Achieving optimal-dynamic path planning for unmanned surface vehicles: a rational multi-objective approach and a sensory-vector re-planner

Liang Zhao [a], Yong Bai [a, *], Jeom Kee Paik [b, c,d]
[a] College of Civil Engineering and Architecture, Zhejiang University, Hangzhou 310058, People's Republic of China
[b] Department of Mechanical Engineering, University College London, London, the United Kingdom
[c] School of Maritime and Transportation, Ningbo University, Ningbo, People's Republic of China
[d] Yantai Research Institute, Harbin Engineering University, Yantai, People's Republic of China

Corresponding author: Yong Bai, baiyong@zju.edu.cn

Abstract

In this paper, under complex and unforeseen circumstances, a novel path planning framework incorporating the multi-objective optimization and a sensory-vector replanning strategy is created for an unmanned surface vehicle (USV). First, by encapsulating the intricate nature of ocean environment and ship dynamics, a nonlinear multi-objective path planning problem is designed, providing a comprehensive and in-depth portrayal of the underlying mechanism. By integrating the principles of candidate set random testing and adaptive crowding distance, an adaptive enhanced non-dominated sorting genetic algorithm (AENSGA-II) is devised to fully exploit the underlying optimization problem in constrained dynamics. To avoid over-subjective choice in the Pareto set, a fuzzy-linguistic satisfactory degree is deliberately designed, where the linguistic importance preference of the objectives is re-evaluated in the Pareto set, aiming at facilitating the decision-making. By inserting virtual sensory vector onto the USV, a seamless interface between global path and COLREG-compliant replanning mechanism is devised, thereby contributing to the entire hierarchical scheme. Eventually, the framework merits autonomous global-planning and local-reaction in an organically way. Comprehensive simulations and comparisons in various ocean scenarios demonstrate the effectiveness and superiority of the proposed path planning framework.

Keywords: Unmanned surface vehicles; Multi-objective optimization; Path planning; Genetic algorithm

1. Introduction

With artificial intelligence at the helm, the advancements of Unmanned Surface Vehicles (USVs) have been propelled to new heights, charting a course towards a brighter future of autonomous exploration and unlocking the secrets of our world and beyond (Öztürk et al., 2022; N. Wang et al., 2022; Wang and Xu, 2020; Zhao et al., 2022a, 2022b). Recently, USVs have been resorted to supporting various oceanic and marine applications such as the detection of radioactive chemicals (Chang et al., 2021), biological studies (Zhang et al., 2016), bathymetric surveys (Sahalan et al., 2016), measuring marine elements (temperature or salinity) (Cryer et al., 2020; Madeo et al., 2020), and observing water columns or warming trend (Smith et al., 2021). The level of autonomy pertaining to a USV ranges from manual control to full autonomy, with the path planning technique, connecting sensory hardware and control functionalities, playing a crucial role (N. Wang et al., 2022). However, navigating USVs is a complex task due to the uncertainties associated with the intricate ocean environment. The primary concern in deploying a USV is to attain secure navigation and obstacle avoidance, ensuring safety in the presence of other marine traffic. Consequently, in order to guarantee the efficiency and effectiveness of marine operations, it is imperative that the issue of path planning is properly addressed. (MahmoudZadeh et al., 2022; Zhao et al., 2022d).

Recently, booming academic advancements related to the path planning of USVs have emerged in the latest research works. Researchers have attempted to develop a variety of methods to solve the path planning problem including grid-based algorithms such as A* (Shah and Gupta, 2020; Song et al., 2019; Yu et al., 2021; Zhao et al., 2022c), D* (Han et al., 2022; Yao et al., 2021; Yu et al., 2022a, 2022b), fast marching square

(Beser and Yildirim, 2018; Liu et al., 2017; Tan et al., 2020), and meta-heuristic algorithms such as particle swarm optimization (PSO) (Guo et al., 2020; Krell et al., 2022), ant colony optimization (ACO) (Liang et al., 2020; Vahid and Dideban, 2022), genetic algorithm (GA) (Kim et al., 2017), and artificial fish swarm algorithm (AFSA) (F. Wang et al., 2022; Zhao et al., 2022d, 2022a). Grid-based methods involve the discretization of the environment into a set of grids, with each cell representing a potential location for the vehicle to traverse (Wang and Xu, 2020). The optimal path is then constructed by selecting a sequence of these cells. While this approach can be efficient for simple problems, it presents limitations in dealing with complex constraints and can result in a computationally intensive process, particularly in high-dimensional planning spaces (Lyridis, 2021). This highlights the importance of considering alternative methods that better handle the complexities of real-world scenarios. Note that meta-heuristic methods can satisfy complex constraints and multiple objectives, allowing for the formulation of sophisticated path planning problems (Nazarahari et al., 2019). However, their reliance on the linear-weighted method has been met with criticism. This method, although simple and widely used, has been proven to be subjective and may not accurately capture the decision maker's preferences (Lyridis, 2021; Sathiya et al., 2022). Additionally, the linear-weighted method is limited in its scalability and inflexibility in handling conflicting objectives, making it unsuitable for complex multi-objective problems, such as path planning. These limitations make it imperative to seek methods that better address the complex nature of multi-objective problems.

Alternatively, the multi-objective optimization (MOO) algorithms, such as NSGA and SPEA, may offer improved performance in complex multi-objective problems by presenting Pareto optimal solutions. Presently, the field of path planning has seen a surge of academic and technological advancements with a growing body of research dedicated to the application of multi-objective optimization techniques. In early studies, (Ahmed and Deb, 2013) applied the Non-dominated Sorting Genetic Algorithm (NSGA-II) in a discrete space, considering both the travel distance and path safety to attain Pareto optimality. (Davoodi et al., 2013) furthered this research (Ahmed and Deb, 2013) by taking path safety into account. More recently, (Ma et al., 2018) developed the Dynamic Augmented Particle Swarm Optimization algorithm to enhance path planning for USVs under current effects. To address non-holonomic constraints, (Sathiya et al., 2022) proposed the Fuzzy Enhanced Improved Multi-Objective Particle Swarm Optimization (FIMOPSO) algorithm, considering kino-dynamic and non-holonomic constraints. (Ntakolia and Iakovidis, 2021) developed a Swarm Intelligence Graph-Based Pathfinding algorithm for route planning and navigation for tourists, incorporating a novel multiple-criteria decision analysis to support decision making. (Lyridis, 2021) and (Ntakolia and Lyridis, 2022) conducted a series of studies on the fuzzy enhanced ant colony optimization method, achieving improved convergence speed and solution quality in path planning for USVs. (Ning et al., 2020) proposed a modified fuzzy dynamic risk of collision model for resolving collision avoidance and path planning challenges among multiple vessels. This model is based on the combination of time and space collision risk index and aligns more closely with actual ship applications. In addition, (Hu et al., 2020) introduced a multi-objective optimization approach for vessel path planning that unifies the COLREGs with the principles of good seamanship. This approach is particularly noteworthy as it follows a hierarchical, rather than simultaneous, approach to incorporating objectives.

Despite remarkable advancements in multi-objective algorithms, existing methods are still confronted with challenges of limited global searching ability and slow convergence speed, especially for non-convex problems like path planning. This limitation can be traced to the prevalent usage of conventional crowding distance (CD) methods and random initialization (Deng et al., 2022). The CD strategy limits the exploration of the solution space and can result in premature convergence to locally optimal solutions. Moreover, random initialization generates low-quality initial population in objective space and result in slow convergence and a high probability of getting stuck in local optimal trap (Wang et al., 2011). These limitations undermine the ability of multi-objective algorithms to effectively balance multiple objectives and find the globally optimal solution. Therefore, there is an imperative need for innovative techniques that can enhance the global searching capability and convergence rate of multi-objective algorithms for path planning.

Another issue that has rarely been addressed by the existing studies is how to choose a reasonable solution from the Pareto set. Path planning involves several conflicting objectives that have incompatible goals or contradict each other. Dealing with conflict objectives requires a trade-off between them, which can be difficult to achieve because they vary in optimization directions, rendering it impossible to achieve simultaneous optimization of all objectives. The selection of a reasonable solution from the Pareto set, therefore, presents a formidable challenge that often involves subjective preference for one set of objectives over another. Previous works have adopted a range of approaches, including the utilization of specific preferences to choose the lowest objective (e.g., Hu et al., 2020), the implementation of weight bias to model preferences (e.g., Ma et al., 2018), or failing to address the issue altogether (e.g., Ahmed and Deb, 2013; Davoodi et al., 2013; Sathiya et al., 2022). However, simply choosing the lowest objective is inherently over-subjective, and as such, lacks a rational basis for decision making. The consequence of this approach may result in unfavorable scenarios where one objective value becomes extremely high. For instance, the pursuit of the shortest path may result in a trajectory that is perilously close to obstacles, which is unacceptable in real-world applications (Ahmed and Deb, 2013). The weighted method, on the other hand, is also susceptible to high subjectivity in the selection of weight values. This approach has been met with criticism for not accurately reflecting the preferences (Lyridis, 2021; Nazarahari et al., 2019). Therefore, these limitations underscore the need for a more nuanced and sophisticated approach to decision-making in Pareto set, one that is grounded in a feasible understanding of the problem.

Moreover, achieving coordination between global planning and local avoidance presents a formidable challenge. On the one hand, the limited computational resources have laid harsh constraint on the replanning time, whereby the new path should be immediately transferred to the control system or the collision would be inevitable. It should be noted that this cannot be satisfied by most existing methods. Though previous researchers have made a great many attempts to reduce the computational burden (Han et al., 2022; Lyridis, 2021; Meng et al., 2022; Yao et al., 2021; Yu et al., 2022b), we are still of the opinion that they are not supportive for an effective path replanning. However, this paper solves the problem from another aspect, i.e., introducing a transition path to soften the harsh constraint on the replanning time. On the other hand, another issue is the sharp turning at the conjunction where the replanned and original paths meet. Such behavior is actually infeasible for a USV since immediate steering maneuver would lead to significant sideslip which deviates from the planned path (Wang and Xu, 2020). In this context, the continuous maneuvering should also be considered in achieving collision avoidance.

As observed from the foregoing works, although domestic and foreign researchers have conducted a series of studies in the path planning of USVs, it should be noted that past research has certain shortcomings:

(1) Designing approaches that could achieve global-planning and local-reaction jointly is still a challenging work within which the coordination between the two modules become a critical problem.
(2) For non-convex problem like path planning that involves multiple objectives, traditional MOO algorithms feature low convergence and lack sufficient global searching ability to facilitate diverse Pareto fronts.
(3) Since the simultaneously optimized indices are often reciprocally restrained, how to determine the solution preference in Pareto set for decision-making under the trade-offs remains a challenge.
(4) For an intensive overview in Table 1, none of the previously cited works has modeled the problem comprehensively, whereby the four general objectives (length/smoothness/energy/safety), vehicle constraints (nonholonomic/dynamic constraints), and environmental effects (currents) are omitted occasionally.

Inspired by the observations, this paper proposes a path planning framework to address the aforementioned challenging problems. The highlights of our work are illustrated as follows:

(1) A novel path planning framework is proposed to formulate global-planning and local-reaction in an organically way. We introduce the virtual sensory vector for environment perception and governing feasible actions of USVs under dynamically unforeseen environments. Seamlessly bridged by the transition Clothoid-path, not only sufficient time for replanning is provided but also guarantee the continuity of the course change. Augmented practicability has been achieved by extensive simulation and experimental evaluations under complex environments.
(2) By incorporating the candidate-based adaptive random testing initialization and adaptive crowding distance strategy, AENSGA-II merits strong global searching ability and facilitates the diverse Pareto

frontiers simultaneously. In such a case, optimal Pareto fronts are practically meaningful for decision-makers and can provide more high-quality solutions for the problem.

(3) By devising the fuzzy-linguistic satisfactory degree among the solutions, a novel method for determining the feasible solution in the Pareto set is developed. The linguistic importance preference between objectives is modeled as satisfactory degrees based on fuzzy rules, thereby contributing to the reasonable decision-making.

(4) Unlike the previous works, the problem model formulated in this work addresses more practical issues such as ocean currents, USV kinematics/non-holonomic constraints, dynamic obstacles, and COLREG rules. These elements are rarely considered comprehensively in the previous studies.

The remaining sections of the paper are organized as follows: Section 2 devises the path planning problem model. Section 3 proposes the AENSGA-II in combination with the COLREG-compliant strategy. In Section 4, simulation experiments for the path planning of USV are conducted in various scenarios. Finally, Section 5 concludes this research.

Table 1. Summary of recent literature

References	Length	Smooth	Safety	Energy	Current effects	Dynamic obstacles	COLREG
(Vahid and Dideban, 2022)	×	●	●	●	×	×	×
(Kim et al., 2017)	●	×	●	●	●	×	×
(Xia et al., 2020)	×	●	×	●	×	●	●
(Krell et al., 2022)	×	×	×	●	●	×	×
(Xue, 2022)	●	●	●	×	×	×	×
(Zhong et al., 2021)	●	×	×	×	×	×	×
(Liang et al., 2020)	●	●	×	×	×	×	×
(Zhao et al., 2022a)	●	●	×	×	×	×	×
(Zhao et al., 2022d)	●	●	●	×	×	×	×
(Ma et al., 2018)	●	●	●	●	●	×	×
(Yao et al., 2021)	●	●	●	×	×	●	×
(Shah and Gupta, 2020)	●	×	×	×	●	×	×
(Song et al., 2019)	●	●	×	×	×	×	×
(Xie et al., 2019)	●	●	×	×	×	×	×
(Yu et al., 2021)	●	×	●	×	×	●	×

2. Problem Formulation

Nomenclature for environment modelling

$\mathcal{M}$	Marine surface domain
$\mathcal{M}_f$	Obstacle-free motion area
$\mathcal{M}_o$	Obstacle area, $\mathcal{M}_o = \{O_1, O_2, \ldots, O_k\}$
P	Path, $P = \cup_{i=1}^{m} \boldsymbol{p}_i,\ i = 1,2,3,\ldots,m$
$\boldsymbol{p}_i$	Path segment, $\boldsymbol{p}_i = (x_\mathrm{i}, y_\mathrm{i}),\ i = 1,2,3,\ldots,m$
$\boldsymbol{p}_S$	Initial position, $\boldsymbol{p}_S = (x_S, y_S)$
$\boldsymbol{p}_E$	Destination position, $\boldsymbol{p}_E(x_E, y_E)$
O_i	Obstacles
k	Number of obstacles
$\boldsymbol{v}$	Velocity of USV

ψ	Heading angle of USV
$\boldsymbol{v}_c$	Velocity of currents
$\boldsymbol{v}_r$	Velocity of the USV considering the current effects

Nomenclature for constraints defined

$\boldsymbol{d}_i$	Position vector, $\boldsymbol{d}_i = [x_{i+1} - x_i,\ y_{i+1} - y_i, 0]^T$
$\vartheta_{i,i}$	Angle between $\boldsymbol{d}_i$ and path segment s_i
R_i	Turning radius of at $\boldsymbol{p}_i$
$\Delta\psi_i$	Change of heading angle at $\boldsymbol{p}_i$
$\Delta\psi_{max}$	Allowable maximum change of heading angle
R_{min}	Minimum turning radius

Nomenclature for objective functions

Objective 1 L	min $L = \sum_{i=1}^{m} L_i\,, i = 1,2,3,\ldots,m$
L_i	Length between s_i and s_{i-1}
Objective 2 θ	min $\theta = \sum_{m}^{i=2} \Delta\psi_i\,, i = 2,3,\ldots,m$
Objective 3 E	min $E = \sum_{i=1}^{m} L_i/\boldsymbol{v}_r \cdot f, i = 1,2,3,\ldots,m$
f	Fuel consumption per unit time (kg/min)
Objective 4 D	min $D = \sum_{i=1}^{m} D_i, i = 1,2,3,\ldots,m$
D_i	Safety value for path segment $\boldsymbol{p}_i$
d_i	Clearance between path segment $\boldsymbol{p}_i$ and its nearest obstacle O_i
d_{max}	Maximum clearance from the obstacles
d_{min}	Minimum clearance from the obstacles

2.1.Environment modeling

2.1.1. Motion area

First, we define the marine surface domain as $\mathcal{M}$ in Euclidean space $\mathbb{R}^2$. Suppose the Suppose the USV's path P consists of a sequence of linked elementary path segments $\boldsymbol{p}_i(i = 1,2,3,\ldots,m)$. Following the path P, the USV navigates from the initial position $\boldsymbol{p}_S(x_S, y_S)$ to the destination $\boldsymbol{p}_E(x_E, y_E)$ in the presence of numerous obstacles $\mathcal{M}_o = \{O_1, O_2, \ldots, O_k\}$ k is the number of obstacles). Therefore, the obstacle-free motion area of the USV is calculated as follows:

$$\mathcal{M}_f = \mathcal{M} - \mathcal{M}_o \tag{1}$$

Accordingly, to guarantee the safety, the generated path should be restricted to $\mathcal{M}_f$ which is given as:

$$P = \cup_{i=1}^{m} \boldsymbol{p}_i \subset \mathcal{M}_f \tag{2}$$

As can be seen from Eq. (1) and (2), the motion of USV is strictly bounded in the obstacle-free area $\mathcal{M}_f$.

