

Route Planning for Autonomous Vessels Based on Improved Artificial Fish Swarm Algorithm

Liang Zhao^a, Yong Bai^{a*}

^aSchool of Civil Engineering and Architecture, Zhejiang University, Hangzhou 310058, China

Abstract

Path planning is one of the key technologies in the research of autonomous surface vessels (ASVs). In this paper, an improved artificial fish swarm algorithm (IAFSA) has been proposed. The algorithm is modified from four perspectives: (1) A directional operator is introduced to improve the efficiency of the prey behavior; (2) To avoid local optimum, a weight factor is proposed to adjust the probability of executing random behavior; (3) An adaptive operator has been applied to the algorithm aims at better convergence performance; (4) The waypoint modifying path smoother is used to improve the path quality. A comparative study has been carried out between IAFSA and the other state-of-the-art algorithms, the results shows that the proposed algorithm outperforms the others in both efficiency and path quality. The route given by IAFSA is also less time-cost and more suitable for navigation. Finally, IAFSA is integrated into the GNC system in a model ship. A computer-based sea trial around Nan Hai area has been conducted and environment disturbances including wind, waves, and currents are considered. The result has shown that IAFSA has great feasibility in practical application.

Keywords: ASV; Path Planning; GNC system; AFSA

Contact: Yong Bai

Email: baiyong@zju.edu.cn

1. Introduction

With the development of the autonomy technology, there has been a growing appeal for the research of autonomous systems. In particular, much attention is given to autonomous surface vessels (ASVs) due to their potential benefits of improving safety and efficiency. ASVs can be defined as unmanned ships which perform tasks in a variety of cluttered environments without any human intervention, and essentially exhibit highly nonlinear dynamics. ASVs have already showed their contributions in some areas like ocean research, ocean resource exploration, military, maritime transport, and rescue ([Zhou et al. 2020](#)). [Table 1](#) shows the examples of ASVs' applications in the recent years.

Path planning is one of the key technologies in the process of automating the ASVs and carrying out complex tasks of ASVs ([Fossen 2011](#); [Lazarowska 2015](#)). In the recent years, there is an increasing demand for maritime safe navigation and path planning technology. According to the Annual Overview of Marine Casualties and Incidents 2019 ([EMSA 2019](#)) collected by the European Maritime Safety Agency (EMSA), during 2011-2018 more than 54% of all accidents with ships were navigational accidents. Also, 65.8% of these accidents were attributed to human actions. The high percentage of navigational casualties and human erroneous actions can be largely reduced by introducing path planning algorithms.

Developing path planning algorithms with great computing efficiency, robustness, and higher-quality

solutions is an attractive topic in the recent studies. Sang et al. (2021) proposed a multiple sub-target artificial potential field (MTAPF) based on traditional APF. The MTAPF can greatly reduce the probability of USVs falling into the local minimum by switching target points. Xie et al. (2019) modified A* algorithm by combining scalar mode, adaptive step and penalty mode. Comparing with the real-case trajectory, the distance to the obstacles increased more than 3 times and path length is much shorter. Liang et al. (2020) proposed a leader-vertex ant colony optimization algorithm (LVACO) and applied it to ASV control system. Simulation results reveal that the route given by LVACO algorithm is more efficiency and suitable for ship navigation. Another attempt was made by Zhong et al. (2021), they combined particle swarm optimization with orientation angle-based grouping. The modified algorithm showed better performance in converging time and path length. To implementing algorithm to ship autonomous navigation, Lazarowska (2020) used a discrete artificial potential field method combining with COLREGs. The method was validated by using real navigation data from training ship *Horyzont II*. Guo et al. (2020) proposed a chaotic and sharing-learning particle swarm optimization (CSPSO) algorithm to solve the multi-objective path planning problem. Simulation experiments validate that CSPSO algorithm and collision avoidance rules are effective and justifiable.

Furthermore, there is an increasing interest in considering dynamic obstacles and multiple ASVs path planning. To cope with dynamic obstacles, Lyridis (2021) improved ant colony optimization (ACO) algorithm with fuzzy logic method. The path planner considered multi-objectives including traveled distance, the path smoothness and the fuel consumption. By using a multilayer path planner, Wang et al. (2019) solved the ASV path planning problem in complex marine environments including dynamic obstacles and arisen reefs. The B-spline method and stochastic dynamic coastal environment (SDCE) model is built to adapt the time-varying environments. A new path planning algorithm based on self-organizing map (SOM) and fast marching method (FMM) has been proposed by Liu et al. (2019). The algorithm has been verified through a number of multiple USVs simulations and has been proven to work effectively in both simulated and practical maritime environments. To solve online relative optimal path planning problem of multiple USVs, Wen et al. (2020) improved rapidly extending random tree (RRT*) for path optimization. The dynamic obstacle avoidance has been investigated based on COLREGs and the algorithm was proved to be relatively easier to execute and lower in fuel expenditure than traditional schemes.

Artificial Fish Swarm Algorithm (AFSA) is a new bionic algorithm imitating the social behavior of fish in nature, which was firstly brought forward by Li et al. (2002). The algorithm has been used to solve mostly robot or ground vehicle path planning while there are very limited researches introduce its application in autonomous vessels. Zhang et al. (2016) adopted inertia factor to AFSA to improve the convergence rate and accuracy. The experiments were carried out on Pioneer 3-DX mobile robot based on robot operation system (ROS). To ensure the unmanned ground vehicle's safety in an uncertain environment, Zhou et al. (2021) combined trial-based forward search (TFS) with ASFA. The experiment results have shown the storage efficiency and convergence rate are sufficiently enhanced comparing to dynamic programming.

Through the literature review we have observed a number of limitations in recent studies for path planning. They are summarized as follows:

- Many algorithms that appear to be efficiency theoretically are not applied and tested in ship control system, which means they are not convincing in handling real-world situation (Vagale et al. 2021).
- In many cases, the developed algorithms do not consider external disturbances such as wind, waves, or current, see Table 2. The modelled environment is not complete and the performance of the

algorithms under real conditions would differ ([Huang et al. 2020](#); [Vagale et al. 2021](#)).

Therefore, to cope with these limitations, an improved waypoint based IAFSA has been developed and validated. The contributions of this work can be addressed in the following three aspects:

- An improved artificial fish swarm algorithm (IAFSA) is proposed, which combines a directional operator, probability weight factor, adaptive operator, and a path smoother to improve the efficiency and feasibility of the algorithm.
- A competitive study between IAFSA and other state-of-the-art algorithms is provided to illustrate the excellent performance of the algorithm.
- Simulation-based sea trials around Nan Hai area by integrating IAFSA with ship GNC system is carried out to prove its feasibility in real-world situation. Ocean disturbances including wind, waves and current are considered during the trials.

