [1] [10]. Moreover, the other issue seldom considered is that many algorithms that appear to be efficient theoretically are not applied and tested in the USV control system, which means they are not convincing in handling real-world situation [10-11]. These challenging issues provide the

2022 IEEE International Conference on Unmanned Systems (ICUS) | 978-1-6654-8456-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICUS55513.2022.9986811

main motivation for this research. In this paper, we present Hybrid A* (HA*) accompanied with node cutting technique and path smoother to address the challenge of efficiency and path quality. Also, the algorithm is implemented into the USV control system model to demonstrate its feasibility in real application. The basic scheme of the method is presented in Fig. 1.

The reminder of this paper is arranged as: Section II introduces the idea of Hybrid A* and the path smoother technique. Section III presents the simulation results of the efficiency test and its application in the USV control system. The conclusion is addressed in Section IV.

II. Problem Statement

Formally, the problem being addressed in this paper can be stated as follows: Given a start and goal position of $\{S, G\} \subset \Omega$, plan a path $P \subset \Omega$ denoting the set of positions from S to G. For any position $p_i(x_i, y_i) \in P$ along the path, low computational cost and smooth maneuvering are considered to achieve a high-quality solution for USV navigation. In the end, report P or $\emptyset$ if such a path has been found or not

III. Methodology

A. Hybrid A*

The HA* is first presented by [12] for generating paths in unknown environment, where obstacles are detected by sensors. In this paper, the searching strategy of HA* has been adapted. The algorithm uses a modified version of the well-known A* applied to the three-state kinematic model of the vessel, but with an improved strategy that generates continuous states in the discrete search space, see Fig. 2. In traditional A*, the moving space is simple, just including up, down, left, and right movements. Moreover, the conventional method only allows the agent to visit the center of the grids which results in a non-optimal solution. However, unlike the discrete algorithms, HA* supposes that the search agent can reach any continuous points in the cell and not necessarily the vertex. Hence, the transformation from one grid to another in continuous space presents the resilience and possibility to take the vessel dynamics into the algorithm.

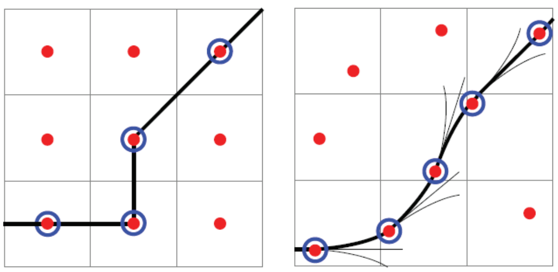

Fig. 2. Searching process of Traditional A* (left); Searching process of Hybrid A* (right) [12]

Heuristic strategy: HA* is guided by two heuristic strategies, the first one ignores obstacles but considers the kinematics constraint of the vessel. To calculate it, assume the goal state is (x, y, ψ) and calculate the shortest path to the goal state for every point (x, y, ψ) in the neighborhood of the goal point regardless of the presence of obstacles. Then the without-obstacle cost and Euclidean distance are used as the heuristic function. The advantage of this strategy is that it cuts the search branches which approach the destination in the wrong directions. The second heuristic strategy is a dual of the first one, which uses the map including the obstacles to calculate the shortest distance by executing dynamic programming in two-dimensional space. The main contribution of the second strategy is that it discovers the U-shaped obstacles and dead-ends and prevents the agent from falling into these areas. Both heuristics are mathematically acceptable in the algorithm, and the final heuristics will be the maximum one of the two values. It is proven to be effectively reducing the number of expanded nodes compared to the Euclidean cost [12].