2.1.2. Effects of currents

(Krell et al., 2022) and (Ma et al., 2018) have shown that energy consumption of USVs can be significantly affected by ocean currents. When engaging in the activities, the vessels favor the path which allows them to take full advantage of the currents to reduce the energy consumption. Suppose $\boldsymbol{v}$ is the velocity of the USV at $\boldsymbol{p}_i$ and the current velocity is $\boldsymbol{v}_c$, see Fig. 1. (a). Then the USV velocity considered the effects of the currents $\boldsymbol{v}_r$ can be calculated as:

$$\boldsymbol{v}_r = \boldsymbol{v} + \boldsymbol{v}_c \tag{3}$$

Remark 1. In some severe condition where the USV moves along with the currents, this is due to the extreme large value of v_c. In this research, we assume the USV can endure the negative effects of currents and satisfy the following constraint:

$$\boldsymbol{v} + \boldsymbol{v}_c \geq 0 \tag{4}$$

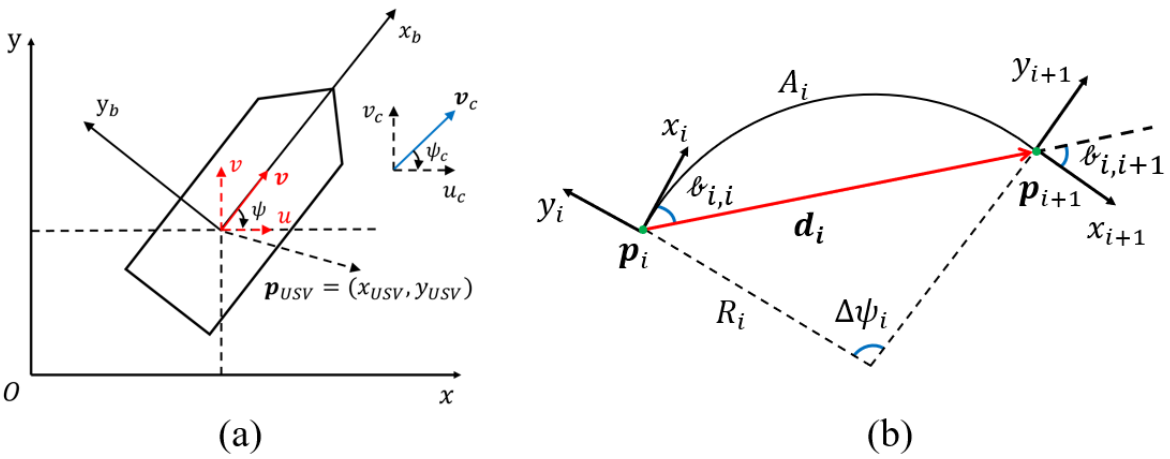


Fig. 1. (a) Coordinate system; (b) Definition of a path curve

2.2.Dynamic obstacles

In this paper, we assume the location of the obstacle will change with time. The movement of the dynamic obstacles in this work is considered to be a straight line with specific velocity ($\boldsymbol{v}_{DO}$) and direction (ψ_{DO}) according to the following relationship:

$$\begin{aligned} x_{DO}(t+1) &= x_{DO}(t) + \boldsymbol{v}_{DO} \cdot \cos\psi_{DO} \\ y_{DO}(t+1) &= y_{DO}(t) + \boldsymbol{v}_{DO} \cdot \sin\psi_{DO} \end{aligned} \tag{5}$$

where (x_{DO}, y_{DO}) is the coordinate of the dynamic obstacle, t denote the time step.

2.3.Motion Constraints

There are two constraints related to the USV's non-holonomic feature considered in this research: (1) It is imperative to ensure continuity of the path at turning points in order to mitigate abrupt changes. Failure to do so will result in the generation of an instantaneous extra control signal, thereby negatively impacting the tracking performance. (Song et al., 2019). (2) The curvature at any point on the path must be restricted in the dynamic bounds. For the USVs, the curvature is equivalent to the yaw rate, which should be less than the maximum acceleration provided by the propellers.

Definition 1. As denoted in Fig. 1. (b), suppose $\boldsymbol{d}_i = [x_{i+1} - x_i,\ y_{i+1} - y_i, 0]^T$ is the position vector between two consecutive poses and $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i+1}$ denote the path segments, then $\mathscr{b}_{i,i}$ and $\mathscr{b}_{i,i+1}$ define the angle between $\boldsymbol{d}_i$ and path segment $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i+1}$, respectively.

To achieve continuous path, the straight line and turning motions require two consecutive positions $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i+1}$ to be located on a common arc of constant curvature, which gives:

$$\mathscr{b}_{i,i} = \mathscr{b}_{i,i+1} \tag{6}$$

Definition 2. Suppose R_i and $\Delta\psi_i$ denote the turning radius and change of the heading angle at i^{th} path segments, respectively. L_i is the arc length defined by $L_i = R_i \Delta\psi_i$. Then, the maximum steering angle change $\Delta\psi_{max}$ causes a minimum turning radius R_{min}.

Therefore, the turning radius R_i is to be larger than its allowable minimum value, see the following expression:

$$R_i \geq R_{min} \tag{7}$$

2.4. Objective functions

2.4.1. *Path length*

The path P consists of several sequential path segments $\boldsymbol{p}_i(i = 1,2,3, \dots, m)$ from the start position $\boldsymbol{p}_S(x_S, y_S)$ to the destination $\boldsymbol{p}_E(x_E, y_E)$.

Definition 3. Let $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i-1}$ be the two consecutive points. The length between $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i-1}$ is $L_i = \|\boldsymbol{p}_i - \boldsymbol{p}_{i-1}\|$. Then the path is denoted as $L = \sum_{i=2}^{m} L_i$.

Therefore, the shortest path length objective can be defined as:

$$\min L \tag{8}$$

2.4.2. *Path smoothness*

The extra yaw-cost is deeply related to the USV motion control performance. Therefore, the smoothness objective function is introduced.

Definition 4. Let $\psi_i = atan\big((y_i - y_{i-1})/(x_i - x_{i-1})\big)$ and $\psi_{i-1} = atan\big((y_{i-1} - y_{i-2})/(x_{i-1} - x_{i-2})\big)$. The turning angle between $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i-1}$ within the path P is denoted as $\Delta\psi_i$. Then $\Delta\psi_i = |\psi_i - \psi_{i-1}|$.

The smoothest path requires the $\theta = \sum_{m}^{i=2} \Delta\psi_i \,, i = 2,3, \dots, m$ should be as small as possible. Consequently, the smoothest path criterion is defined as

$$\min \theta \tag{9}$$

2.4.3. *Energy consumption*

To reduce the energy consumption, not only does the USV get a path as short as possible, but also move along with the direction of currents.

Definition 5. Let $\boldsymbol{v}_r$ be the velocity of the USV with currents effects, f is the fuel consumption per unit time (kg/min), then the energy cost $E = \sum_{i=1}^{m} L_i/\boldsymbol{v}_r \cdot f$.

Therefore, the path with minimum energy consumption or least time goal is defined as:

$$\min \mathrm{E} \tag{10}$$

2.4.4. *The safest path*

Achieving the safest path for the USVs to traverse from its starting position to its final destination is imperative for guaranteeing its safety. We use the clearance from obstacles d_i to determine whether the solution is safe or not.

Definition 6. Suppose there are two invisible circle area with the radius of d_{min} and d_{max} around each path segment $\boldsymbol{p}_i$. The distance between each path segment $\boldsymbol{p}_i$ to its nearest obstacle O_i $(O_i \subset \mathcal{M}_o)$ is denoted as $d_i = \|\boldsymbol{p}_i, O_i\|, (i = 1,2,3, \dots, m)$.

Then the path safety of each segment can be expressed as:

$$D_i = \begin{cases} 0, & d_i \geq d_{max} \\ \dfrac{d_{max} - d_i}{d_{max} - d_{min}}, & d_{min} < d_i < d_{max}, i = 1,2,3, \dots, m \\ 1, & d_i \leq d_{min} \end{cases} \tag{11}$$

$$D = \operatorname{argmin} \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_i\}, \qquad i = 1,2,3, \dots, m$$

Consequently, the path safety is guaranteed when the minimum value of $\mathcal{D}_i$ is as small as possible, which gives the third objective:

$$\min D \tag{12}$$

2.5. Optimization problem statement

The goal of AENSGA-II is to find a shortest, smoothest, most energy-saving and safest path within the predefined constraints and ocean environment. Consequently, the optimization model for the problem is stated:

$$\min L = \sum_{i=1}^{m} L_i \, , i = 2,3, \dots , m \tag{13}$$

$$\min \theta = \sum_{m}^{i=2} \Delta\psi_i \, , i = 2,3, \dots , m \tag{14}$$

$$\min E = \sum_{i=1}^{m} L_i/\boldsymbol{v}_r \cdot f . , i = 1,2,3, \dots , m \tag{15}$$

$$\min D = \operatorname{argmin} \{D_1, D_2, \dots , D_m\} \, , i = 1,2,3, \dots , m \tag{16}$$

s.t.

$$\begin{aligned}
&\mathcal{M}_f = \mathcal{M} - \mathcal{M}_o \\
&P = \cup_{i=1}^{m} \boldsymbol{p}_i \subset \mathcal{M}_f , i = 1,2,3, \dots , m \\
&\boldsymbol{p}_1(x_1, y_1) = \boldsymbol{p}_S(x_S, y_S) \\
&\boldsymbol{p}_M(x_m, y_m) = \boldsymbol{p}_E(x_E, y_E) \\
&\boldsymbol{v}_r = \boldsymbol{v} + \boldsymbol{v}_c \\
&\boldsymbol{v} + \boldsymbol{v}_c \geq 0 \\
&L_i = \|\boldsymbol{p}_i - \boldsymbol{p}_{i-1}\| , \quad (i = 2,3, \dots , m) \\
&\psi_i = \operatorname{atan}\left(\frac{y_i - y_{i-1}}{x_i - x_{i-1}}\right) , \quad (i = 2,3, \dots , m) \\
&\Delta\psi_i = |\psi_i - \psi_{i-1}| , \quad (i = 2,3, \dots , m) \\
&d_i = \|\boldsymbol{p}_i , O_i\| , (i = 1,2,3, \dots , m) \\
&D_i = \begin{cases} 0, & d_i \geq d_{max} \\ \dfrac{d_{max} - d_i}{d_{max} - d_{min}}, & d_{min} < d_i < d_{max}, i = 1,2,3, \dots , m \\ 1, & d_i \leq d_{min} \end{cases} \\
&D = \operatorname{argmin} \{D_1, D_2, \dots , D_m\}, \qquad i = 1,2,3, \dots , m \\
&\mathcal{b}_{i,i} = \mathcal{b}_{i,i+1} , i = 1,2,3, \dots , m - 1 \\
&R_i \geq R_{min} , i = 2,3, \dots , m
\end{aligned} \tag{17}$$

Remark 2. The constraints consist of the moveable area (the first and second line of Eq. (17)), motion boundaries (the third and fourth line of Eq. (17)), current effects (the fifth and sixth line of Eq. (17)), and the expression of variables including the path length L_i ,the expressions of smoothness (ψ_i and $\Delta\psi_i$) and path safety (d_i and D_i). The last two lines in Eq. (17) represent the non-holonomic constraint and dynamic constraint. It is worth noting that the protocol constraints are introduced in Section 3.4.

Remark 3. Details of the variables in the model are expounded as below. For the first objective, the variables include $\boldsymbol{p}_i$, x_i, y_i, and i, where $\boldsymbol{p}_i$ is the path segment, x_i and y_i are the coordinates of $\boldsymbol{p}_i(x_i, y_i)$, and i is the number of path segments. For the second objective, its variables are x_i, y_i, ψ_i, $\Delta\psi_i$, and i, where ψ_i and $\Delta\psi_i$ can be obtained by the expressions in Eq. (17). The third objective includes $\boldsymbol{p}_i$, $\boldsymbol{v}$, $\boldsymbol{v}_c$, f, and i, where the value of f is a constant denoting the fuel consumption per minute, $\boldsymbol{v}$ is a constant with the same direction of the path at $\boldsymbol{p}_i$, and $\boldsymbol{v}_c$ is obtained by the predefined water current distribution function. For the last objective, the variables are $\boldsymbol{p}_i$, O_i, d_i, d_{min}, d_{max}, D_i and i, where O_i is the coordinate of the obstacle nearest $\boldsymbol{p}_i$, d_i and D_i can be obtained by the expressions in Eq. (17), d_{min} and d_{max} is the predefined safety distance and the largest distance from the obstacles. The upper and lower limits of the objectives are $0 \leq L \leq \infty$, $0 \leq \theta \leq \infty$, $0 \leq E \leq \infty$, and $0 \leq D \leq 1$,.

3. Methodology

There are four major conceptual parts in the proposed hierarchical framework, i.e., the multi-objective optimization problem model (introduced in Section 2), the AENSGA-II (introduced in Section 3.1/3.2), the fuzzy inference selector (introduced in Section 3.3), and the sensory vector based replanning strategy (introduced in Section 3.4). The hierarchical flowchart is shown in the end of this section, see Fig.11.

3.1.Framework of NSGA-II

The Non-dominated Sorting Genetic Algorithm II (NSGA-II), commonly referred to as the Fast and Elitist Multi-Objective Sorting Genetic Algorithm, is a refinement of its predecessor, the Non-dominated Sorting Genetic Algorithm (NSGA). The main steps of NSGA-II are described as follows (Deb et al., 2002).

Step 1: Initialize the population
While Gen < MaxGen **do**
Step 2: Compute the objective function and sort the non-dominated solutions
Step 3: Compute the crowding degree
Step 4: Optimization based on selection, crossover, and mutation operators
Step 5: Merge the child population and parent population
Step 6: Sort the non-dominated solutions and compute the crowding degree
Step 7: Select the individuals of population size that rank well and return to **Step 2**
End while
Step 8: Output the Pareto optimal set

3.2.AENSGA-II

The searching performance of conventional CD strategy and operators adopted by NSGA-II is relatively weak, as the crowding distance may not well reflect the density information around the individual, which decreases the solution diversity. Referring to AENSGA-II, by employing CSART-based initialization, local optima are prevented and convergence speed is enhanced. By introducing the ACD strategy and improved binary tournament selection, population diversity is maintained in the removal process.

3.2.1. Real-coded representation

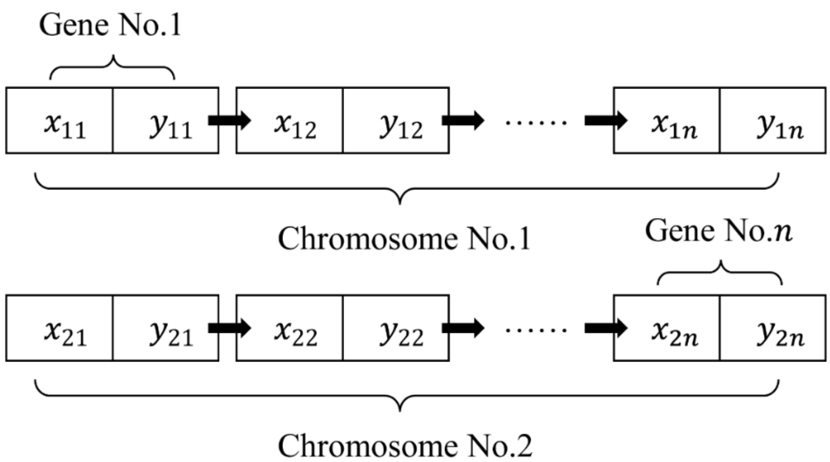


Fig. 2. Chromosome representation

Generally, there are two representations of a chromosome in the evolutionary algorithm, namely binary-coded and real-coded representations. In this study, we use the real-coded and take a chromosome as a complete solution, i.e., a path for the PP problem. It is a sequence of points beginning at a given origin position and ending at a particular destination point. To improve performance, chromosomes are represented as a single linked list in which each node stores a point. For example, for a path in a two-dimensional plane for a point

$\boldsymbol{p}_i = (x_i, y_i)$, we save x_i, y_i, and a pointer to the next point in the path. The algorithm will find the location of intermediate points and then a Clothoid curve (Silva and Grassi, 2018) is used to represent the path. Fig. 2 shows the data structure of a chromosome.

3.2.2. *Initialization using candidate set adaptive random testing (CSART)*

Randomly generating the starting population is easy to apply in NSGA-II. However, this will lead to the loss of the population diversity and easily falling into the local optimal in the later stage.