The paper is organized as follows: The methodology of improve artificial fish swarm algorithm and a competitive study of the algorithms are presented in Section 2. In Section 3, the IAFSA is integrated into the GNC system of a model ship and a computer-based sea trial has been conducted. Conclusion is addressed in Section 4.

Table 1. Examples of ASVs' applications

Field	Applications
Scientific Research	Marine biology studies (Katzschmann et al. 2018); Bathymetric survey (Sato et al. 2015); water quality monitoring (Yang et al. 2018).
Ocean Resource Exploration	Oil, gas and mine explorations (Pastore and Djapic 2010 ; Roberts and Sutton 2006)
Military	port, harbor, and coastal surveillance (Zhao 2018); anti-submarine (Fahey and Luqi 2016)
Other Applications	transportations and maritime rescue (Shafer et al. 2008 ; Wilde and Murphy 2018)

Table 2. Lists of Recent Studies

Research	Method	Incorporating with GNC system	Considering wind, waves and currents
(Sang et al. 2021)	MTAPF	No	No
(Xie et al. 2019)	Improved multi-direction A* algorithm	No	No
(Liang et al. 2020)	LVACO	Yes	Yes
(Zhong et al. 2021)	Improved PSO	Yes	Yes
(Lazarowska 2020)	Discrete APF	No	No
(Guo et al. 2020)	CSPSO	Yes	Yes
(Lyridis 2021)	Improved ACO	No	No
(Wang et al. 2019)	Improved FMM	No	No
(Liu et al. 2019)	FMM combining SOM	No	No
(Wen et al. 2020)	Improved RRT*	No	No

2. Methodology

2.1. Traditional AFSA for Route Planning

Artificial fish (AF) is a fictitious entity of fish in real world. The AF imitates the social behaviors of real fish to find the position with the best food consistency. The algorithm consists four behaviors: prey

behavior, follow behavior, swarm behavior and random behavior. The basic idea of AFSA can be described as follows:

The total population of the fish is N , the position state of each AF is $X = (X_1, X_2, X_3, \dots, X_n)$. $Y = f(X_i)$ denotes the food concentration in the current position. *Visual* and *Step* is the maximum range in which AF can search and move.

(1) Prey behavior: In nature, prey behavior is a basic biological behavior for fish to find food, see [Figure 1](#). The current state of an AF is $X_i(t)$. $X_j(t)$ is the random position state neighboring the current AF in the range of visual, which can be expressed as (1).

$$X_j(t) = X_i(t) + Visual \times rand(0,1) \quad (1)$$

If the food concentration $Y_i > Y_j$, the AF will take a *Step* in the direction of $X_j(t)$, which is done by

$$X_i(t+1) = X_i(t) + \frac{X_j(t) - X_i(t)}{\|X_j(t) - X_i(t)\|} \times Step \times rand(0,1) \quad (2)$$

If $Y_i < Y_j$, then $X_j(t)$ will be selected again. if the AF is not satisfied with the forward condition after *try_num* times, the concerned AF performs random behavior to avoid the local optimum.

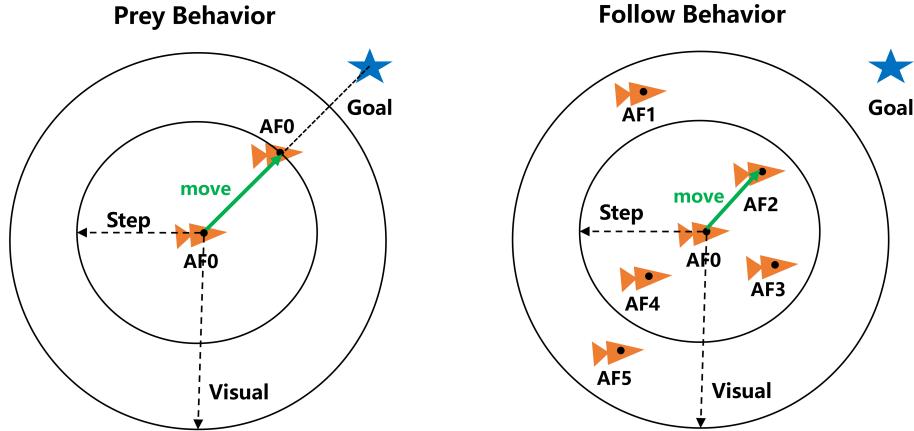


Figure 1. Illustration of prey behavior (left) and follow behavior (right)

(2) Follow behavior: During the follow behavior, once an AF find better food consistency, the neighbor AFs will follow and reach it immediately, see [Figure 1](#). Suppose the current position of AF is $X_i(t)$ and position $X_j(t)$ denotes the neighbor fish in its visual scope. And n_f is the number of AF swarms in the visual scope of the swarm center defined as $X_c(t)$, see equation (4). δ is the parameter denoting the level of crowds. If $Y_j < Y_i$ and $n_f/N < \delta$, which means position $X_j(t)$ has better food consistency and is not crowded. Then the AF moves a step to the direction of $X_j(t)$. The expression can be determined as (3)

$$X_i(t+1) = X_i(t) + \frac{X_j(t) - X_i(t)}{\|X_j(t) - X_i(t)\|} \times Step \times rand(0,1) \quad (3)$$

if there are no neighbor to be found or the condition is not satisfied, the AF will perform prey behavior.

(3) Swarm behavior: In order to keep swarm generality, AFs attempt to move towards the central position in every time of iterations, the follow behavior is illustrated by [Figure 2](#). The central position is determined in the following equation:

$$X_c(t) = \frac{1}{N} \sum_{i=1}^N X_i(t) \quad (4)$$

Where $X_c(t)$ is the arithmetic average of all AFs' state. Denote n_f as the number of AF swarms in the visual range of $X_c(t)$. If $n_f/N < \delta$ and $Y_c < Y_i$, it means the center position has better food

consistency and the swarm is not crowded, then the AF can take a step in the direction of $X_c(t)$, which is done by

$$X_i(t+1) = X_i(t) + \frac{X_c(t) - X_i(t)}{\|X_c(t) - X_i(t)\|} \times Step \times rand(0,1) \quad (5)$$

Otherwise, the AF executes prey behavior.