Node Expansion: In conventional A*, the search space is discretized which means it will never reach the exact continuous goal state (the accuracy depends on the number of grids). However, HA* utilizes a continuous action space to search for new state to address the challenge. The expansion strategy is based on the Reed-Shepp model, with which an expansion tree is generated by simulating the kinematic model of the vessel. It is worth mentioning that it's not efficient to expand with Reed-Shepp for every node. Therefore, to reduce the time cost, the expansion only occurs to one of every n nodes, and n decreases as the searching process moves forward, which leads to more frequent expansions at the final stage of the algorithm. The idea of combining the Reed-Shepp model and continuous action space guarantees that the path generated by HA* is suitable for ship navigation.

B. Waypoint Modifying

The path generated by HA* still contains numerous redundant changes in yaw angle which may cause large control inputs, further smoothing is then required. Therefore, we use the waypoint modifying method (presented in [1]) to improve the path quality for node cutting. The method is constructed in two parts:

- Remove the adjacent points and connect with the latter one if the route does not cross the obstacles.
- Remove the unnecessary turnings and connect with the latter one if the route does not cross the obstacles.

C. B-spline Path Smoother

B-spline is a generalization of the Bezier curve. For m real values x_i , $x_0 \leq x_1 \leq \cdots \leq x_{m-1}$. The B-spline parametric curve of degree n, is composed of a linear combination of basis B-splines $b_{i,n}$ of degree n:

$$S(x): \sum_{i}^{m-n-2} P_i B_{i,n}(x), x \in [x_n, x_{m-n-1}] \tag{1}$$

Where P_i denotes the control points, and $(m-n-1)$ control points form a convex curve. The B-spline of degree n is defined as below with $j = 0, 1, \dots, m-2$:

$$B_{j,n}(x) = \frac{x - x_j}{x_{j+n} - x_j} B_{j,n-1}(x) + \frac{x_{j+n+1} - x}{x_{j+n+1} - x_{j+1}} B_{j+1,n-1}(x) \tag{2}$$

IV. SIMULATION

In this section, simulations have been carried out to evaluate the performance of our proposed method. We have selected some other state-of-the-art methods from existing reliable references for comparison. The simulations are conducted via MATLAB environment with a PC configured with Intel (R) Core (TM) i7-8700 CPU and 8-GB RAM. Furthermore, to obtain reliable results, it should be noted that 50 runs are conducted for each case.

The selected environment map is presented in Fig. 3, and the start and goal points are marked as blue and red dots respectively. A comparative study is conducted between standard Hybrid A* [12] and D* lite [13].

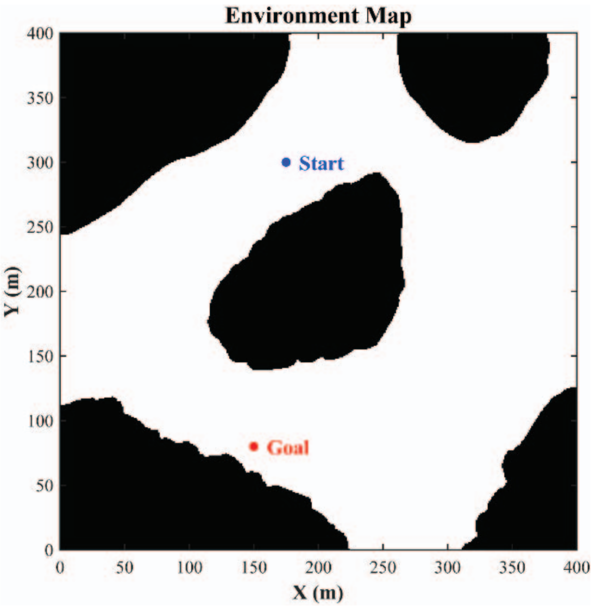

Fig. 3. Environment map (assume 1 pixel = 1 m, Start = (175, 300), End = (150, 80))

TABLE I. QUANTITATIVE RESULTS

Parameter	*Proposed*	*HA**	*D* lite*
Average time cost (s)	0.371	0.609	8.115
Time Standard Deviation (s)	0.018	0.030	0.026
Minimum Path length (m)	244	248.2	258.8
Smoothness value	0.001	1.313	1.262