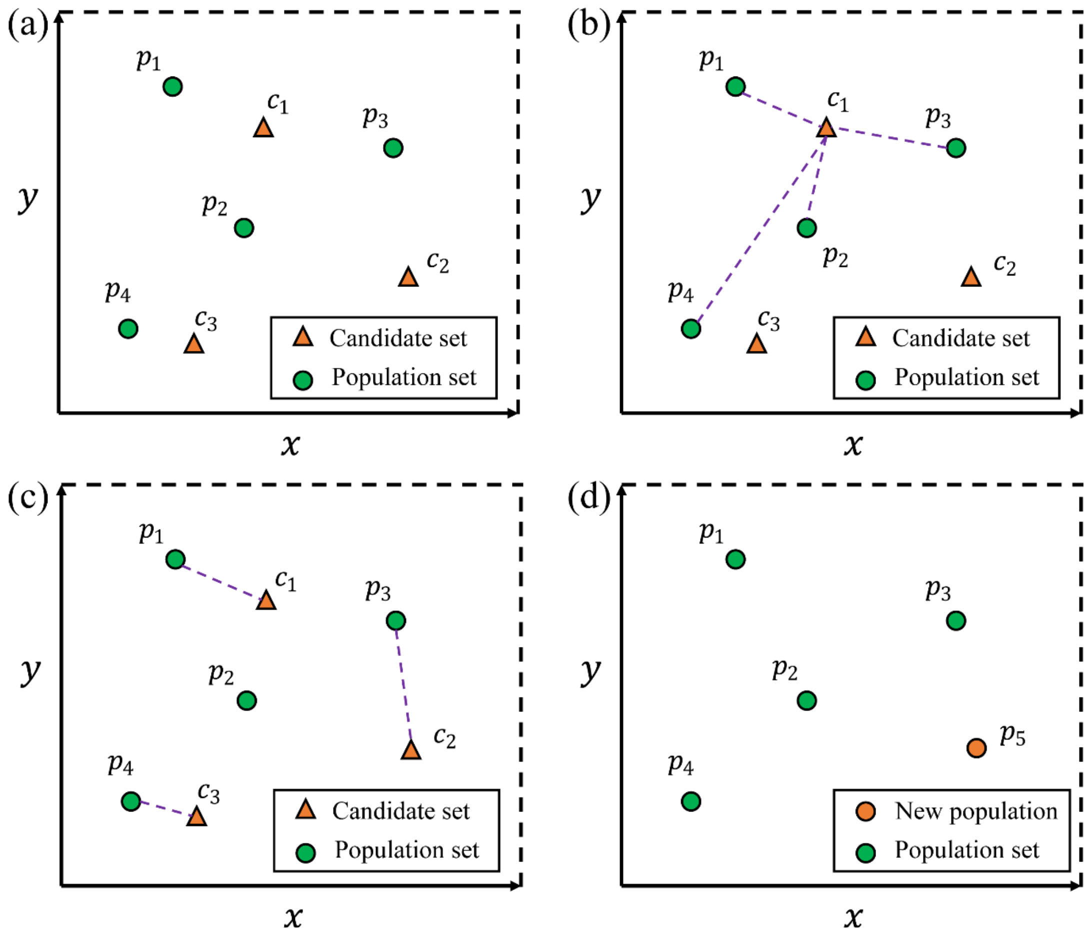


Fig. 3. CSART initialization (a) step 1; (b) step 2; (c) step 3; (4) step 4

Inspired by the failure analysis in software system, we adopted the candidate set adaptive random testing to modified the initialization of AENSGA-II. CSART is first applied to verify the quality of software systems (Chen et al., 2009). The basic idea is to generate a set of test cases that are widely distributed in the workspace. Likewise, we want a more dispersed distribution of the initial population to increase the diversity. Therefore, it is adopted by AENSGA-II. The main steps of the initialization process are illustrated as follows:

Step 1: Generating m candidate individuals $C = c_1, c_2, \dots, c_m$ randomly, see Fig. 3. (a).
Step 2: Calculating the distances between each candidate $C = c_1, c_2, \dots, c_m$ with the current individuals in the population set $P = p_1, p_2, \dots, p_n$, see Fig. 3. (b).
Step 3: Find the shortest distance between each candidate individuals $C = c_1, c_2, \dots, c_m$ with the population set $P = p_1, p_2, \dots, p_n$, see Fig. 3. (c).
Step 4: Choose the maximum value of the distances and put corresponding candidate individual into the population set P, see Fig. 3. (d).

The pseudocode of CSART initialization is presented in Algorithm 1.

Algorithm 1. Pseudocode of CSART initialization

Algorithm 1. CSART initialization
Input: $P = \{\}$ and $C = \{\}$

Output: initial population $P = \{p_1, p_2, \dots, p_{PopSize}\}$
Randomly generate q individuals using uniform distribution for $P = \{p_1, p_2, \dots, p_q\}$
while $q + 1 < PopSize$ **do**
    Randomly generate m individuals using uniform distribution for $C = c_1, c_2, \dots, c_m$
    for each candidate $c_j \in C$, $j = 1,2,3, \dots, m$ **do**
        Calculate the shortest distance d_j between $s_i \in S$ and c_j
    end for
    find $c_{max} \in C$ where $d_{max} > d_j, j = 1,2,3, \dots, m$
    $p_{q+1} = c_{max}$
    $P = \{p_1, p_2, \dots, p_{q+1}\}$
    $q = q + 1$
end while
return $P = \{p_1, p_2, \dots, p_{PopSize}\}$

3.2.3. Adaptive crowding distance (ACD) strategy

The NSGA-II uses crowding distance (CD) to remove the excess individuals found in the non-dominated set when the number of non-dominated solutions exceeds the population size. It can be calculated as follows:

$$CD_i = \frac{1}{N_{obj}} \sum_{k=1}^{N_{obj}} |f^k(x_{i+1}) - f^k(x_{i-1})| \tag{18}$$

where N_{obj} is the number of objectives, $f^k(x_{i+1})$ is the kth objective of the $i+1$th individual. The individuals with lower CD are preferred over the others in the removal process.

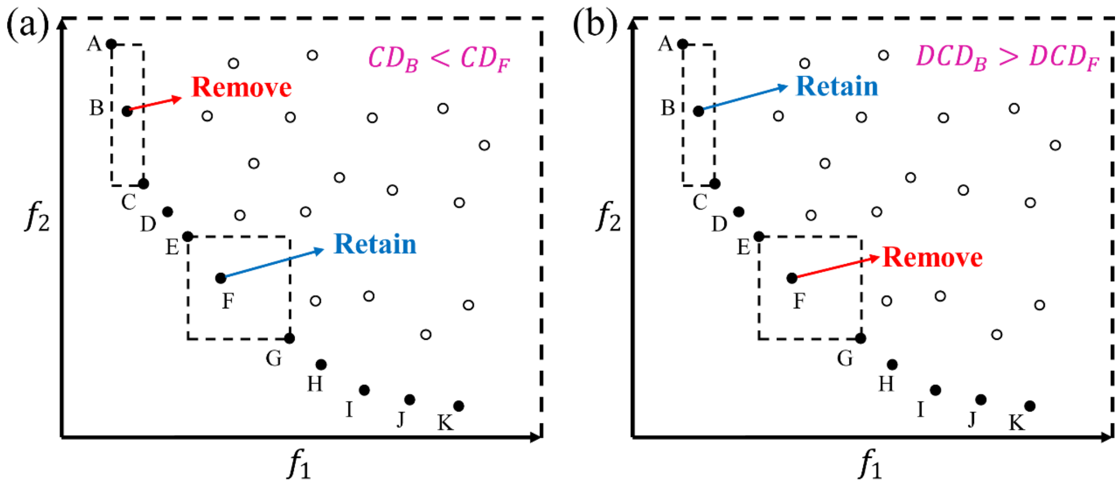


Fig. 4. (a) Traditional CD; (b) DCD

The major drawback of CD is the lack of uniform diversity in the solutions, which means some parts of pareto-front are too crowded and some parts are sparse (Dhanalakshmi et al., 2011). In Fig. 4. (a), CD denotes the half perimeter of the rectangular around the point. If we apply the traditional CD measurement, the individual B is removed because one side of the rectangle is very short which leads to smaller CD value. However, the CD of F is higher because the length of both sides is large, and F will be retained in the removal process. However, in order to reach good horizontal diversity, B should be the one retained and F should be removed. To address this issue, the adaptive crowding distance strategy is presented here. The CD value is modified into dynamic crowding distance:

$$DCD_i = \frac{CD_i}{\ln \frac{1}{Var_i}} \tag{19}$$

where CD is calculated by Eq. (18). Var_i is based on the following expression:

$$Var_i = \frac{1}{N_{obj}} \sum_{k=1}^{N_{obj}} (|f^k(x_{i+1}) - f^k(x_{i-1})| - CD_i)^2 \tag{20}$$

Var_i is the variance of CD values of neighboring individuals indexed by i. Var_i presents information about the level of difference of CD value of these objectives. An example is given to illustrate the process, in Fig. 4. (b) Var_i of B is larger than F which leads to a larger value of DCD. Therefore, B has more chance to retain and the diversity is maintained.

The pseudocode of adaptive crowding distance strategy is presented in Algorithm 2.

Algorithm 2. Pseudocode of adaptive crowding distance (ACD) strategy

Algorithm 2. ACD strategy

```
Input: PopSize    % population size
       P = {p_1, p_2, ..., p_N}    % non-dominated solutions in the current generation
       N    % Number of populations in non-dominated solutions
Output: P = {p_1, p_2, ..., p_PopSize}
if N <= PopSize then
   return    % Population number haven't exceeded
else
   while N > PopSize do
      calculate DCD_i (i = 1,2,3, ..., N) for all individuals
      sort the individuals based on DCD
      find p_k ∈ P where DCD_k < DCD_i, i = 1,2,3, ..., N
      P.pop(p_k)
      N = N − 1
   end while
end if
return
```

3.2.4. *Improved binary tournament selection*

A binary tournament selection is used in this research to improve the individual quality. Different from the traditional NSGA-II, we use the ranking and DCD to evaluate the individual. The operation is shown in Algorithm 3.

Algorithm 3. Pseudocode of improved binary tournament selection

Algorithm 3. Improved binary tournament selection

```
Input: P = {p_1, p_2, ..., p_PopSize}    % non-dominated solutions in the current generation
Output: p_i ∈ P    % better individual selected
for i in (0, PopSize) do
   rand choose two individuals: p_m, p_n
end for
if Rank(p_m) > Rank(p_n) then
   return p_m
else if Rank(p_m) < Rank(p_n) then
   return p_n
else if Rank(p_m) = Rank(p_n) then
      if DCD(p_m) > DCD(p_n) then
```

```
            return p_m
        else
            return p_n
        end if
    end if
    return
```

The flow chart of AENSGA-II is presented in Fig. 5.

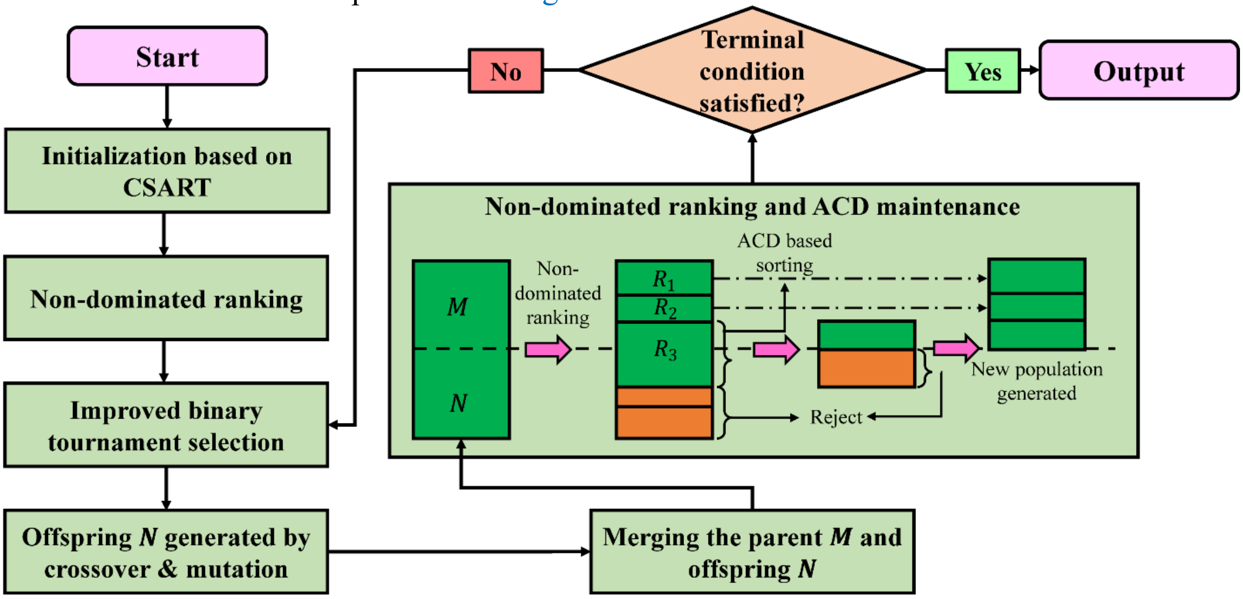


Fig. 5. Flowchart of AENSGA-II

3.3. Fuzzy satisfactory degree

Choosing the feasible solution in the Pareto set under the trade-off between the four considered objectives is challenging. Existing literature adopted the weight bias or simply choosing the lowest objective with preference is proved to be over-subjective (Lyridis, 2021; Ma et al., 2018) To select a reasonable solution for the USV, we design a fuzzy-based selection criterion to quantify the linguistic importance. Consequently, a fuzzy selector is devised in which the objectives undergo fuzzification, and a linguistic preference model is established.

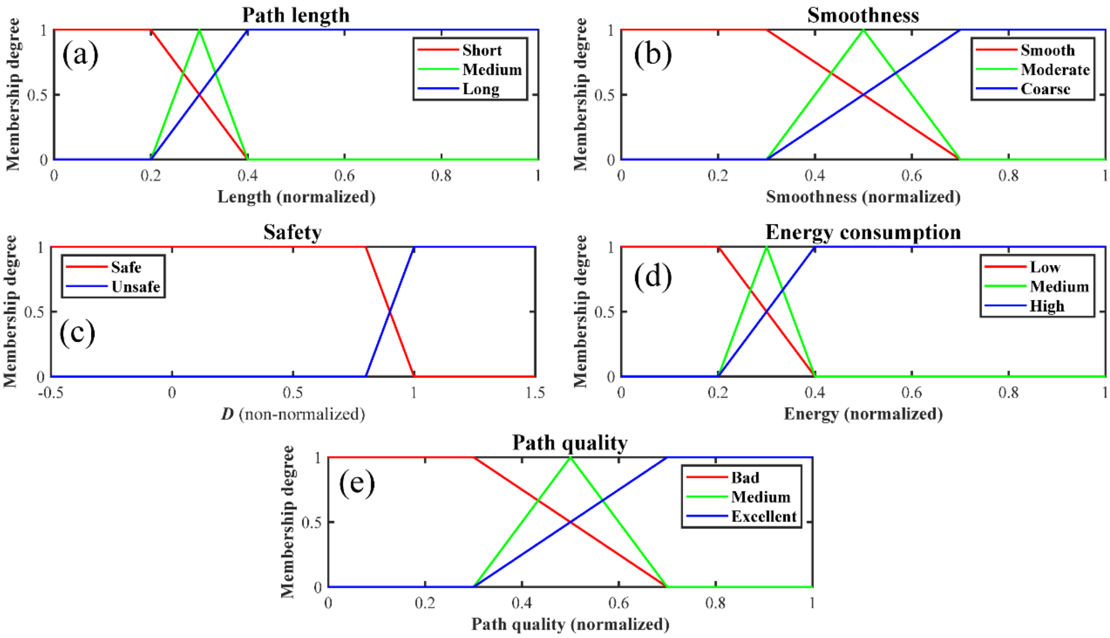


Fig. 6. Membership function: (a) Path length (normalized); (b) Smoothness (normalized); (c) Safety (non-normalized); (d) Energy (normalized); (e) Path quality

Three membership functions are deliberately designed, one for each objective function and one for the output variable that indicates the satisfactory degree of the solution. The inputs of the membership function are the four objective values and the output is the defuzzification value. The total distance is divided into three subsets {Short, Medium, Long}, the second objective Smoothness is classified into {Smooth, Moderate, Coarse}, the third objective Safety is classified into {Unsafe, Safe}, and the last objective Energy Consumption is classified into {Low, Medium, High}. Moreover, the output solution quality is divided into three subsets {Excellent, Medium, Bad}. Commonly, linear membership functions are defined for fuzzy relations, which are depicted in Fig. 6.

The process of fuzzy inference selection is illustrated as follows:

Step 1: input all the path values in the solution set and rescale using normalized root mean square error (Ntakolia and Lyridis, 2022).
Step 2: Fuzzified the crisp values and determine the membership degree according to Fig. 6 and Table 2.
Step 3: Evaluate the rules based on Mamdani inference system.
Step 4: Defuzzification based on Fig. 6 and output the path with the highest path quality value.

Table 2 Fuzzy rules

Quality	Length	Smoothness	Safety	Energy
Excellent	Short or medium	Smooth	Safe	Low
Excellent	Medium	Smooth or moderate	Safe	Low
Excellent	Short	Smooth	Safe	Low or medium
Medium	Medium	Moderate	Safe or Unsafe	Low or medium
Medium	Medium	Smooth or moderate	Unsafe	Low or medium
Medium	Medium	Smooth or moderate	Safe or Unsafe	Medium
Medium	Short or medium	Moderate	Unsafe	Low or medium
Medium	Short or medium	Moderate	Safe or Unsafe	Medium
Medium	Short or medium	Smooth or moderate	Unsafe	Medium
Bad	Long	Coarse or moderate	Safe or Unsafe	High or medium
Bad	Long or medium	Coarse	Safe or Unsafe	High or medium
Bad	Long or medium	Coarse or moderate	Unsafe	High or medium
Bad	Long or medium	Coarse or moderate	Safe or Unsafe	High

3.4. Replanning strategy based on sensory vector

3.4.1. Sensory vector structure

In this section, the virtual sensor deployment and a sensory-vector-based replanning strategy is proposed for avoiding obstacles in uncertain environment. The sensing module is performed by incorporating a virtual Lidar system that encompasses a circular region around the USV. The Lidar sensors are evenly distributed and are capable of covering a range of 30 degrees each, with a specified Sensing Range (SR) value, as shown in Fig. 7. This distance is provided by the USV's Lidar and is set to 50 m. Moreover, the dynamic obstacles are expanded by with a radius of the minimum distance d_{min} defined in Section 2, so that it can be considered as a circle area.