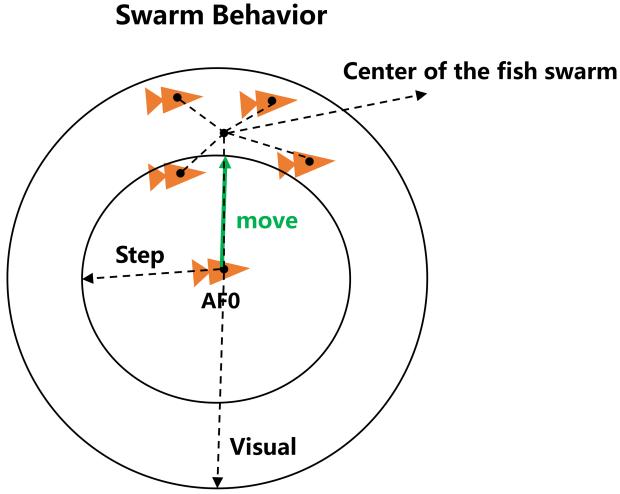


Figure 2. Illustration of swarm behavior

- (4) **Random behavior:** To prevent local optimum, the AF would execute random behavior at a specific frequency to seek food in large ranges. Random behavior means the AF chooses an arbitrary state or position randomly in its visual field, and then it swims towards the selected state. It can be described as:

$$X_i(t+1) = X_i(t) + Step \times rand(0,1) \quad (6)$$

2.2. The Improved Artificial Fish Swarm Algorithm

(1) Directional Operator and Probability Weight Factor

In the traditional AFSA, before moving to the next position, AF usually searches for the next state randomly in the range of its visual and then determines whether the food concentration is better than the current position. There are some drawbacks in this process. First, it is a time-consuming task. There is uncertainty in selecting the next position. AF needs to try all the possible directions until the position is found. Second, the path generated by the prey behavior is not optimal. Due to the random process in prey behavior, the path quality is greatly influenced by the jag effect.

To overcome the problems mentioned above, we introduce a heuristic directional operator. During the prey behavior, the AF is inspired by the operator and consciously choose the best position by itself. Suppose that $X_{i,j}^p$ represents the potential position number j of AF number i in the step range. P represents the set of all the potential position, it can be expressed as

$$P = \{X_{i,j}^p | X_{i,j}^p - X_i \leq Step, i = 1, 2, 3, \dots, N, j = 1, 2, \dots, M\} \quad (7)$$

In prey behavior, the food concentration of all the potential positions in P is calculated and then the best position $X_{i,best}^p$ is selected to be the next step, see equation (8) and (9). The directional operator firstly guarantee the AF can find the best position with one cycle of calculation, which significantly reduce the computing time comparing with traditional method. Secondly, the directional operator replaces the random process thereby the redundant turns and unnecessary nodes are reduced.

$$X_{i,best}^p = \min\{f(X_{i,1}^p), f(X_{i,2}^p), f(X_{i,3}^p), \dots, f(X_{i,M}^p)\} \quad (8)$$

$$X_i(t+1) = X_i(t) + \frac{X_{i,best}^p(t) - X_i(t)}{\|X_{i,best}^p(t) - X_i(t)\|} \times Step \times rand(0,1) \quad (9)$$

However, after eliminating the random process, the adaptability of the AFSA is weakened. The algorithm is easier getting into local optimum. In order to compensate for it, we introduce a probability weight factor m , which follows the Bernoulli distribution. Therefore, the AF has a certain probability make random jumps to avoid the algorithm falling into the local optimum. Equation (9) is rewritten as (10).

$$X_i(t+1) = \begin{cases} X_i(t) + \frac{X_{i,best}^p(t) - X_i(t)}{\|X_{i,best}^p(t) - X_i(t)\|} \times Step \times rand(0,1), & m = 0 \\ X_i(t) + \frac{X_j(t) - X_i(t)}{\|X_j(t) - X_i(t)\|} \times Step \times rand(0,1), & m = 1 \end{cases} \quad (10)$$

(2) Adaptive Factor

In standard AFSA, the visual and step are fixed. At the beginning of the algorithm, the large values of visual and step lead the AFs to quickly move to the target position. However, when AFs move close to the final position, the large values will cause problems such as local optimum or iterative jumps. Also, if the values are too small, the global search ability is decreased which more time cost. Therefore, the algorithm has different requirements on the size of visual and step in different optimization periods. To balance the global search ability and convergence rate, an adaptive factor is introduced.

Suppose N is the total number of AFs, each AF is recorded as $F_i (i = 1, 2, 3, \dots, N)$. And D_{ij} denotes the distance between two AFs F_i and F_j . The weight factor of the distance D_{ij} is ω_{ij} . Therefore, the visual can be calculated by the weighted average method:

$$V_i = \frac{\sum_{j=1; j \neq i}^N D_{ij} \omega_{ij}}{\sum_{j=1; j \neq i}^N \omega_{ij}} \quad (11)$$

At the early stage of the algorithm, the large visual value will lead AFs move towards positions with high food concentration and make them gathering. The distance between each of the individuals become smaller. Therefore, according to (11) the visual is gradually getting smaller as the algorithm proceeding, which will improve the efficiency of searching.

Step is another vital parameter which determines the accuracy of the algorithm. Large step means low accuracy, conversely, small step will improve the accuracy but lead to more time cost. To balance the accuracy and iteration speed, an adaptive step is introduced as (12).

$$Step = Step \times (1 - \frac{i}{MaxIter}) \quad (12)$$

i denotes the current iteration and $MaxIter$ is the total iterations. From (12), in the initial stage the step is relatively large and convergence speed is fast, the space is easily to be searched. As the iteration process, the step gradually gets smaller which leads to accurate solution.

(3) Waypoint Modifying Path Smoother

The result of traditional method usually contains jags and unnecessary waypoints. Closely placed waypoints and turns will generate extra control demand, which could lead to degraded tracking performance. The path dose not conform to the ASV dynamics in reality. Therefore, further smoothing method is required. To overcome this problem, waypoint modifying path smoother (WMPS) has been used.