Quantitative results have been presented in Table I. As can be seen from the data, the proposed algorithm yields the least time cost with 0.371 s for the simulation. The calculation speed increases by more than 60% compared with the traditional Hybrid A*. Furthermore, the robustness of the algorithm is validated by the standard deviation (SD) of the time cost, which is calculated by (3). The proposed algorithm presents the least SD value of 0.018 s for the case. As for the path length, it is shown in the data that the proposed method obtains the least value in terms of minimum length, which is 244 pixels. It is also can be seen from the table that the path given by [12] and [13] is relatively dangerous because they locate too close to the obstacles, mostly because they are optimal search based on A* which aims at the shortest path. The smoothness value of the path is calculated by (4). It is worth mentioning that the smaller the value presented, the smoother the path is. As for path smoothness, the proposed method also presents the smoothest paths (see Fig. 4) with the value of 0.001, which yields significantly better results than the others (i.e. 1000 times smaller than the other two algorithms).

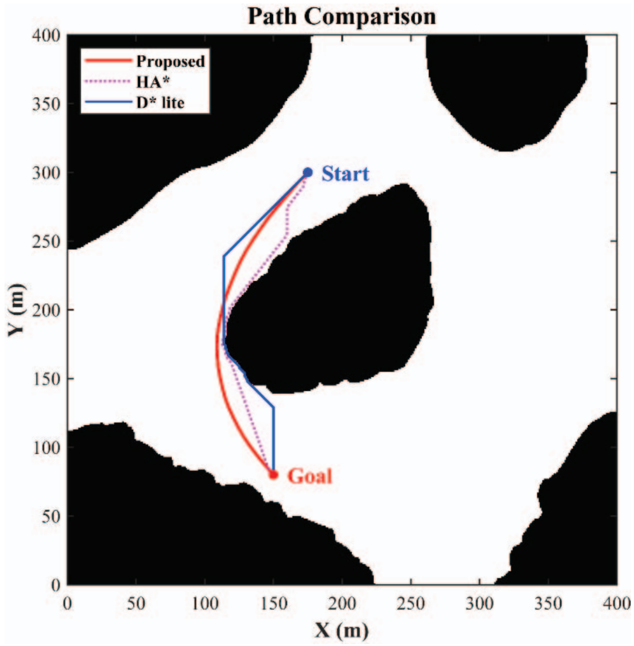

Fig. 4. Path comparison

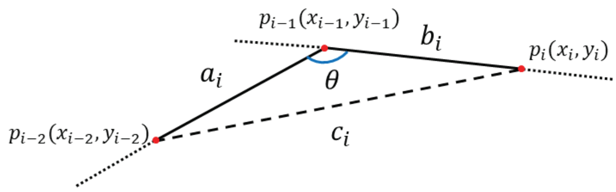

Fig. 5. Path segments

$$SD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f_i - f_{avg})} \tag{3}$$

$$S = \sum_{i=2}^{n-1}(\frac{5(\pi-\theta)}{a_i + b_i})^2 \tag{4}$$

$$\theta = \arccos\,(\frac{a_i^2 + b_i^2 - c_i^2}{2a_ib_i}) \tag{5}$$

where

$$a_i = \sqrt{(x_{i-1}-x_{i-2})^2 + (y_{i-1}-y_{i-2})^2} \tag{6}$$

$$b_i = \sqrt{(x_{i-1}-x_i)^2 + (y_{i-1}-y_i)^2} \tag{7}$$

$$c_i = \sqrt{(x_{i-2}-x_i)^2 + (y_{i-2}-y_i)^2} \tag{8}$$

They represent the edge length in the triangle formed by consecutive path segments. θ is the angle at the location of middle segment in the triangle, see Fig. 5.

To illustrate the performance of the proposed algorithm in practical application, we use a model ship in MSS toolbox [14] from a real USV prototype, see Fig. 6. All the three paths are selected for the simulation.