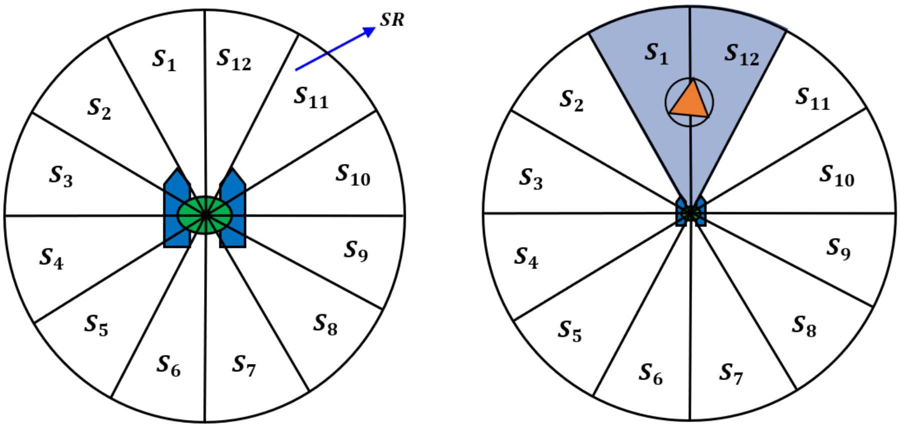


Fig. 7. Sensory structure

The sensory vector V_s is formed as:

$$V_s = [a(1), a(2), \dots, a(12)] \tag{21}$$

where $a(i), i = 1, 2, \dots, 12$ are variables with binary values. V_s reflects the status of an obstacle extant in an angle range $S_i, i = 1, 2, \dots, 12$. An example shown in the right figure of Fig. 7, with $a(1)$ and $a(12)$ equals logic “1”, this indicates that the obstacle is located inside SR and in the angle range S_1 and S_{12}, while the logic “0” represents a free space in the corresponding S_i.

3.4.2. Formulation of V_s

To find V_s, for each obstacle located inside SR, we first determine the potential collision risk of the obstacle. We adopted two indexes to measure the risk, i.e., Distance to Closest Point of Approach (DCPA) and Time to Closest Point of Approach (TCPA), which is determined by Eq. (22) and Eq. (23), respectively. As the DCPA becomes lower, the likelihood of a collision increases, and as the TCPA decreases, the necessity for immediate obstacle avoidance measures becomes urgent.

Assumption. 1: In this research, we assume the motion of the dynamic obstacle is known once it has been detected in SR range.

$$t_{CPA} = \frac{(\vec{p}_{USV} - \vec{p}_{DO}) \cdot (\vec{v}_{USV} - \vec{v}_{DO})}{\|\vec{v}_{USV} - \vec{v}_{DO}\|^2} \tag{22}$$

$$D_{CPA} = \|(\vec{p}_{USV} + \vec{v}_{USV} t_{CPA}) - (\vec{p}_{DO} + \vec{v}_{DO} t_{CPA})\| \tag{23}$$

Based on the above assumptions, the motion planners examines whether the situation is likely to lead to a collision in the short-term future, that is, by checking if the following equations are satisfied (Liu et al., 2022):

$$\begin{gathered} D_{CPA} \le SD \\ 0 \le t_{CPA} \le t_{th} \end{gathered} \tag{24}$$

where SD is the safe distance to check whether it is a collision, t_{th} is threshold value that indicates the emergency level, smaller t_{th} means more urgent.

Once the collision risk is detected by Eq. (24), then we shift the dynamic obstacle to the position of CPA (x_{CPA}, y_{CPA}) with an expanded circle whose radius is d_{min}, see Fig. 8. (b). Then the position angle θ can be determined by:

$$\theta = \text{atan2}(y_{CPA} - y_{USV}, x_{CPA} - x_{USV}) - \psi_{USV} \tag{25}$$

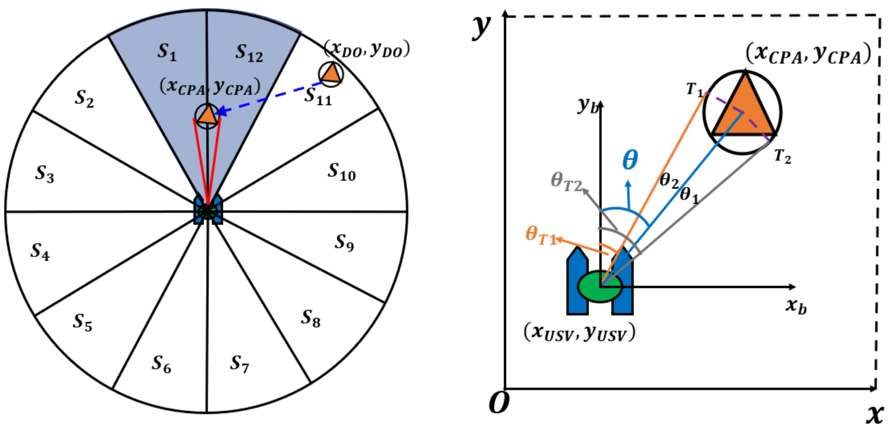
Since the angle $\theta_1 = \theta_2$ can be easily obtained by geometrical relationships based on Pythagoras theorem, and then the angle of two tangency points T_1 and T_2 can be calculated by:

$$\theta_{T1} = \theta + \theta_1$$
$$\theta_{T2} = \theta - \theta_1 \tag{26}$$

Once we had θ_{T1} and θ_{T2}, the sensory vector is found by setting the values of $a(i)$ in the V_s to logic "1" if the corresponding angle S_i is partly covered by θ_{T1} or θ_{T2}. An example shown in Fig. 8. (a), in this case:

$$V_s = [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1] \tag{27}$$

The CPA obstacle lies inside SR, and the angle difference $\theta_{T2} - \theta_{T1}$ lies in the ranges S_{11} and S_{12}.



(a) (b)

Fig. 8. (a) Example of CPA position; (b) Angle definition

3.4.3. Formulation of gap vector V_g based on COLREGs

The avoidance of obstacles is attained through the utilization of a gap vector V_g, which constitutes a binary vector where a logic value of "1" signifies the presence of an unavailable gap, and a logic value of "0" signifies the free gap. V_g and V_s share the same length. The USV chooses the gap required by the COLREG rules and moves towards destination.

Once condition (24) is satisfied, the rule selector identifies which COLREG rule is activated by examining the relative course ψ_r and position angle α between the USV and dynamic obstacles, see:

$$\psi_r = \psi_{DO} - \psi_{USV} \tag{28}$$

$$\alpha = \text{atan2}(y_{DO} - y_{USV}, x_{DO} - x_{USV}) - \psi_{USV} \tag{29}$$

We determine the collision situation based on Fig. 9 and Table 3.

Table 3 Judgement of the encounter scenario

Scenarios	Position angle α	Relative course ψ_r
Heading on	$\lvert\alpha\rvert \le 15°$	$\lvert\psi_r\rvert \ge 90°$
Overtaking	$\lvert\alpha\rvert \le 15°$	$\lvert\psi_r\rvert < 90°$
Overtaken	$\lvert\alpha\rvert \ge 112.5°$	-
Right-crossing	$15° < \alpha < 112.5°$	-
Left-crossing	$-112.5° < \alpha < -15°$	-

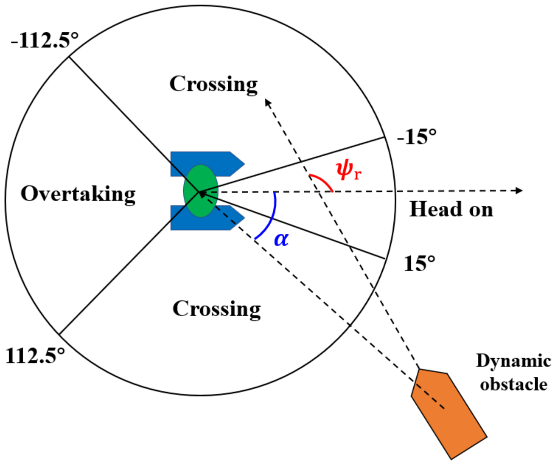


Fig. 9. Encounter scenario

After the identification of the COLREG situation, V_g first inherits the value of V_s, and then we implement the occupied gaps according to COLREGs.

Situation 1 (Heading on, Rule 14): In COLREG rule 14, the USV should bypass the obstacle from the starboard. In this case, the port side of the USV is set to be occupancy gaps, which yields

$$V_g(i) = 1,\ \ i = 1, 2, \ldots, 6 \tag{30}$$

Situation 2 (Overtake, Rule 13): In COLREG rule 13, the USV can overtake the DO from either side, which means no extra occupancy gaps are embedded:

$$V_g = V_s \tag{31}$$

Situation 3 (Crossing from the right, Rule 15): According to COLREG rule 15, the USV should bypass DO from the right. Similar to heading on situation, the port side of the USV is occupied, which yields

$$V_g(i) = 1,\ \ i = 1, 2, \ldots, 6 \tag{32}$$

Situation 4 (Crossing from the left, Rule 15): In COLREG rule 15, the USV should bypass DO from the port side. Consequently, the starboard of the USV is set to be occupancy gaps, which yields

$$V_g(i) = 1,\ \ i = 7, 8, \ldots, 12 \tag{33}$$

3.4.4. Formulation of transition path

Upon the construction of V_g, a number of free gaps, which serve as potential available positions for the USV, are generated. The angle β_i is determined by $\beta_i = i * 30$, where i stands as the index number of "0" in V_g. Then the next suggested position $\boldsymbol{p}_g(x_g, y_g)$ is determined, by which the transition path is generated and USV will evade the obstacles and start replanning from $\boldsymbol{p}_g(x_g, y_g)$ to the destination $\boldsymbol{p}_E(x_E, y_E)$. The procedure is explained in Algorithm 4.

Algorithm 4 Local obstacle avoidance

Algorithm 4. Local obstacle avoidance

1: **Input:** gap vector V_g; CPA position of obstacle $\boldsymbol{p}_{CPA\_O}(x_{CPA}, y_{CPA})$; destination $\boldsymbol{p}_E(x_E, y_E)$

Current position of USV $\boldsymbol{p}_{USV}(x_{USV}, y_{USV})$

2: **Output:** best permissible USV position $\boldsymbol{p}_g(x_g, y_g)$ to avoid obstacle

Calculate the moving distance $d_{md} = \|\boldsymbol{p}_{USV} - \boldsymbol{p}_{CPA\_O}\|$

3: $n = count(V_g == 0)$ % Calculate the number of permissible positions

$\beta = \{\beta_1, \beta_2, \dots, \beta_n\}$ % Calculate the angle of each permissible position

$d_{min} = \infty$ % Initialize the minimum distance to $\boldsymbol{p}_E(x_E, y_E)$

while $i < n$ **do**

  $x_{gi} = x_{USV} + d_{md} \times \cos \beta_i$

  $y_{gi} = y_{USV} + d_{md} \times \sin \beta_i$

  $d_i = \|\boldsymbol{p}_{gi} - \boldsymbol{p}_E\|$ % Calculate the distance from $\boldsymbol{p}_{gi}$ to $\boldsymbol{p}_E(x_E, y_E)$

  if $d_i < d_{min}$ **then**

    $\boldsymbol{p}_g = \boldsymbol{p}_{gi}$

    $d_{min} = d_i$

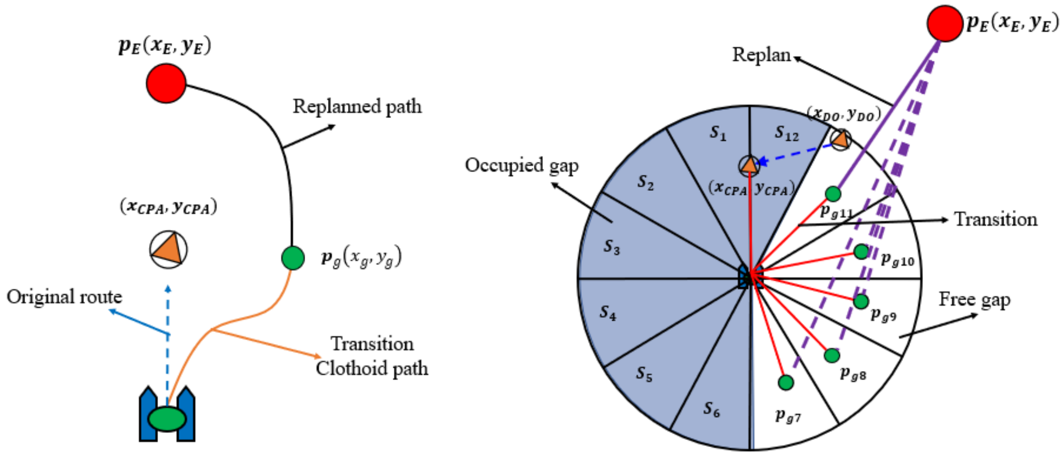
  end if

  $i = i + 1$

end while

return $\boldsymbol{p}_g(x_g, y_g)$

Once the next position $\boldsymbol{p}_g(x_g, y_g)$ is determined, a Clothoid curve (detail in (Silva and Grassi, 2018)) is generated between $\boldsymbol{p}_{USV}(x_{USV}, y_{USV})$ and $\boldsymbol{p}_g(x_g, y_g)$ immediately, see Fig. 10. (a). It is worth noting that the Clothoid curve is able to guarantee the continuity of the heading change and conform to the non-holonomic constraint of USVs. As the USV navigates through the transition route, the replanning between $\boldsymbol{p}_g(x_g, y_g)$ and $\boldsymbol{p}_E(x_E, y_E)$ is performed simultaneously.



(a) (b)

Fig. 10. (a) Illustration of transition path and replanning path; (b) Example of a crossing scenario

Fig. 10. (b). offers an illustrative example of the sensory vector method. First, an obstacle $\boldsymbol{p}_{DO}(x_{DO}, y_{DO})$ is detected in SR range and we evaluate its collision risk based on Eq. (24). If the collision risk is detected, we shift the obstacle to the CPA position and the sensory vector V_s and gap vector V_g are initially constructed as $V_s = V_g = [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]$. According to Fig. 9 and Table 3, the corresponding COLREG rule is identified and then we implement the rules to V_g. In this case, there are five available positions in free gaps in $V_g = [1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1]$, labelled p_{g11}, p_{g10}, p_{g9}, p_{g8}, p_{g7}. Then, we select the position $\boldsymbol{p}_g$ with the shortest distance to $\boldsymbol{p}_E(x_E, y_E)$ as the next position of USV, which is p_{g11}. Finally, path replanning is conducted between $\boldsymbol{p}_g$ and $\boldsymbol{p}_E$ while the USV navigates to $\boldsymbol{p}_g$.

To sum up the proposed model, we present Fig. 11 to illustrate the hierarchical structure of the methodology.