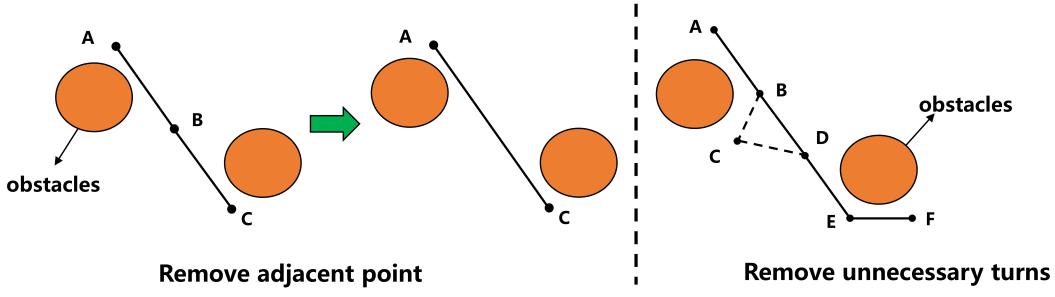


Figure 3. Illustration of waypoint modifying path smoother

The method is divided into two steps. (1) Remove the adjacent points if they are colinear. (2) Remove the unnecessary turns if the path does not pass the obstacles. For each iteration, the WMPS checks all waypoints in consecutive three groups: the current waypoint denoted as A; the next via waypoint of A denoted as B; B's via waypoint denoted as C. [Figure 3](#) illustrates how it works. The number of waypoints is reduced and jags are being avoided after modifying. The path generated by WMPS can provide optimum number of waypoints to achieve better control performance for ASV.

The pseudocode of IAFSA is shown in [Algorithm 1](#).

Algorithm 1: Pseudocode of IAFSA

Input: N : the population of the artificial fish
 X_i : the position state of AF i
 $Visual$: the maximum range in which AF can search
 $Step$: the maximum range AF can move in each iteration

```

1 initialize the artificial fish population:  $N$ 
2 initialize the state of each AF:  $X_i$ 
3 while the path is not found do
4   foreach  $AF \in N$  do
5     flag=state evalutation();
6     case flag=prey ;
7        $X_i(t+1) \leftarrow X_i(t) + step \text{ towards best potential position}, m = 1$ 
8        $X_i(t+1) \leftarrow X_i(t) + step \times rand(0, 1), m = 0$ 
9     case flag=follow ;
10       $X_i(t+1) \leftarrow X_i(t) + step \text{ towards neighbor fish with better position}$ 
11    case flag=swarm ;
12       $X_i(t+1) \leftarrow X_i(t) + step \text{ towards the center of fish swarm}$ 
13    case flag=random ;
14       $X_i(t+1) \leftarrow X_i(t) + step \text{ towards random direction}$ 
15  end
16  Update the state of each AF  $X_i$ ;
17  Update  $Visual$  and  $Step$ ;
18 end
19 route=AFselection( $X_i$ );
20 Waypoints=PathSmoothen(route);
Output: Waypoints
```

2.3. Simulation Results

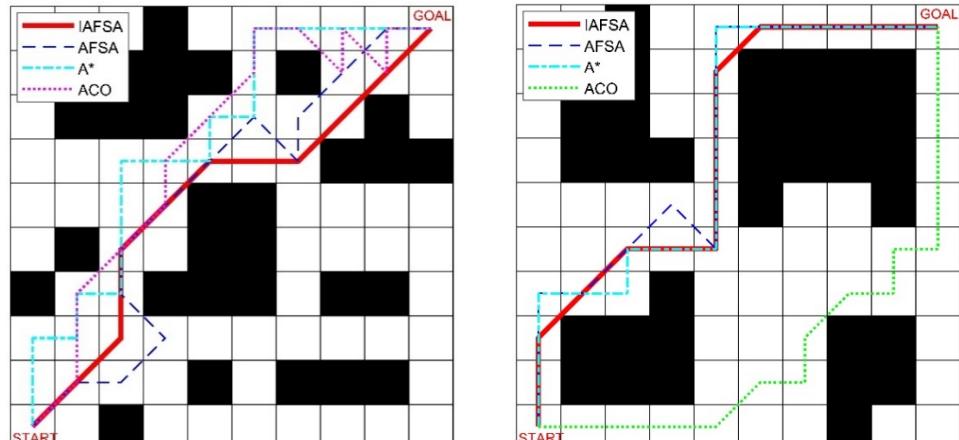
This section presents the results of the computer simulations. To verify the effectiveness of IAFSA, four scenarios of grid maps are used and several other algorithms are compared in the simulations. Both starting parameters of IAFSA and AFSA are set to be the same, therefore ensuring the same initial condition. The parameters of ACO are set according to liang et al. (2020). Also, the simulations are conducted via MATLAB environment with a PC configured with Intel(R) Core(TM) i7-8700 CPU and 8-GB RAM. Nevertheless, to eliminate the randomness of the algorithms, it should be noted that 100 runs

are conducted to obtain a dataset of each scenario.

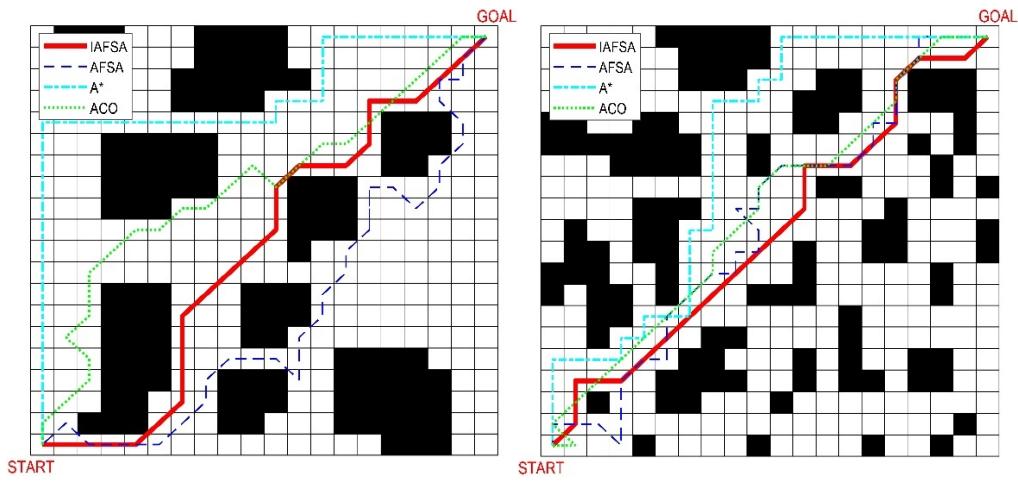
[Figure 4](#) demonstrates the generated paths by the four algorithms in 10*10 and 20*20 grid maps, respectively. Quantitative results are presented in [Table 3](#), the average value of path length (pixels), the best value of path length (pixels), standard deviation (SD) and average time consuming (s) are included. It is worth mentioning that the time cost and path length are key factors to reflect the effectiveness. Furthermore, the SD, which is defined by equation (13), reflects the departure degree of dataset from the average and it is a vital index to evaluate the algorithm robustness. The best value of the path length generated by IAFSA are presented in bold. [Figure 5](#) presents the competitive results between different methods.

$$SD = \sqrt{\frac{1}{100} \sum_{i=1}^{100} (L_i - AVG)^2} \quad (13)$$

Overall, the IAFSA have relatively better performance than the others in all the scenarios. From [Figure 4](#), IAFSA achieved better solutions in terms of path quality. The line in blue and green are the paths generated by AFSA and ACO. It can be seen that a number of zig zags are formed, making it hard for practical use. However, the line in red is obtained by IAFSA, it gives the routes with less waypoints and unnecessary turns, which is more suitable for ASV navigation.