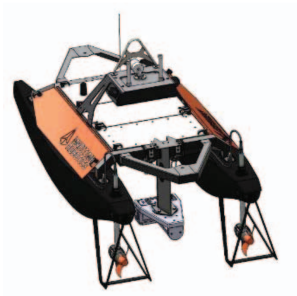

Fig. 6. USV prototype

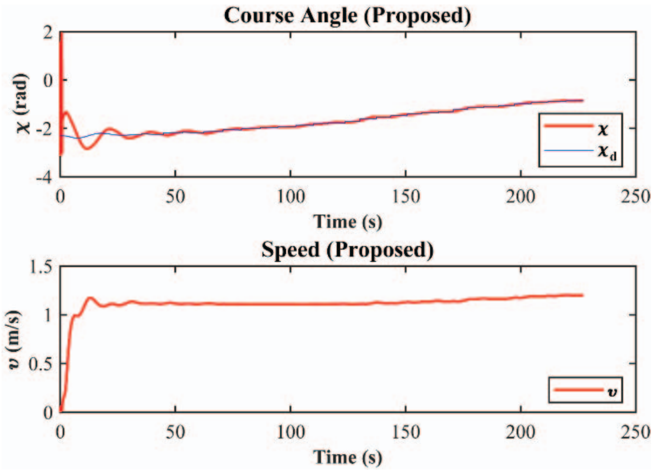

Fig. 7. Course angle and speed changes of USV during the simulation (proposed algorithm)

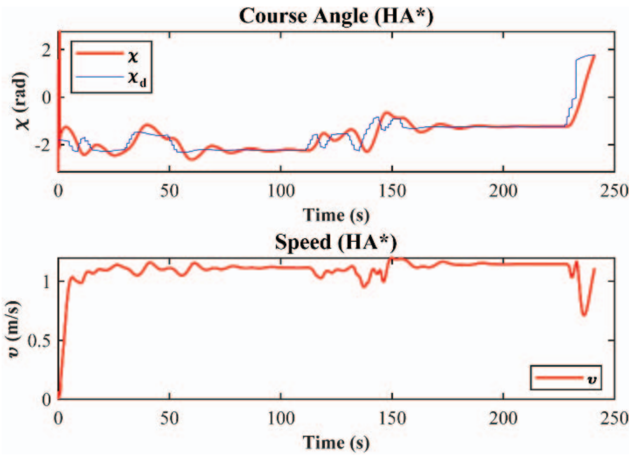

Fig. 8. Course angle and speed changes of USV during the simulation ([12])

Simulation results in the USV model are presented in Fig. 7-9. As can be seen from Fig. 7, the path given by the proposed method shows mild and smooth changes in terms of course angle and speed. The deviation between the actual and desired signal is small, which demonstrates its feasibility of cooperating with the USV control system. However, the path given by [12] and [13] presents abrupt changes in course angle and velocity, see Fig. 8 and Fig. 9. Large deviation can be observed at the location where the course suddenly changes, which will lead to large control inputs.

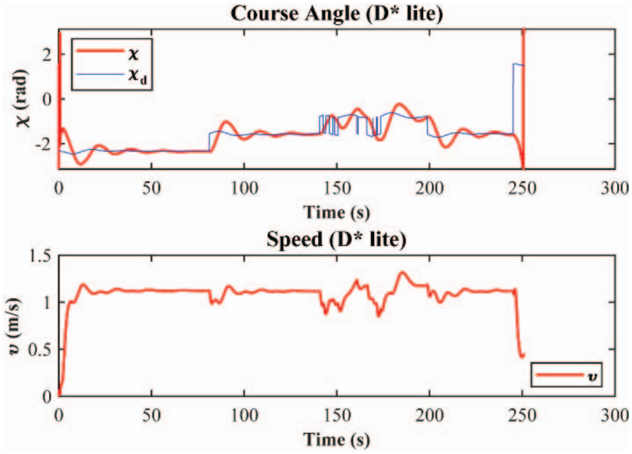

Fig. 9. Course angle and speed changes of USV during the simulation ([13])