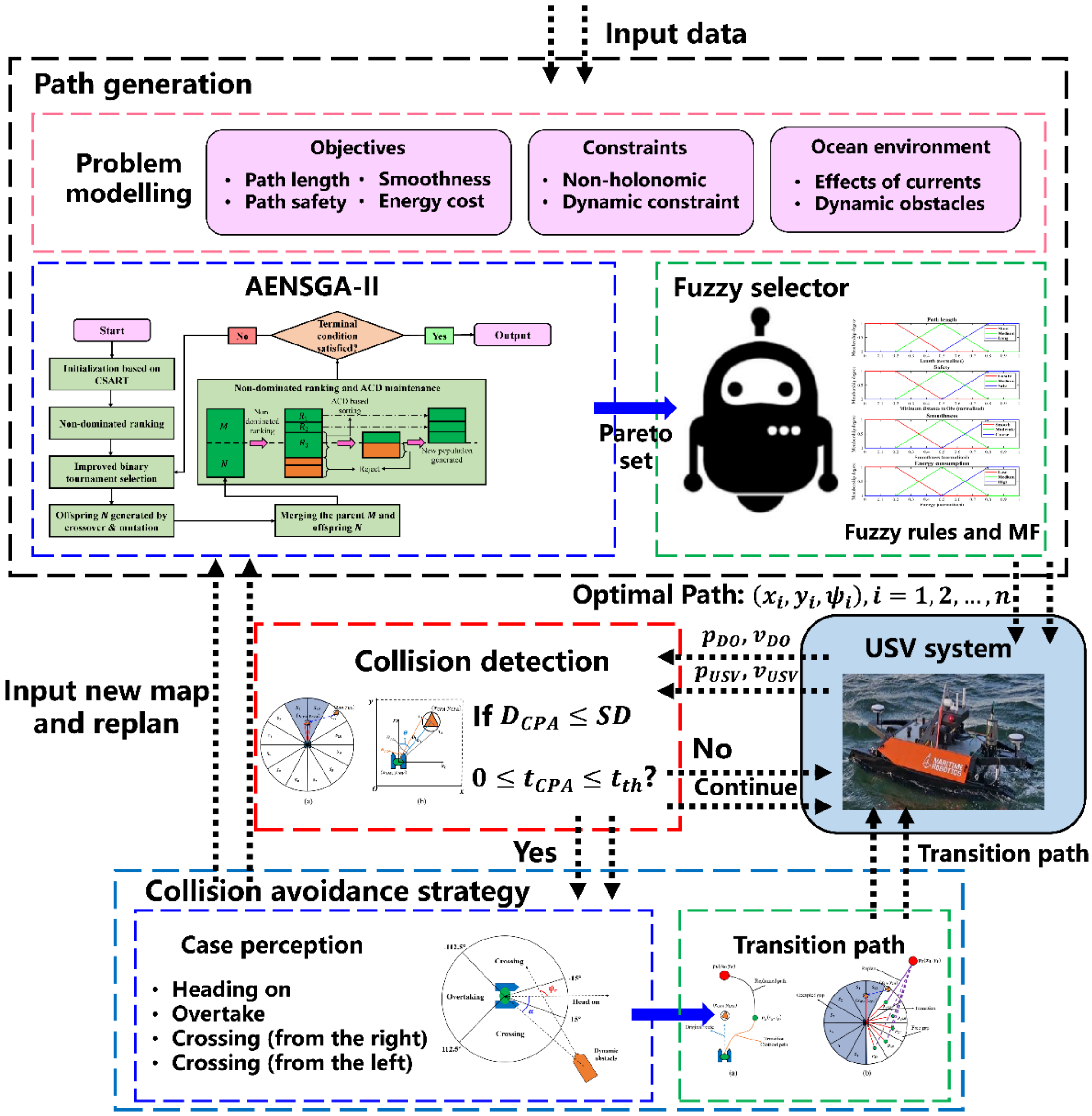


Fig. 11. Hierarchical framework of the proposed method

4. Results and Discussion

4.1. Convergence and diversity analysis

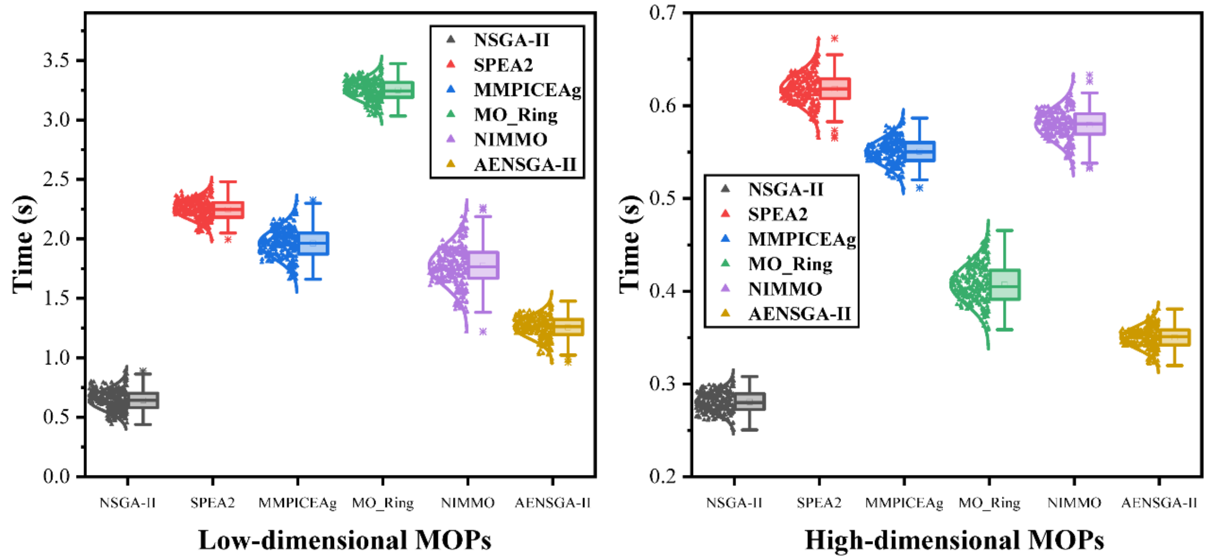
To validate the convergence characteristic and effectiveness of our proposed algorithm, we conduct the performance evaluation by using classical benchmark MOPs and state-of-the-art MOO algorithms. The benchmark MOPs include: (1) Five low-dimensional (d=2) MOPs in ZDT family problems (ZDT1, ZDT2, ZDT3, ZDT4, ZDT6). (2) Seven low-dimensional MOPs in DTLZ family problems (d=3). (3) Fourteen high-dimensional MOPs from DTLZ family problems with d=4, 5. The MOO algorithms we selected include: (1) Non-dominated sorting genetic algorithm (NSGA-II, (Deb et al., 2002)). (2) Improved strength Pareto evolutionary algorithm (SPEA2, (Zitzler et al., 2001)). (3) Preference-inspired coevolutionary algorithm (MMPICEAg, (Wang et al., 2021)). (4) MO_Ring_CSO_SCD (Wang et al., 2019). (5) Niching indicator-based multi-modal many-objective optimizer (NIMMO, (Tanabe and Ishibuchi, 2019)).

To evaluate the general performance of a MOO algorithm, a common approach is to assess the solutions in terms of convergence and diversity of the proximate Pareto frontier. Consequently, we adopt hypervolume (HV, (Jiang et al., 2015)) in the performance evaluation work. HV denotes the hypervolume contribution between the non-dominated solution set $X = x_1, x_2, x_3, \ldots, x_n$ and reference point P. It indicates the convergence and diversity of the solutions jointly. The larger HV value means better convergence and diversity performance.

The major parameter settings are outlined as follows: The population size N is 200 for all problems. The maximum number of function evaluations (FES) is set as 15000. In NSGA-II, SPEA2 and AENSGA-II, the crossover probability and mutation probability are $p_c = 0.8$ and $p_m = 1/n$ (n is the number of decision variables). The distribution indexes for crossover and mutation are $\eta_c = 20$ and $\eta_m = 20$. Other parameters are set as the default values reported in the references (Tanabe and Ishibuchi, 2019; Wang et al., 2021, 2019). The experimental results were acquired through the execution of 20 independent runs of each method. The Wilcoxon's rank sum test was used to determine if there were any statistically significant differences between the two algorithms at a 95% confidence level. The operation system is Windows 10 21H1, CPU is Intel(R) Core (TM) i7-8700 @ 3.20GHz 3.19 GHz, memory is 16GB, and the programs are running in MATLAB 2021a.

Table 4 presents the general performance scores and quantitative results of the MOO algorithms on the MOPs. The symbols "+", "_", or "≈" signify that the performance of the competitor algorithm is significantly superior, inferior, and comparable to that of AENSGA-II, respectively. The bold data represents the best result of the MOP. Fig. 12 shows the box-whisker plot in terms of effectiveness. Table 5 and Table 6 present the quantitative results of HV on 12 low-dimensional MOPs and 14 high-dimensional MOPs, respectively. From the corresponding simulation results, it allows the following conclusions to be drawn:

(1) Indicated by Table 4, AENSGA-II has presented satisfactory results on both low-dimensional and high-dimensional MOPs.
(2) As is shown in Table 5-6, AENSGA-II has shown better performance in terms of solution diversity than most of the MOO algorithms. This is mainly due to the implementation of ACD strategy and IBTS, which maintains diversity in the removal process. Referring to other algorithms, MMPICEAg and MO_Ring_CSO_SCD have slightly advantages in several cases (ZDT4, DTLZ2, DTLZ5, DTLZ6).
(3) In Fig. 12, AENSGA-II outperforms the other MOO algorithms except for NSGA-II in terms of effectiveness. This indicates that the computational cost of the ACD strategy is higher than conventional CD. Meanwhile, the algorithm robustness is shown by the IQR in Fig. 12 (size of the box in y-direction). It is observed that AENSGA-II obtained more stable results with respect to time cost.



(a) (b)

Fig. 12. Box-whisker plot of time cost on 12 low-dimensional MOPs

Table 4 HV and time scores on 12 low-dimensional MOPs. The symbols "+", "_" or "≈" indicate that the competitor algorithm performs significantly better, worse, and comparably to AENSGA-II.

Algorithms	HV score (+/-/≈)	Time score (+/-/≈)

12 low-dimensional MOPs	NSGA-II	0/10/2	11/0/1
	SPEA2	0/10/2	0/11/1
	MMPICEAg	1/6/5	0/10/2
	MO_Ring_CSO_SCD	0/7/5	2/7/3
	NIMMO	2/3/5	0/9/3
14 high-dimensional MOPs	NSGA-II	1/11/3	11/0/3
	SPEA2	1/13/0	0/11/3
	MMPICEAg	2/7/5	2/11/1
	MO_Ring_CSO_SCD	2/11/1	0/14/0
	NIMMO	2/7/5	3/8/3

Table 5 Quantitative results of HV on 12 low-dimensional MOPs over 20 independent runs

MOPs	NSGA-II	SPEA2	MMPICEAg	MO_Ring_CSO _SCD	NIMMO	AENSGA-II
ZDT1	6.192E-01 (−)	6.188E-01 (−)	6.312E-01 (≈)	6.291E-01 (−)	6.561E-01 (≈)	**6.572E-01**
ZDT2	3.092E-01 (−)	3.001E-01 (−)	3.202E-01 (≈)	3.218E-01 (≈)	3.240E-01 (≈)	**3.248E-01**
ZDT3	5.112E-01 (≈)	5.122E-01 (≈)	5.092E-01 (−)	5.142E-01 (≈)	5.124E-01 (≈)	**5.144E-01**
ZDT4	6.304E-01 (−)	6.407E-01 (−)	6.304E-01 (−)	6.224E-01 (−)	**6.529E-01** (≈)	6.512E-01
ZDT6	3.842E-01 (−)	3.724E-01 (−)	3.949E-01 (≈)	3.916E-01 (≈)	3.873E-01 (−)	**3.962E-01**
DTLZ1	6.748E-01 (−)	6.832E-01 (−)	7.638E-01 (≈)	7.204E-01 (−)	6.826E-01 (−)	**7.661E-01**
DTLZ2	3.381E-01 (−)	3.813E-01 (−)	**4.182E-01** (+)	3.894E-01 (−)	3.942E-01 (≈)	4.078E-01
DTLZ3	0.000E+00 (≈)	0.000E+00 (≈)	0.000E+00 (≈)	0.000E+00 (≈)	0.000E+00 (≈)	0.000E+00
DTLZ4	2.016E-01 (−)	2.035E-01 (−)	2.465E-01 (−)	2.328E-01 (−)	2.051E-01 (−)	**2.841E-01**
DTLZ5	8.956E-02 (−)	8.620E-02 (−)	9.060E-02 (−)	8.756E-02 (−)	**9.202E-02** (+)	9.112E-02
DTLZ6	3.065E-02 (−)	3.120E-02 (−)	3.085E-02 (−)	3.563E-02 (−)	**7.056E-02** (+)	5.692E-02
DTLZ7	1.862E-01 (−)	9.564E-02 (−)	1.983E-01 (≈)	1.965E-01 (≈)	1.970E-01 (≈)	**2.011E-01**
Mean	3.296E-01	3.246E-01	3.528E-01	3.451E-01	3.478E-01	**3.626E-01**

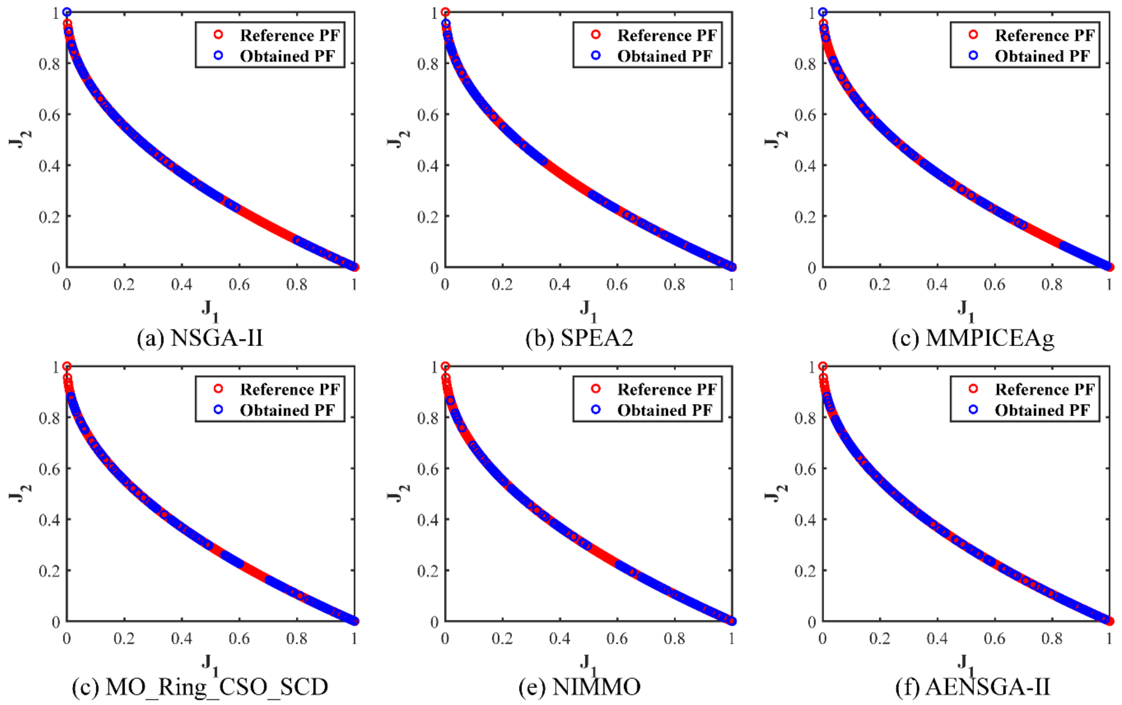


Fig. 13. PF distribution on ZDT1 for (a) NSGA-II; (b) SPEA2; (c) MMPICEAg; (d) MO_Ring_CSO_SCD;

(e) NIMMO; (f) AENSGA-II

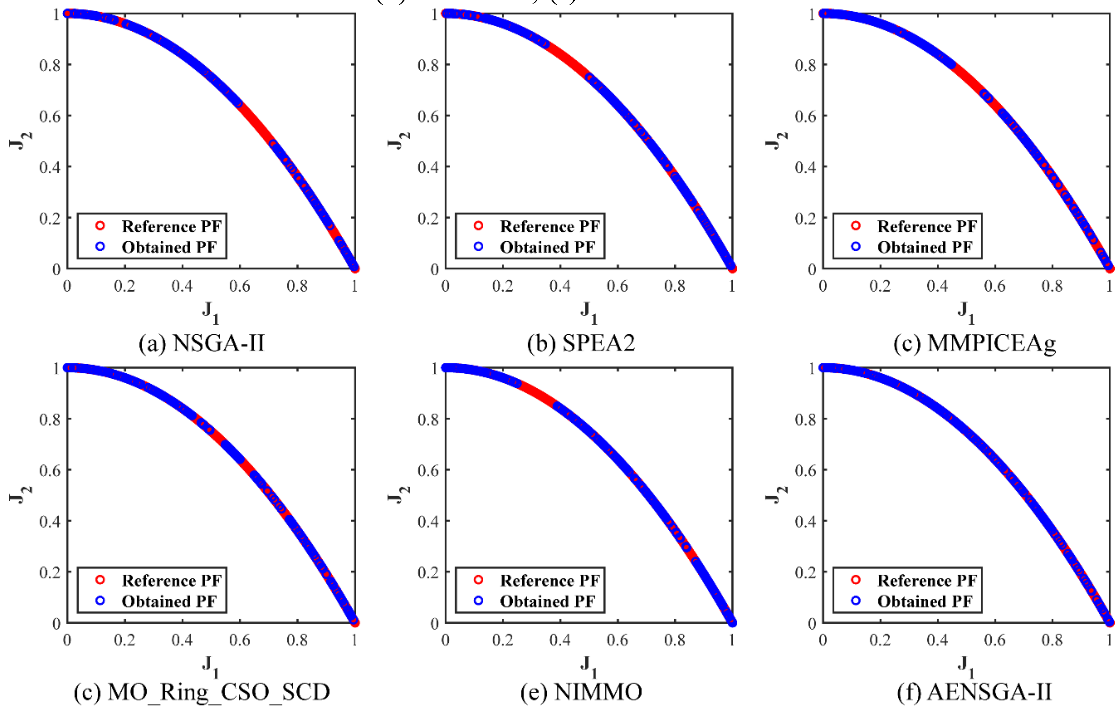


Fig. 14. PF distribution on ZDT3 for (a) NSGA-II; (b) SPEA2; (c) MMPICEAg; (d) MO_Ring_CSO_SCD; (e) NIMMO; (f) AENSGA-II

Table 6 Quantitative results of HV on 14 high-dimensional MOPs over 20 independent runs

MOPs	NSGA-II	SPEA2	MMPICEAg	MO_Ring_CSO_SCD	NIMMO	AENSGA-II
DTLZ1D4	4.523E-01 (−)	2.564E-01 (−)	8.623E-01 (≈)	5.265E-01 (−)	8.125E-01 (−)	**8.662E-01**
DTLZ2D4	6.265E-01 (−)	5.884E-01 (−)	6.904E-01 (−)	6.854E-01 (−)	**7.097E-01** (≈)	7.002E-01
DTLZ3D4	8.451E-01 (−)	6.207E-01 (−)	9.775E-01 (−)	9.325E-01 (−)	9.145E-01 (−)	**9.952E-01**
DTLZ4D4	4.524E-01 (+)	4.775E-01 (+)	4.023E-01 (≈)	**4.775E-01** (+)	4.524E-01 (+)	4.021E-01
DTLZ5D4	7.823E-01 (≈)	7.512E-01 (−)	7.765E-01 (−)	7.652E-01 (−)	7.732E-01 (−)	**7.934E-01**
DTLZ6D4	5.242E-01 (−)	8.254E-01 (−)	9.212E-01 (+)	**9.314E-01** (+)	8.852E-01 (−)	9.157E-01
DTLZ7D4	2.485E-01 (−)	2.354E-01 (−)	2.514E-01 (≈)	2.354E-01 (−)	2.492E-01 (≈)	**2.584E-01**
DTLZ1D5	0.000E+00 (−)	0.000E+00 (−)	8.770E-01 (−)	8.324E-01 (−)	8.916E-01 (−)	**9.264E-01**
DTLZ2D5	6.304E-01 (−)	6.425E-01 (−)	**8.893E-01** (≈)	6.926E-01 (−)	8.265E-01 (−)	8.872E-01
DTLZ3D5	0.000E+00 (−)	0.000E+00 (−)	9.862E-01 (≈)	9.901E-01 (≈)	9.910E-01 (≈)	**9.924E-01**
DTLZ4D5	8.956E-01 (−)	8.911E-01 (−)	**9.686E-01** (+)	9.328E-01 (−)	9.295E-01 (−)	9.497E-01
DTLZ5D5	7.821E-01 (−)	7.751E-01 (−)	7.925E-01 (≈)	7.733E-01 (−)	8.042E-01 (≈)	**8.051E-01**
DTLZ6D5	5.124E-01 (−)	6.314E-01 (−)	9.247E-01 (−)	9.214E-01 (−)	**9.571E-01** (+)	9.321E-01
DTLZ7D5	3.842E-01 (≈)	3.375E-01 (−)	**3.956E-01** (≈)	3.824E-01 (−)	3.914E-01 (≈)	3.921E-01
Mean	5.097E-01	5.023E-01	7.654E-01	7.199E-01	7.563E-01	**7.726E-01**

4.2.Simulation under static environment

In this section, simulations are provided to validate the performance of AENSGA-II in a static environment. We selected some state-of-the-art algorithms from existing reliable references in the comparative study, i.e., NSGA-II and EPSO. It is worth noting that both fixed currents and time-varying currents are considered. The testing environment is set the same as in Section 4.1.