(a)



(b)

Figure 4. (a) Scenario 1 and 2 (10×10); (b) Scenario 3 and 4 (20×20)

From [Table 3](#), we observe that IAFSA presented the least value in both average and best path length. The shortest path is obtained with the value of 13.9 pixels, 15.07 pixels, 30.38 pixels and 28.63 pixels, respectively. Also, the superior robustness of IAFSA was shown from the SD values of 0.39 pixels, 0.46 pixels, 0.45 pixels and 0.49 pixels. Meanwhile, the time consuming is significantly reduced due to the directional operator and adaptive operator. In particular, the average time cost is almost half of the other algorithms in scenario 3 and 4.

Table 3 Performance comparison between different methods for four scenarios

Method	Results	Scenario 1	Scenario 2	Scenario 3	Scenario 4
IAFSA	AVG L (pixels)	14.06	16.26	31.56	30.26
	Best (pixels)	13.90	15.07	30.38	28.63
	SD (pixels)	0.39	0.46	0.45	0.49
	AVG T (s)	0.46	0.60	1.59	1.46
AFSA	AVG L (pixels)	16.61	17.86	38.97	37.75
	Best (pixels)	15.07	16.24	35.07	31.80
	SD (pixels)	0.72	1.00	1.62	1.57
	AVG T (s)	0.68	0.73	2.23	2.31
ACO	AVG L (pixels)	16.84	16.60	32.97	32.07
	Best (pixels)	15.07	15.07	30.97	29.21
	SD (pixels)	0.94	0.82	0.95	1.19
	AVG T (s)	0.68	0.65	3.14	3.11
A*	AVG L (pixels)	18	18	38	38
	Best (pixels)	-	-	-	-
	SD (pixels)	-	-	-	-
	AVG T (s)	0.99	1.11	3.83	4.04

[Figure 5](#) shows the competitive results of the algorithms. First, the proposed algorithm achieved better performance in term of computational time for generating a path for avoiding the detected obstacles. [Figure 5 \(a\)](#) shows IAFSA the convergence speed is largely improved, especially in complex environment such as scenario 3 and 4. Regarding the length of the generated path, IAFSA gives the optimal in all cases. In particular, the path length is 10%-20% shorter in each scenario comparing with the conventional AFSA. As for the number of waypoints, the waypoints given by IAFSA are significantly less than other competitive algorithms according to [Figure 5 \(c\)](#). Hence, we can conclude that IAFSA is suitable for ASV path planning.

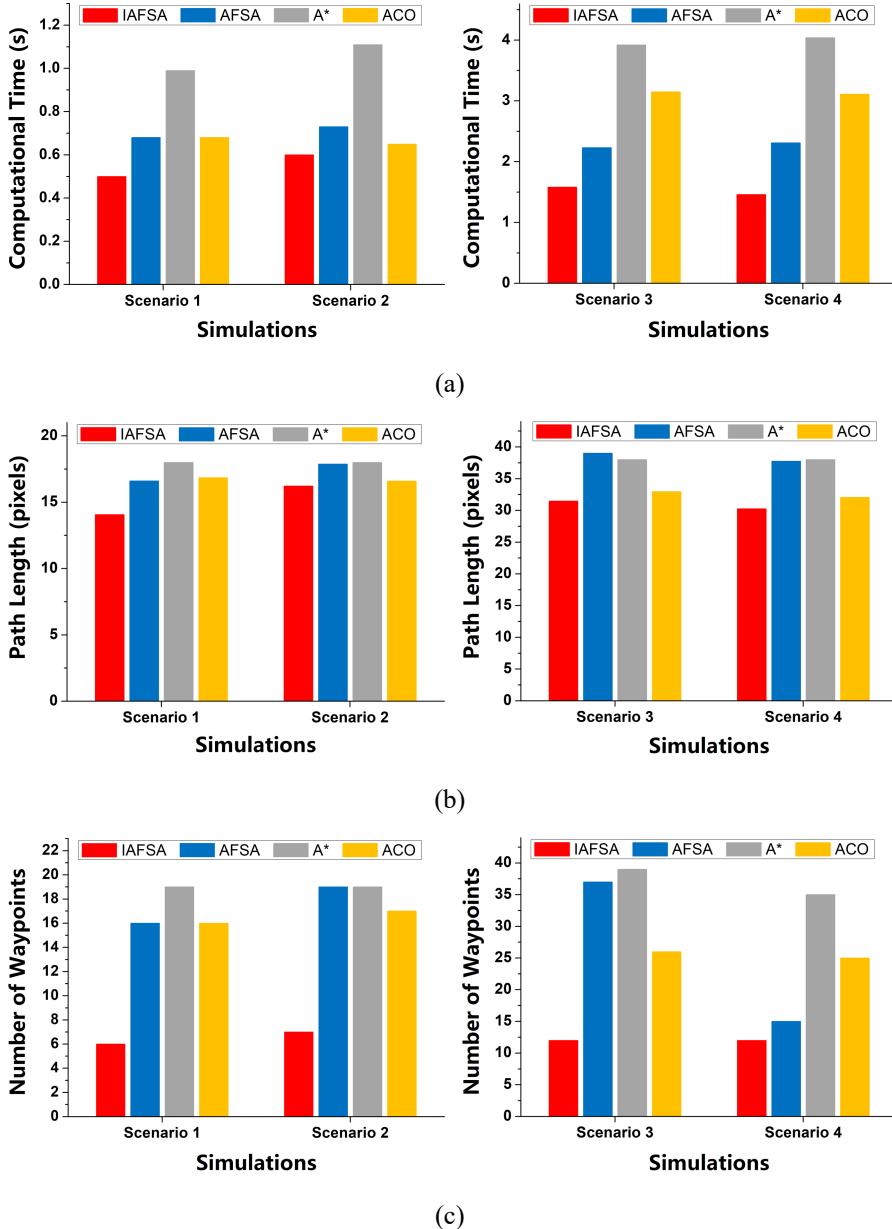


Figure 5. (a) Computational time per scenario per algorithm; (b) Path length per scenario per algorithm; (c) Number of waypoints per scenario per algorithm