V. Conclusion

In this article, we prosed a modified Hybrid A* algorithm with node-cutting technique and path smoother to solve local path planning problem. The efficiency and path quality test verified that it is better than the several other algorithms from existing reliable references in terms of effectiveness and smoothness. Moreover, the simulation conducted in the USV model demonstrates that the proposed method can perfectly coordinate with the control system. Therefore, the simulation results have presented strong evidence that the proposed algorithm is reliable for practical application.

Acknowledgment

I would like to express my gratitude to all those who have helped improving the quality of the paper. This paper is mainly supported by the College of Civil Engineering and Architecture at Zhejiang University.

References

[1] L. Zhao, F. Wang, and Y. Bai, “Route planning for autonomous vessels based on improved artificial fish swarm algorithm,” Ships and Offshore Structures, Jun. 2022.

[2] R. K. Katzschmann, J. DelPreto, R. MacCurdy, and D. Rus, “Exploration of underwater life with an acoustically controlled soft robotic fish,” *Science Robotics*, vol. 3, no. 16, p. eaar3449, 2018.

[3] T. Pastore and V. Djapic, “Improving autonomy and control of autonomous surface vehicles in port protection and mine countermeasure scenarios,” *Journal of Field Robotics*, vol. 27, no. 6, pp. 903–914, 2010.

[4] G. A. Wilde and R. R. Murphy, "User interface for unmanned surface vehicles used to rescue drowning victims," 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), 2018, pp. 1-8.

[5] C. Ntakolia and D. V. Lyridis, “A comparative study on Ant Colony Optimization algorithm approaches for solving multi-objective path planning problems in case of unmanned surface vehicles,” *Ocean Engineering*, vol. 255, p. 111418, Jul. 2022.

[6] D. Ma, S. Hao, W. Ma, H. Zheng, and X. Xu, “An optimal control-based path planning method for unmanned surface vehicles in complex environments,” *Ocean Engineering*, vol. 245, p. 110532, Feb. 2022.

[7] H. Sang, Y. You, X. Sun, Y. Zhou, and F. Liu, “The hybrid path planning algorithm based on improved A* and artificial potential field for unmanned surface vehicle formations,” *Ocean Engineering*, vol. 223, p. 108709, Mar. 2021.

[8] Y. Ma, Y. Zhao, Z. Li, X. Yan, H. Bi, and G. Królczyk, “A new coverage path planning algorithm for unmanned surface mapping vehicle based on A-star based searching,” *Applied Ocean Research*, vol. 123, p. 103163, Jun. 2022.

[9] L. Zhao, Y. Bai, F. Wang, and J. Bai, "Path planning for autonomous surface vessels based on improved artificial fish swarm algorithm: a further study," *Ships and Offshore Strucures*, Sept. 2022.

[10] A. Vagale, R. Oucheikh, R. Bye, O. Osen, and T. Fossen, "Path planning and collision avoidance for autonomous surface vehicles I: a review," *Journal of Marine Science and Technology*, vol. 26, Jan. 2021.

[11] A. Vagale, R. Bye, R. Oucheikh, O. Osen, and T. Fossen, "Path planning and collision avoidance for autonomous surface vehicles II: a comparative study of algorithms," *Journal of Marine Science and Technology*, Feb. 2021.

[12] D. Dolgov, S. Thrun, M. Montemerlo, and J. Diebel, "Practical search techniques in path planning for autonomous driving," *AAAI Workshop - Technical Report*, Jan. 2008.

[13] Y. Yao *et al.*, "Path Planning Method Based on D* lite Algorithm for Unmanned Surface Vehicles in Complex Environments," *China Ocean Engineering*, vol. 35, no. 3, pp. 372–383, Jul. 2021.

[14] T. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. 2011.