The parameters of the simulations are set as follows:

- **Environment set (fixed currents)**: MapSize = 800*800 (m); Start = (340 m, 750 m); Goal = (360 m, 70 m); $d_{min} = 15$ m, $d_{max} = 25$ m; Direction of currents = -70°, Velocity of currents = 0.2 m/s.
- **Environment set (Time-varying currents)**: The Time-varying currents distribution is set as Eq. (34). $A = B = 10^{-3}$, $c = 0.003$.
- **USV model**: To calculate the energy consumption, suppose the USV parameters are: $L_{USV} = 2\,m$, $v_{USV} = 2\,m/s$.
- **AENSGA-II**: $N = 100$, $T_{max} = 100$, $p_c = 0.9$, $p_m = 1/\text{n}$, $\eta_c = 10$, $\eta_m = 20$, $R_{min} = 6$ m.
- **EPSO** (Alam et al., 2015): $N = 100$, $T_{max} = 100$, $c_1 = c_2 = 1.4995$
- **NSGA-II** (Ahmed and Deb, 2013): $N = 100$, $T_{max} = 100$, $p_c = 0.9$, $p_m = 1/\text{n}$, $\eta_c = 10$, $\eta_m = 20$.

$$\begin{aligned} u(x,y,t) &= A * y * \cos(x - ct) \\ v(x,y,t) &= B * y * \sin(x - ct) \end{aligned} \tag{34}$$

4.2.1. Fixed currents

Table 7 Calculation results

Algorithm	Convergence time (s)	Number of solutions
AENSGA-II	12	11
EPSO	23	8
NSGA-II	17	5

General calculation results are shown in Table 7. Fig. 15 present the visualized non-dominated solutions for each algorithm. The statistical measurements of the non-dominated solutions obtained by each algorithm are presented in Table 8-10. The visualized objective values are shown in Fig. 16. It is worth mentioning that, in Fig. 16, the length value is reduced by 10 times, and the energy cost is increased by 10 times to balance the scale. Fig. 15 (d) shows the comparison of the optimal solutions given by each algorithm.

From the corresponding simulation results, the following conclusions can be drawn:

(1) As is shown in Table 7, three algorithms successfully find a set of diverse Pareto optimal solutions. In particular, AESNGA-II generated more non-dominated solutions than the others, and the computational efficiency is satisfactory.

(2) As indicated in Table 8-10, AENSGA-II yielded solutions with better path quality (with mean objective value of 714.089 m, 19.850 m, 59.618° and 6.416 min). The mean objective value for EPSO and NSGA-II are 722.199 m, 16.538 m, 55.149°, 6.497 min and 737.272 m, 10.669 m, 77.062°, 6.550 min, respectively.

(3) Displayed in Fig. 15 (a), the paths presented by AENSGA-II take full advantage of the currents. Most of the paths are located at the right side of the central obstacle to flow with the currents, which leads to a shorter distance and lower energy consumption. As a consequence of the encoding method and curvature constraints, the paths created by AENSGA-II are composed of continuous points, whereas the paths generated by other algorithms consist of several line segments.

(4) Inspection of Fig. 15 (d) indicates that the fuzzy inference system is able to select a more reasonable path in the Pareto optimal set. Simply selecting the path with optimal value cannot guarantee its practicability.

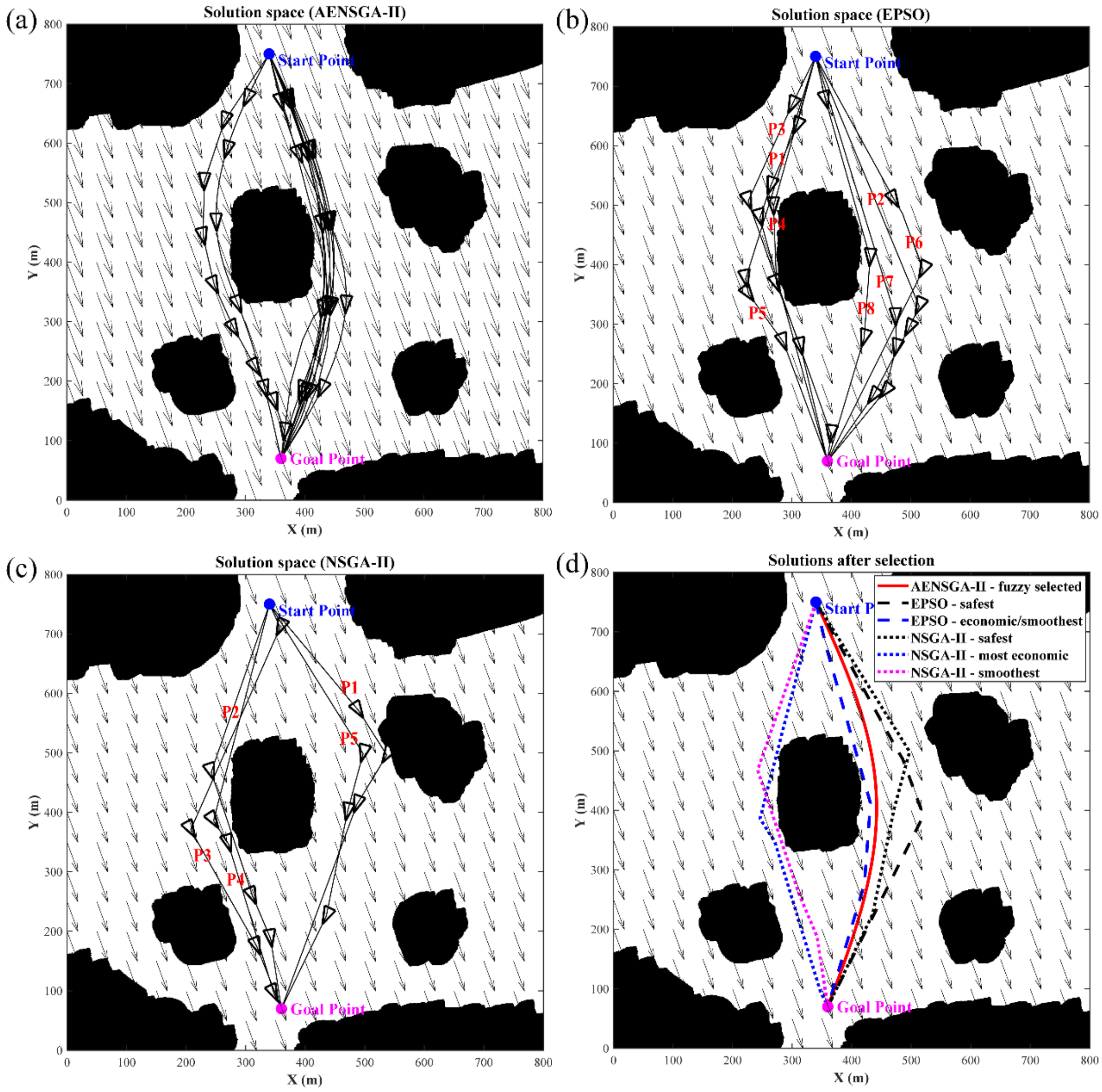


Fig. 15. Solutions (a) AENSGA-II (Solutions number is not given in the figure since they are too close to each other, they can be found in original data in Acknowledgement); (b) EPSO; (c)NSGA-II (d) Comparison of the optimal solutions

Table 8. Statistic measurements of the solutions obtained by AENSGA-II

Solutions	Length (m)	distToObs (m)	Smoothness	Energy cost (min)	Path quality
No.1	707.595	6.000	88.377	6.420	0.359
No.2	712.769	7.000	57.375	6.422	0.511
No.3	705.385	14.000	46.535	6.337	0.888
No.4	704.472	15.000	41.312	6.342	0.937
No.5	708.156	18.111	49.177	6.373	0.847
No.6	713.717	22.825	54.382	6.415	0.761
No.7	**709.711**	**24.021**	**45.431**	**6.366**	**0.950**
No.8	738.203	25.495	99.127	6.584	0.275
No.9	727.810	26.683	70.026	6.524	0.402
No.10	713.629	28.071	48.386	6.401	0.896
No.11	713.534	31.145	55.664	6.386	0.918
Mean	714.089	19.850	59.618	6.416	-

Note: The fuzzy selected path (with the highest path quality) is presented in bold

Table 9. Statistic measurements of the solutions obtained by EPSO

Solutions	Length	distToObs	Smoothness	Energy
No.1	702.892	2.236	47.562	6.377
No.2	746.720	41.146	57.571	6.718
No.3	724.014	4.243	72.357	6.540
No.4	700.327	1.414	44.236	6.314
No.5	720.441	16.125	68.703	6.362
No.6	760.407	**48.000**	59.020	6.767
No.7	727.743	14.142	56.456	6.591
No.8	**695.047**	5.000	**35.284**	**6.304**
Mean	722.199	16.538	55.149	6.497

Note: The best value of each objective is in bold

Table 10. Statistic measurements of the solutions obtained by NSGA-II

Solutions	Length	distToObs	Smoothness	Energy
No.1	781.380	5.831	82.165	6.856
No.2	707.111	4.243	**56.356**	6.359
No.3	747.471	2.828	109.677	6.662
No.4	707.108	3.000	71.980	**6.273**
No.5	743.291	**37.443**	65.132	6.602
Mean	737.272	10.669	77.0619	6.550

Note: The best value of each objective is in bold

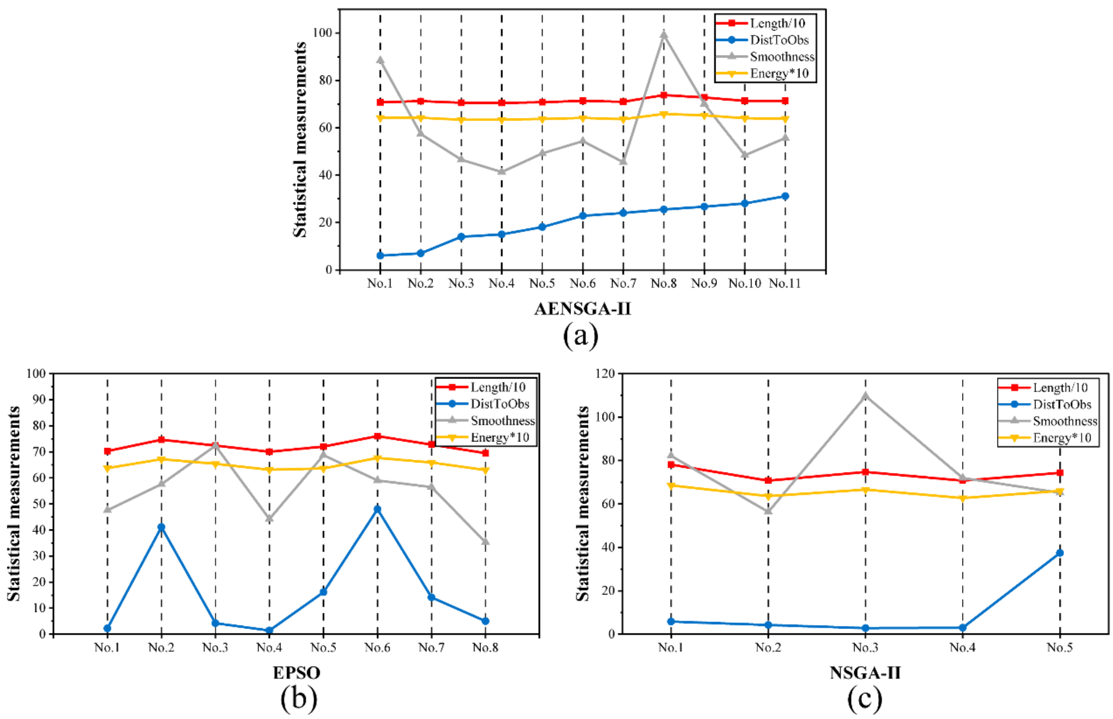


Fig. 16. (a) Scaled measures of 11 solutions obtained by AENSGA-II; (b) Scaled measures of 8 solutions obtained by EPSO; (c) Scaled measures of 5 solutions obtained by NSGA-II

4.2.2. Time-varying currents

In this section, the simulation is performed in time-varying ocean currents. The parameter settings and the testing environment are set as the same in Section 4.2.1.

Table 11. Calculation results

Algorithm	Convergence time (s)	Number of solutions
AENSGA-II	11	10
EPSO	27	8
NSGA-II	16	5

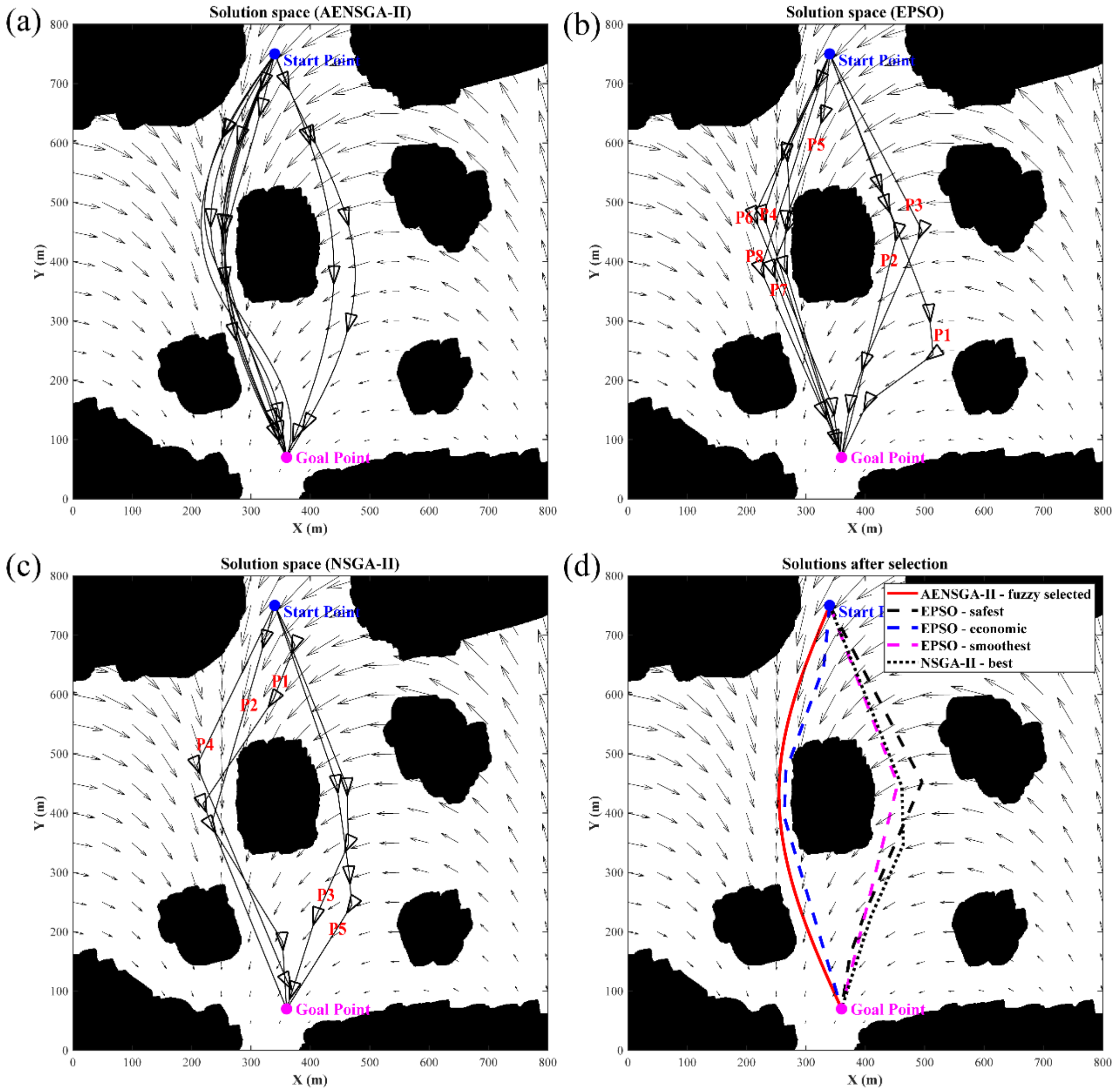


Fig. 17. Solution (a) AENSGA-II (Solution number is not given in the figure since they are too close to each other, they can be found in original data in Acknowledgement); (b) EPSO; (c) NSGA-II; (d) Comparison of the optimal solutions

The calculation results are provided in Table 11. The visualized non-dominated solutions are given in Fig. 17. Fig. 17 (d) presents the optimal path generated by each algorithm. The quantitative measurements for the objective values are presented in Table 12-14 and Fig. 18. From the simulation results, the findings are summarized as follows:

(1) In general, the results demonstrated in the time-varying ocean situation are in line with the previous studies. It appears that AENSGA-II has shown excellent performance in terms of efficiency and the solution count.