3. Computer-based Sea Trial on GNC System

In this section, the IAFSA is applied to the GNC system in a model ship (see Figure 6). The model ship is equipped with a Class 3 DP system to accomplish the mission. It is driven by 8 diesel generators and 12 thrusters. The mathematical model is designed according to Fossen (2011), see equation (14-15).

$$\begin{aligned} \dot{\eta}_p &= v \\ M\dot{v} + Dv &= R^T(\psi)b + \tau + \tau_{en} \\ \dot{b} &= 0 \end{aligned} \tag{14}$$

Where

$$\tau = Bu \tag{15}$$

In this model, η_p denotes the position vector in vessel parallel coordinates. M and D are mass matrix and damping matrix. The bias vector b is the slowly varying ocean currents and $R^T(\psi)$ is the transfer

matrix between NED coordinate system and body-fixed coordinate system. τ_{en} denotes the environment forces. The control matrix B describes the thruster configuration while u is the control input vector. The GNC system of the model ship, with which the improved artificial fish swarm algorithm has been integrated, is designed for simulation-based sea trials.



Figure 6. Model Ship

The configuration of the GNC system in the model ship is illustrated in [Figure 7](#). It contains three basic systems: the guidance system, the navigation system, and the control system. The mission requirement and information of the environment are first sent to the guidance system to generate a map. Then the IAFSA will compute a safe route for the vessel. According to the information and vessel dynamics, the guidance system calculates the reference trajectory and velocity. The reference trajectory is further received by the control system, which includes a PID controller and a thrust allocation system to generates the control commands. The navigation system employs a GPS and gyroscopic compass to measure the position and the velocity information. To improve the accuracy of the signal, an Extended Kalman Filter is designed to refine the signal collected by the sensors.

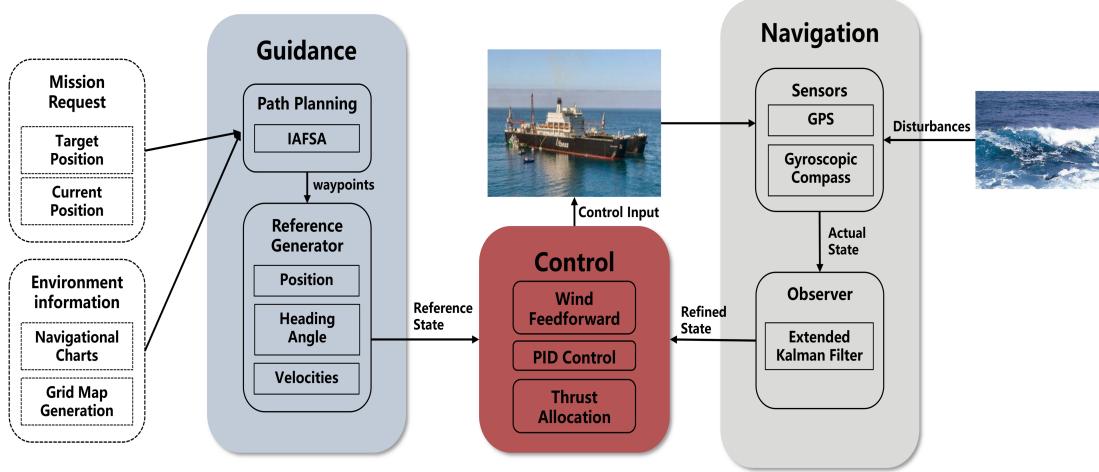


Figure 7. GNC structure of Model Ship

Sea trials have been carried out in a real sea environment around Nan Hai area in southern China, see [Figure 8 \(a\)](#). The start and goal position have been manually selected, see [Figure 8 \(b\)](#). To complete the environment model, it is worth mentioning that the vessel is exposed to environment disturbances including wind, waves, and currents. Also, the vessel dynamics constraints are considered to simulate the real situation. These parameters are shown in [Table 4](#) and [Table 5](#).

Table 4. Parameters of the environment

Parameters	Value
------------	-------

Environment parameters	Wave Hs	1.5m
	Wave Tp	14s
	Wind speed	10m/s
	Current speed	0.1m/s

Table 5. Parameters of vessel dynamics

	Parameters	Value
Vessel dynamics	Max speed	0.5m/s
	Max turn	5 deg/s
	Max power	11000kw

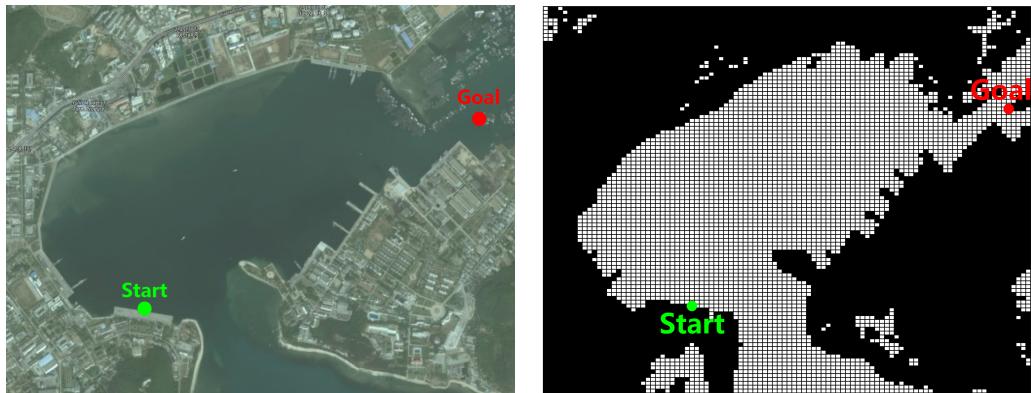


Figure 8. (a) Satellite map of the sea trial site around Nan Hai area; (b) Map after grid process (100*100 grid cells, 1 grid cell column length = 22.3m and row length = 33.5m)

Overall, the GNC system with IAFSA has good performance executing the sea trial. The vessel autonomously follows the route without any human intervention. Figure 9 shows the simulation results presented by IAFSA. The route is smooth with only 5 waypoints totally, which makes the trajectory easier to track. The blue curve is the reference trajectory generated by the guidance system while the red curve is the real trajectory. We can see these two curves are basically overlapped. Furthermore, according to Figure 10, the tracking error is less than 0.2m in both X and Y direction. It means the actual trajectory has small deviations from the planned path owing to the effects of environment disturbances.



Figure 9. Planned route and real trajectory of the vessel

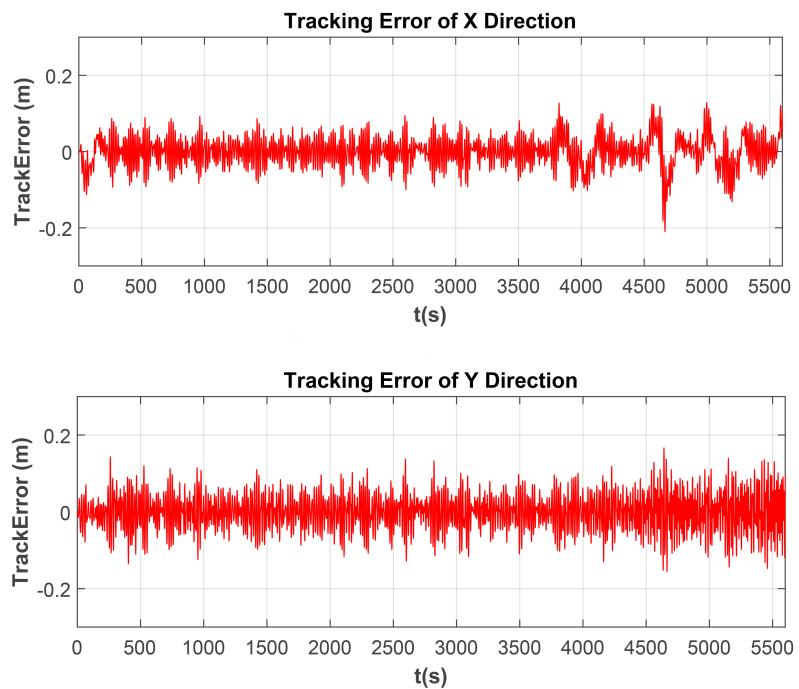
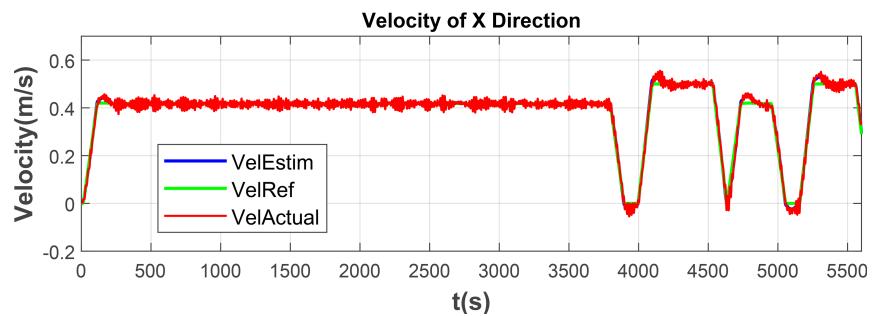


Figure 10. Tracking error of the trajectory



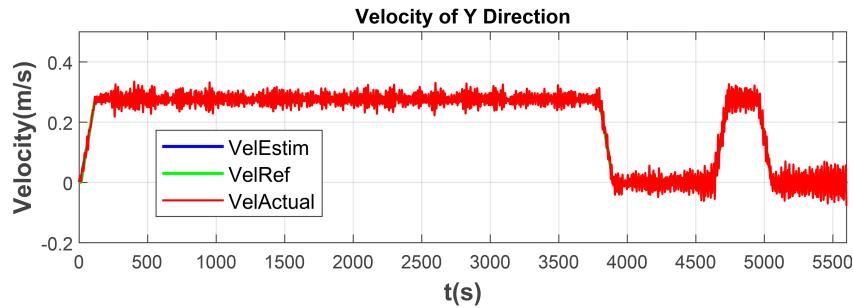


Figure 11. Velocities of the vessel

Figure 11 demonstrates the velocity changes during the trial. The blue curve is the velocity signal refined by the extended Kalman Filter. Green curve is the reference velocity signal generated by guidance system while the red curve is the actual velocity. The curves have shown the excellent performance of the navigation system and control system. It can be observed that a relatively large deviation exists when the vessel changed directions. The reason for this is that the thrust forces changed drastically to turn the direction. Overall, the tracking deviation is small and the vessel dynamics are satisfied during the trial.

4. Conclusion

In this paper, an improved artificial fish swarm algorithm has been proposed, verified and validated by a set of simulations. The algorithm is designed to improve the efficiency and feasibility for path planning of autonomous vessels. Comparing with other state-of-the-art algorithms, IAFSA outperforms the others in both algorithm efficiency and path quality. Besides, the path given by IAFSA has less unnecessary waypoints and is more suitable for ship navigation. Meanwhile, the IAFSA has been integrated into the GNC system of our model ship. The computer-based sea trials under the disturbances including wind, waves, and currents verify its feasibility in practical application.

In terms of the future work, two main issues can be addressed as following: (1) Considering the path planning in dynamic environment. In real situation there could be potential danger occurred during the voyage. This requires the ASVs to cope with the time-vary environment. (2) Implementation on practical ASV. The computer-based sea trial was conducted successfully in this study. However, to further prove its feasibility in practical application. Experiments on real ASV are still needed in the future.

5. Acknowledgements

The authors would like to thank the Editor-in-Chief, the Associate Editor, and the anonymous referees for their comments and suggestions. The work is mainly supported by the Institute of Structural Engineering in Zhejiang University.

6. Disclosure Statement

No potential conflict of interest was reported by authors.

7. References

- Durrant-Whyte H. 1994. Where am I? A tutorial on mobile vehicle localization. *Industrial Robot: An International Journal*. 21(2):11–16. <https://doi.org/10.1108/EUM0000000004145>