(2) In Fig. 17 (a), contrary to the results in fixed currents, most of the paths presented by AENSGA-II are located on the left side of the central obstacle, which is attributed to the distribution of the ocean currents. This indicates that our model takes advantage of the currents and prefers the routes with lower energy consumption.

(3) Comparing the mean objective value in Table 12-14, it is shown that AENSGA-II presents higher solution quality than the other approaches. This directly ties with the previous finding.

(4) As can be seen from Fig. 17 (d), the path selected by the fuzzy rules is more feasible than the other optimal

paths. It generates a path with low energy cost while ensures the path is sufficiently far from the obstacles and smooth enough for path tracking. It is shown that AENSGA-II combining with fuzzy rules have presented results with satisfactory.

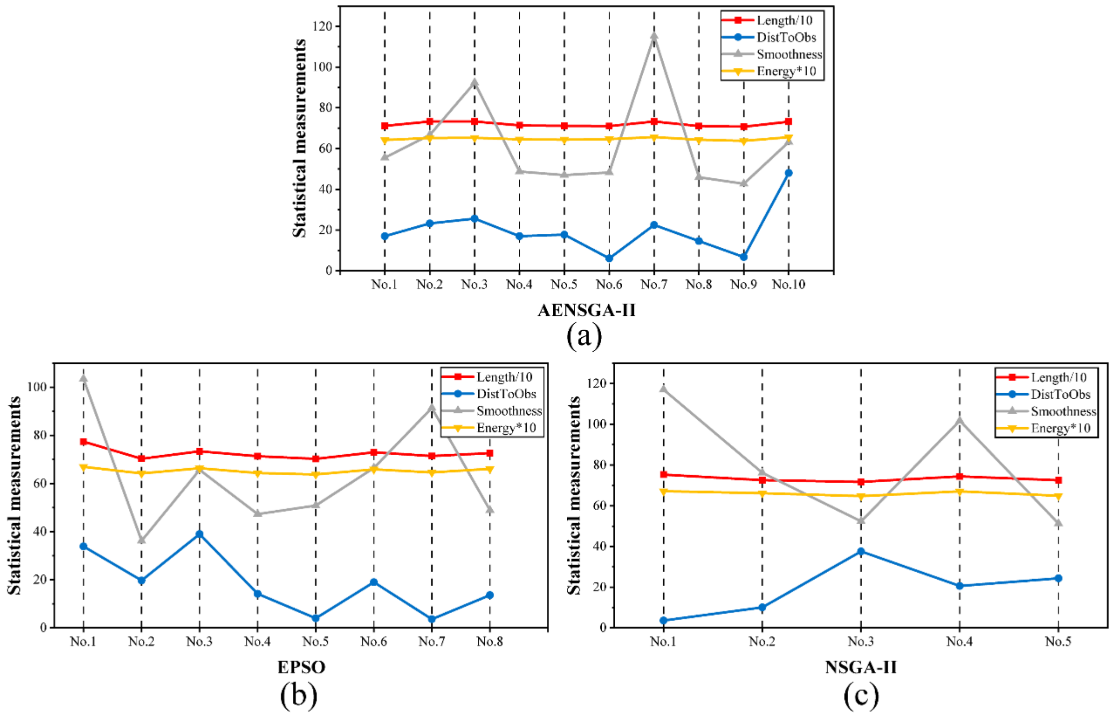


Fig. 18. (a) Scaled measures of 10 solutions obtained by AENSGA-II; (b) Scaled measures of 8 solutions obtained by EPSO; (c) Scaled measures of 5 solutions obtained by NSGA-II

Furthermore, inspection of the patterns shown in Fig. 16 and Fig. 18 presents some extra findings:

(1) Comparing the red and yellow lines in Fig. 16 and Fig. 18, it is evident that path length and energy cost are cooperative, which means they are optimized simultaneously. This is directly in line with the research findings of Davoodi et al., (2013).

(2) Taking a closer look to the red and grey in Fig. 18 (a), there is a tendency for a small increase in path length as smoothness grows (See No.3 and No.7). Similar patterns are depicted in Fig. 16 (c) (No.1, No.2, No.3, No.4) and Fig. 16 (a) (No.7, No.8, No.9) where the safety value is almost the same. This indicates that path smoothness can affect the path length to a certain degree.

(3) As shown by the red and blue lines in Fig. 16 (b), the path length and path safety have shown the same pattern. Similar patterns are also depicted in Fig. 18 (a) and (b). This implies that the path safety and path length are conflicting objectives, where there is a trade-off between them in finding the optimal path.

Table 12. Statistic measurements of the solutions obtained by AENSGA-II

Path number	Length	distToObs	Smoothness	Energy	Path quality
No.1	711.841	17.000	55.517	6.427	0.855
No.2	732.944	23.259	66.713	6.518	0.505
No.3	732.929	25.612	92.414	6.529	0.410
No.4	714.186	16.971	48.703	6.457	0.812
No.5	**711.695**	**17.720**	**46.969**	**6.440**	**0.873**
No.6	710.365	6.083	48.256	6.465	0.774
No.7	732.889	22.472	115.296	6.563	0.259
No.8	710.350	14.560	45.926	6.435	0.669
No.9	707.665	6.708	42.712	6.378	0.691
No.10	732.246	48.000	63.236	6.559	0.622

Mean	719.711	19.839	62.574	6.477	-

Note: The fuzzy selected path (with the highest path quality) is presented in bold

Table 13. Statistic measurements of the solutions obtained by EPSO

Path number	Length	distToObs	Smoothness	Energy
No.1	774.179	33.838	103.468	6.695
No.2	702.979	19.698	**36.174**	6.420
No.3	733.995	**39.000**	65.667	6.637
No.4	713.556	14.142	47.291	6.435
No.5	702.611	4.000	50.842	**6.378**
No.6	729.790	18.974	66.470	6.587
No.7	714.409	3.606	91.388	6.461
No.8	726.225	13.601	48.918	6.608
Mean	724.718	18.357	63.777	6.528

Note: The best value of each objective is in bold

Table 14 Statistic measurements of the solutions obtained by NSGA-II

Path number	Length	distToObs	Smoothness	Energy
Path No.1	752.589	3.606	117.028	6.713
Path No.2	725.706	10.050	76.149	6.614
Path No.3	**716.896**	**37.577**	**52.347**	**6.477**
Path No.4	743.934	20.591	101.757	6.707
Path No.5	725.298	24.352	51.214	6.488
Mean	732.884	19.235	79.699	6.600

Note: The best value of each objective is in bold

4.3.Simulation under dynamic environment

In this subsection, on the basis of Section 4.2, the effectiveness of our proposed model is demonstrated by avoiding unknown dynamic obstacles. The results are provided by conducting experiments on a prototype USV Otter (see www.maritimerobotics.com, Table 15 shows the particulars of the vessel) in time-varying environment. The model consists of three basic subsystems: the line of sight (LOS) guidance system, the PID controller, and extended Kalman filter for observer, please find the details in the author's previous publication (Zhao et al., 2022a, 2022c). It is worth noting that the tracking and replanning can be achieved simultaneously using Parallel Computing Toolbox in MATLAB.

Table 15. Maneuvering derivatives of the USV model

Inertial related	Value	Damping related	Value
m_{11}	85.28	d_{11}	-77.55
m_{22}	162.50	d_{22}	-0.02
m_{33}	41.45	d_{33}	-41.45
m_{23}	4.58	d_{23}	-62.07
m_{32}	4.58	d_{32}	-263.87

The environment and parameters of the simulations are set as follows:

- **Environment set (Case1)**: MapSize = 800*800 (m); Start = (340 m, 750 m); Goal = (360 m, 70 m); currents are set as the same in Section 4.2.2.
- **Environment set (Case 2)**: MapSize = 800*800 (m); Start = (340 m, 750 m); Goal = (360 m, 70 m);

currents are set as the same in Section 4.2.2.

- **AENSGA-II**: $N = 100$, $T_{max} = 100$, $p_c = 0.9$, $p_m = 1/\mathrm{n}$, $\eta_c = 10$, $\eta_m = 20$, $d_{min} = 15$ m, $d_{max} = 25$ m, $R_{min} = 6$ m, $SD = 15$ m, $t_{th} = 50\ s$

The dynamic obstacles are set as presented in Table 16.

Table 16. Setting of dynamic obstacles

	Dynamic obstacles	Position (m)	Speed (m/s)	Direction (deg)
Case 1	DO1	(85, 550)	1 m/s	0
	DO2	(212, 536)	0.5 m/s	-75
Case 2	DO1	(432, 590)	1 m/s	180
	DO2	(255, 300)	0.5 m/s	90

Table 17. Quantitative results of obstacle avoidance

Case	Path	Dynamic obstacles	TCPA (s)	DCPA (m)
Case 1	Original	DO1	183	14.74
		DO2	324	9.66
	Replanned	DO1	172	41.78
		DO2	353	38.29
Case 2	Original	DO1	157	5.75
		DO2	311	0.63
	Replanned	DO1	126	37.31
		DO2	337	31.71

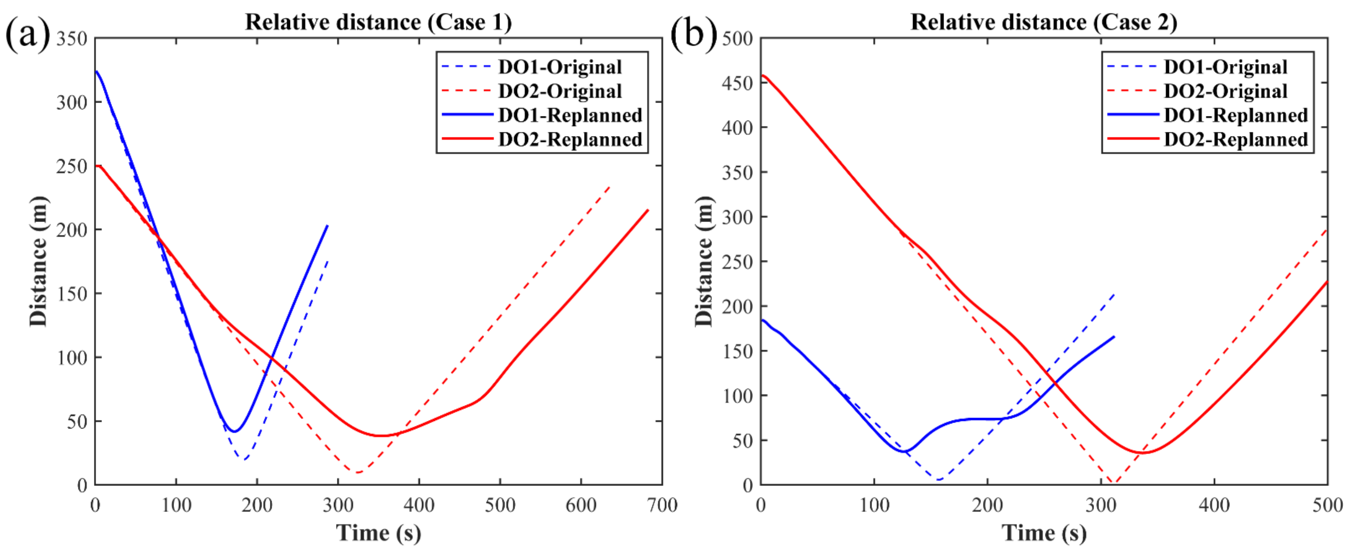


Fig. 19. Relative distance of (a) Case 1, (b) Case 2

Table 18. Time spends on replanning and transition path

Case	Dynamic obstacles being avoided	Replanning (s)	Transition path (s)
Case 1	DO1	18.6	38.8
	DO2	17.3	31.2
Case 2	DO1	19.1	40.1
	DO2	18.4	29.2