- European Maritime Safety Agency. 2019. Annual Overview of Marine Casualties and Incidents 2019. [Lisboa (Portugal): European Maritime Safety Agency; [accessed 2021 Nov 17].
<http://www.emsa.europa.eu/newsroom/latestnews/download/5854/3734/23.html>
- Fahey S, Luqi. 2016. Unmanned vehicles for anti-submarine warfare. In: OCEANS 2016 MTS/IEEE Monterey. p. 1–4. <https://doi.org/10.1109/OCEANS.2016.7761505>
- Fossen T. 2011. Handbook of Marine Craft Hydrodynamics and Motion Control.
<https://doi.org/10.1002/9781119994138>
- Guo X, Ji M, Zhao Z, Wen D, Zhang W. 2020. Global path planning and multi-objective path control for unmanned surface vehicle based on modified particle swarm optimization (PSO) algorithm. Ocean Engineering. 216:107693. <https://doi.org/10.1016/j.oceaneng.2020.107693>
- Hart PE, Nilsson NJ, Raphael B. 1968. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. IEEE Transactions on Systems Science and Cybernetics. 4(2):100–107.
<https://doi.org/10.1109/TSSC.1968.300136>
- Huang Y, Chen L, Chen P, Negenborn RR, Gelder PHAJM van. 2020. Ship collision avoidance methods: State-of-the-art. Safety Science. 121:451–473. <https://doi.org/10.1016/j.ssci.2019.09.018>
- Katzschmann RK, DelPreto J, MacCurdy R, Rus D. 2018. Exploration of underwater life with an acoustically controlled soft robotic fish. Science Robotics. 3(16):eaar3449.
<https://doi.org/10.1126/scirobotics.aar3449>
- Lazarowska A. 2015. Ship's Trajectory Planning for Collision Avoidance at Sea Based on Ant Colony Optimisation. Journal of Navigation. 68(2):291–307. <https://doi.org/10.1017/S0373463314000708>
- Lazarowska A. 2020. A Discrete Artificial Potential Field for Ship Trajectory Planning. Journal of Navigation. 73(1):233–251. <https://doi.org/10.1017/S0373463319000468>
- Liang C, Zhang X, Han X. 2020. Route planning and track keeping control for ships based on the leader-vertex ant colony and nonlinear feedback algorithms. Applied Ocean Research. 101:102239.
<https://doi.org/10.1016/j.apor.2020.102239>
- Liu Y, Song R, Bucknall R, Zhang X. 2019. Intelligent multi-task allocation and planning for multiple unmanned surface vehicles (USVs) using self-organising maps and fast marching method. Information Sciences. 496:180–197. <https://doi.org/10.1016/j.ins.2019.05.029>
- Lyridis DV. 2021. An improved ant colony optimization algorithm for unmanned surface vehicle local path planning with multi-modality constraints. Ocean Engineering. 241:109890.
<https://doi.org/10.1016/j.oceaneng.2021.109890>
- Pastore T, Djapic V. 2010. Improving autonomy and control of autonomous surface vehicles in port protection and mine countermeasure scenarios. Journal of Field Robotics. 27(6):903–914.
<https://doi.org/10.1002/rob.20353>
- Qian, Li X, Shao Z, Xin J. 2002. An Optimizing Method Based on Autonomous Animats: Fish-swarm Algorithm. Systems Engineering - Theory & Practice. 22(11):32. [https://doi.org/10.12011/1000-6788\(2002\)11-32](https://doi.org/10.12011/1000-6788(2002)11-32)
- Roberts GN, Sutton R. 2006. Advances in Unmanned Marine Vehicles.
<https://doi.org/10.1049/PBCE069E>
- Sang H, You Y, Sun X, Zhou Y, Liu F. 2021. The hybrid path planning algorithm based on improved A* and artificial potential field for unmanned surface vehicle formations. Ocean Engineering. 223:108709. <https://doi.org/10.1016/j.oceaneng.2021.108709>

- Sato Y, Maki T, Matsuda T, Sakamaki T. 2015. Detailed 3D seafloor imaging of Kagoshima Bay by AUV Tri-TON2. In: 2015 IEEE Underwater Technology (UT). p. 1–6.
<https://doi.org/10.1109/UT.2015.7108314>
- Shafer AJ, Benjamin MR, Leonard JJ, Curcio J. 2008. Autonomous cooperation of heterogeneous platforms for sea-based search tasks. In: OCEANS 2008. p. 1–10.
<https://doi.org/10.1109/OCEANS.2008.5152100>
- Vagale A, Bye RT, Oucheikh R, Osen OL, Fossen TI. 2021. Path planning and collision avoidance for autonomous surface vehicles II: a comparative study of algorithms. Journal of Marine Science and Technology. <https://doi.org/10.1007/s00773-020-00790-x>
- Wang N, Jin X, Er MJ. 2019. A multilayer path planner for a USV under complex marine environments. Ocean Engineering. 184:1–10. <https://doi.org/10.1016/j.oceaneng.2019.05.017>
- Wen N, Zhang R, Liu G, Wu J. 2020. Online Heuristically Planning for Relative Optimal Paths Using a Stochastic Algorithm for USVs. Journal of Navigation. 73(2):485–508.
<https://doi.org/10.1017/S0373463319000791>
- Wilde GA, Murphy RR. 2018. User Interface for Unmanned Surface Vehicles Used to Rescue Drowning Victims. In: 2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR). p. 1–8. <https://doi.org/10.1109/SSRR.2018.8468608>
- Xie L, Xue S, Zhang J, Zhang M, Tian W, Haugen S. 2019. A path planning approach based on multi-direction A* algorithm for ships navigating within wind farm waters. Ocean Engineering. 184:311–322. <https://doi.org/10.1016/j.oceaneng.2019.04.055>
- Yang TH, Hsiung SH, Kuo CH, Tsai YD, Peng KC, Peng KC, Hsieh YC, Shen ZJ, Feng J, Kuo C. 2018. Development of unmanned surface vehicle for water quality monitoring and measurement. In: Proceedings of 4th IEEE International Conference on Applied System Innovation 2018, ICASI 2018. p. 566–569. <https://doi.org/10.1109/ICASI.2018.8394316>
- Zhang Y, Guan G, Pu X. 2016. The Robot Path Planning Based on Improved Artificial Fish Swarm Algorithm. Liu F, editor. Mathematical Problems in Engineering. 2016:3297585.
<https://doi.org/10.1155/2016/3297585>
- Zhao S. 2018. USV based on automatic stabilizing function. Ordnance Ind Autom. 37(3):32–35.
- Zhong J, Li B, Li S, Yang F, Li P, Cui Y. 2021. Particle swarm optimization with orientation angle-based grouping for practical unmanned surface vehicle path planning. Applied Ocean Research. 111:102658. <https://doi.org/10.1016/j.apor.2021.102658>
- Zhou C, Gu S, Wen Y, Du Z, Xiao C, Huang L, Zhu M. 2020. The review unmanned surface vehicle path planning: Based on multi-modality constraint. Ocean Engineering. 200:107043.
<https://doi.org/10.1016/j.oceaneng.2020.107043>
- Zhou X, Yu X, Zhang Y, Luo Y, Peng X. 2021. Trajectory Planning and Tracking Strategy Applied to an Unmanned Ground Vehicle in the Presence of Obstacles. IEEE Transactions on Automation Science and Engineering. 18(4):1575–1589. <https://doi.org/10.1109/TASE.2020.3010887>