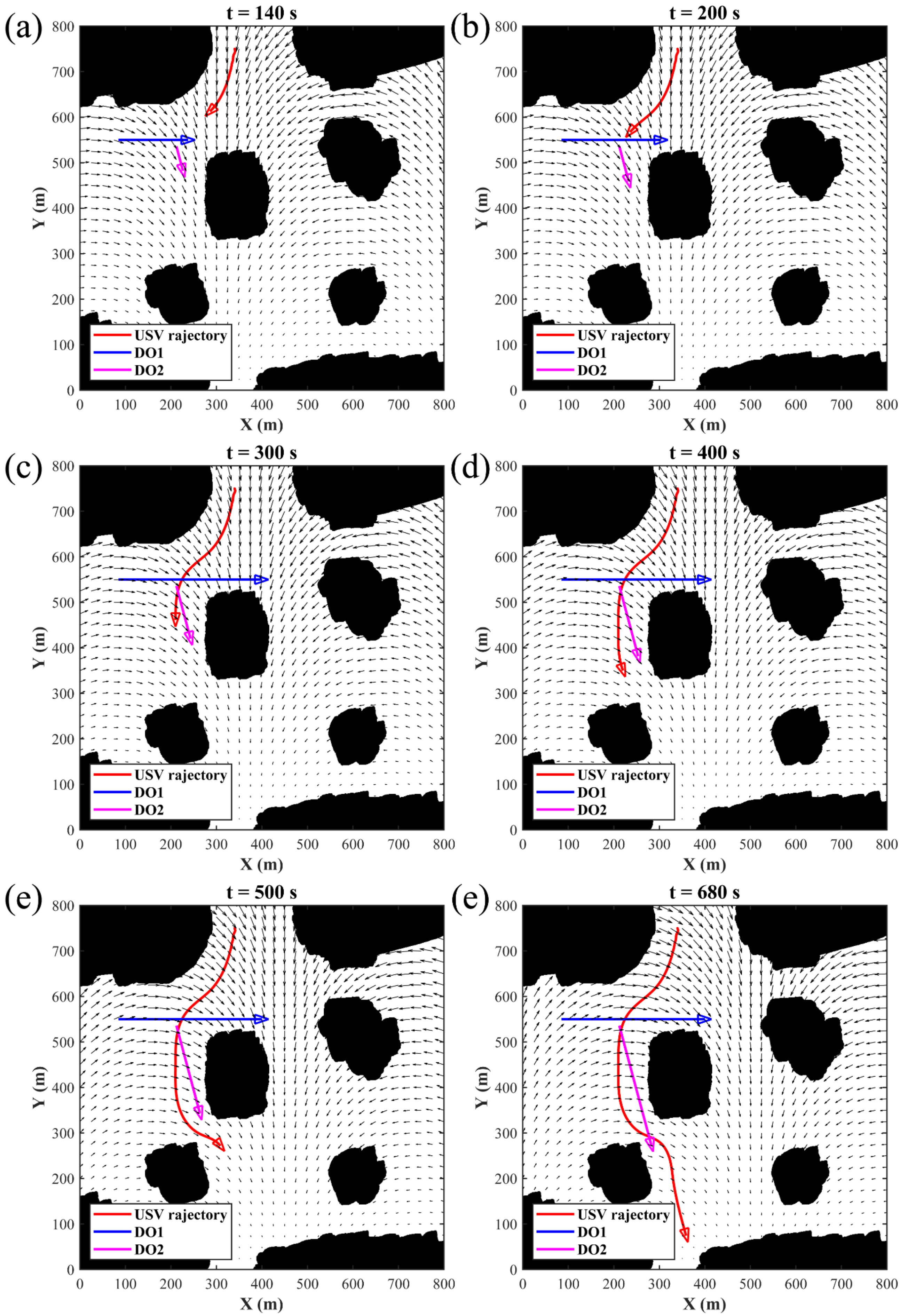


Fig. 20. Visualized trajectory of Case 1

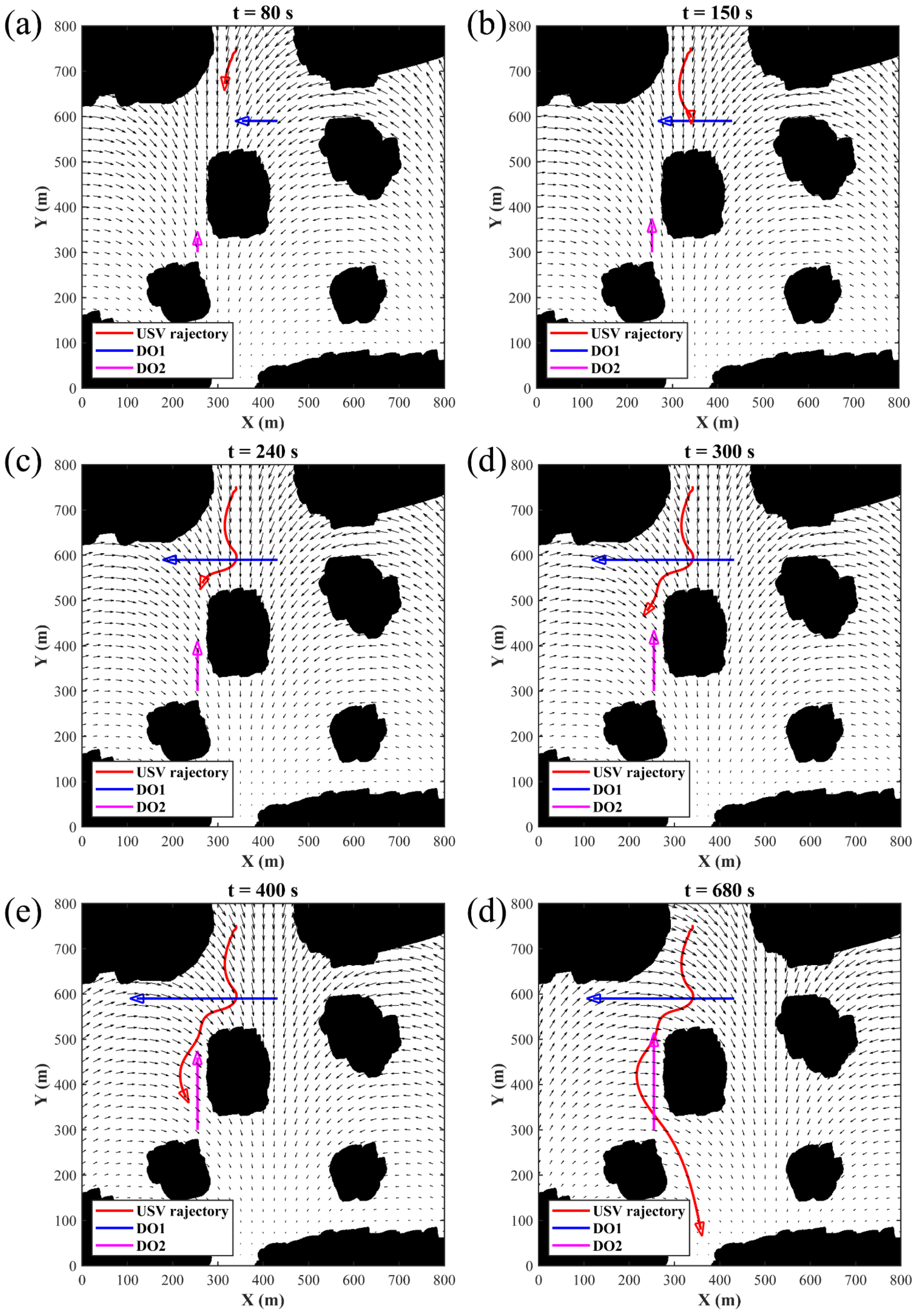


Fig. 21. Visualized trajectory of Case 2

The relative movements of the USV and dynamic obstacles are reflected in Table 17 and Fig. 19. The time cost of the replanning and navigating the transition path is shown in Table 10. Fig. 20-21 show the visualized experiment results for the two cases. It is worth noting that our experiments have considered all the four scenarios defined by COLREGs. Fig. 22 presents the profile of the USV during the simulations. From the corresponding results, the following conclusions are highlighted:

- The proposed path planning framework works well under dynamic environment. As shown in Fig. 20-21, the USV avoids all the moving obstacles in accordance with COLREG rules and adjusts its course autonomously to reach the destination safely.
- As denoted in Table 17 and Fig. 19, the planner ensures the relative distance to be sufficiently larger than the safety distance SD (15 m) and does not cause a potential collision risk. In Case 1, the minimum relative distance are 41.78 m and 38.29 m for DO1 and DO2 respectively, while in Case 2 the minimum distance is 37.31 m and 35.71 m for DO1 and DO2 respectively.
- As shown in Table 18, the transition path has successfully provided sufficient time for the replanning. In both cases, the transition routes allow more than 30-40 s for computing new trajectories, which is totally acceptable in practical situation since it usually takes less than 20 s for our planner to converge. This indicates that our strategy is able to soften the time restriction on the replanning process, which could also be used in combination with other algorithms.
- The proposed scheme can well fit the USV's mechanical system. As is shown in Fig. 22, we can clearly see that all guidance signals of surge and yaw can sufficiently satisfy compounded constraints which accommodate the admissibility and performability. The deviation between the course angle signal and reference is rather small, also, the change of the speed and thruster force are mild and smooth. This indicates that connection between replanning path and transition path is consistently continuous during the voyage, thereby contributing the excellent tracking performance.

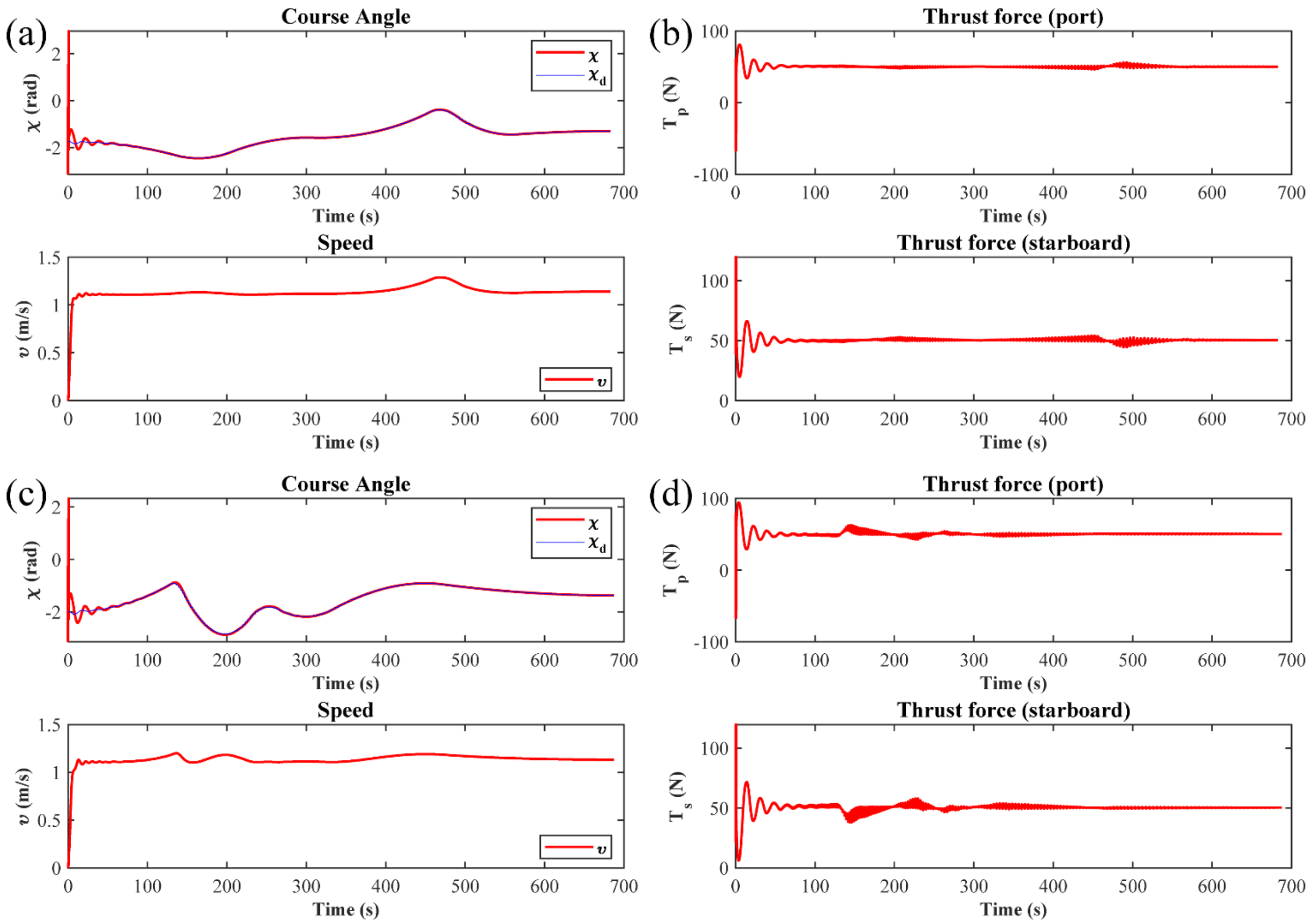


Fig. 22. Profile for (a) Course angle and speed in Case 1; (b) Thrust forces in Case 1; (c) Couse angle and speed in Case 2; (d) Thrust forces in Case 2

Some additional analysis:

- In Case 2 (See Table 9), the minimum distance caused by avoiding DO2 resulted in a relatively lower value (31.71 m) compared to other cases. By analyzing the behaviors of the vessels, this is typical heading-on situation. According to our previous description on the sensory vector, the free gap is very likely to lay on the S1 and S12 under such a situation, which results in a relatively lower β. This explains why the action range is smaller than other cases.
- As shown in Table 10, the time spends on the transition path of avoiding DO2 in Case 2 have shown the least value (29.2 s) compared to other cases. The reasons are twofold: first, the predicted collision location (CPA position) is rather close from the current position. In Algorithm 4, d_{md} depends on the distance between current position of USV and CPA position of DO. The lower d_{md} is, the shorter transition path will be. Second, as we mentioned from our previous analysis, such scenario causes a smaller action range, which also contributes to the shorter transition path.

5. Conclusion

In this paper, the path planning problem for USVs under dynamically unforeseen situations has been investigated and resolved. The formulated path planning problem successfully addresses four general objective functions subject to numerous constraints, the effects of currents, and presence of dynamic obstacles. The AENSGA-II is devised to address the problem, which can not only converge rapidly but also features strong global searching ability. Moreover, a linguistic satisfactory degree is designed based on fuzzy logic to re-evaluate the Pareto solutions, resulting in a more reasonable choice. A local collision avoidance strategy consisting of COLREG-compliant replanning mechanism and a transition path, which dynamically govern feasible actions of USVs under protocol constraints, interacts with unforeseen circumstances successfully. Based on the simulation and experiment results, it allows the conclusion that the proposed method can be regarded as a practical alternative for USV path planning.

Some limitations of the current study need to be addressed in the future work. First, this study only considers some basic rules in COLREGs. More strategies should be designed considering rule 16, rule 17, and velocity planning in the future study. Furthermore, some other effects of severe ocean environment loads are also prominent. It may be another potential topic for us to continuously inherit and develop the method with consideration of winds and waves. Finally, our algorithm appears to be practical theoretically but are not convincing in handling real-world situations due to the lack of experiments. We are planning to perform experimental verification on a real USV in the future work.

Acknowledgements

The authors wish to sincerely thank the Editor-in-Chief, the associate Editor, and the anonymous referees for their comments and suggestions.

References

Ahmed, F., Deb, K., 2013. Multi-objective optimal path planning using elitist non-dominated sorting genetic algorithms. Soft Comput. 17, 1283–1299. https://doi.org/10.1007/s00500-012-0964-8

Alam, S., Dobbie, G., Rehman, S.U., 2015. Analysis of particle swarm optimization based hierarchical data clustering approaches. Swarm Evol. Comput. 25, 36–51. https://doi.org/10.1016/j.swevo.2015.10.003

Beser, F., Yildirim, T., 2018. COLREGS Based Path Planning and Bearing Only Obstacle Avoidance for Autonomous Unmanned Surface Vehicles. Procedia Comput. Sci. 131, 633–640. https://doi.org/10.1016/j.procs.2018.04.306

Chang, H.-C., Hsu, Y.-L., Hung, S.-S., Ou, G.-R., Wu, J.-R., Hsu, C., 2021. Autonomous Water Quality Monitoring and Water Surface Cleaning for Unmanned Surface Vehicle. Sensors 21, 1102. https://doi.org/10.3390/s21041102

Chen, T.Y., Kuo, F.-C., Liu, H., 2009. Adaptive random testing based on distribution metrics. J. Syst. Softw. 82, 1419–1433. https://doi.org/10.1016/j.jss.2009.05.017

Cryer, S., Carvalho, F., Wood, T., Strong, J.A., Brown, P., Loucaides, S., Young, A., Sanders, R., Evans, C., 2020. Evaluating the Sensor-Equipped Autonomous Surface Vehicle C-Worker 4 as a Tool for Identifying Coastal Ocean Acidification and Changes in Carbonate Chemistry. J. Mar. Sci. Eng. 8, 939. https://doi.org/10.3390/jmse8110939

Davoodi, M., Panahi, F., Mohades, A., Hashemi, S.N., 2013. Multi-objective path planning in discrete space. Appl. Soft Comput. 13, 709–720. https://doi.org/10.1016/j.asoc.2012.07.023

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. 6, 182–197. https://doi.org/10.1109/4235.996017

Deng, W., Zhang, X., Zhou, Y., Liu, Y., Zhou, X., Chen, H., Zhao, H., 2022. An enhanced fast non-dominated solution sorting genetic algorithm for multi-objective problems. Inf. Sci. 585, 441–453. https://doi.org/10.1016/j.ins.2021.11.052

Dhanalakshmi, S., Kannan, S., Mahadevan, K., Baskar, S., 2011. Application of modified NSGA-II algorithm to Combined Economic and Emission Dispatch problem. Int. J. Electr. Power Energy Syst. 33, 992–1002. https://doi.org/10.1016/j.ijepes.2011.01.014

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 Eng. 216, 107693. https://doi.org/10.1016/j.oceaneng.2020.107693

Han, S., Wang, L., Wang, Y., He, H., 2022. A dynamically hybrid path planning for unmanned surface vehicles based on non-uniform Theta* and improved dynamic windows approach. Ocean Eng. 257, 111655. https://doi.org/10.1016/j.oceaneng.2022.111655

Hu, L., Naeem, W., Rajabally, E., Watson, G., Mills, T., Bhuiyan, Z., Raeburn, C., Salter, I., Pekcan, C., 2020. A Multiobjective Optimization Approach for COLREGs-Compliant Path Planning of Autonomous Surface Vehicles Verified on Networked Bridge Simulators. IEEE Trans. Intell. Transp. Syst. 21, 1167–1179. https://doi.org/10.1109/TITS.2019.2902927

Jiang, S., Zhang, J., Ong, Y.-S., Zhang, A.N., Tan, P.S., 2015. A Simple and Fast Hypervolume Indicator-Based Multiobjective Evolutionary Algorithm. IEEE Trans. Cybern. 45, 2202–2213. https://doi.org/10.1109/TCYB.2014.2367526

Kim, H., Kim, S.-H., Jeon, M., Kim, J., Song, S., Paik, K.-J., 2017. A study on path optimization method of an unmanned surface vehicle under environmental loads using genetic algorithm. Ocean Eng. 142, 616–624. https://doi.org/10.1016/j.oceaneng.2017.07.040

Krell, E., King, S.A., Garcia Carrillo, L.R., 2022. Autonomous Surface Vehicle energy-efficient and reward-based path planning using Particle Swarm Optimization and Visibility Graphs. Appl. Ocean Res. 122, 103125. https://doi.org/10.1016/j.apor.2022.103125

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. Appl. Ocean Res. 101, 102239. https://doi.org/10.1016/j.apor.2020.102239

Liu, J., Zhang, J., Yan, X., Soares, C.G., 2022. Multi-ship collision avoidance decision-making and coordination mechanism in Mixed Navigation Scenarios. Ocean Eng. 257, 111666. https://doi.org/10.1016/j.oceaneng.2022.111666

Liu, Y., Bucknall, R., Zhang, X., 2017. The fast marching method based intelligent navigation of an unmanned surface vehicle. Ocean Eng. 142, 363–376. https://doi.org/10.1016/j.oceaneng.2017.07.021

Lyridis, D.V., 2021. An improved ant colony optimization algorithm for unmanned surface vehicle local path planning with multi-modality constraints. Ocean Eng. 241, 109890. https://doi.org/10.1016/j.oceaneng.2021.109890

Ma, Y., Hu, M., Yan, X., 2018. Multi-objective path planning for unmanned surface vehicle with currents effects. ISA Trans. 75, 137–156. https://doi.org/10.1016/j.isatra.2018.02.003

Madeo, D., Pozzebon, A., Mocenni, C., Bertoni, D., 2020. A Low-Cost Unmanned Surface Vehicle for Pervasive Water Quality Monitoring. IEEE Trans. Instrum. Meas. 69, 1433–1444. https://doi.org/10.1109/TIM.2019.2963515

MahmoudZadeh, S., Abbasi, A., Yazdani, A., Wang, H., Liu, Y., 2022. Uninterrupted path planning system for Multi-USV sampling mission in a cluttered ocean environment. Ocean Eng. 254, 111328. https://doi.org/10.1016/j.oceaneng.2022.111328

Meng, J., Liu, Y., Bucknall, R., Guo, W., Ji, Z., 2022. Anisotropic GPMP2: A Fast Continuous-Time Gaussian

Processes Based Motion Planner for Unmanned Surface Vehicles in Environments With Ocean Currents. IEEE Trans. Autom. Sci. Eng. 19, 3914–3931. https://doi.org/10.1109/TASE.2021.3139163

Nazarahari, M., Khanmirza, E., Doostie, S., 2019. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm. Expert Syst. Appl. 115, 106–120. https://doi.org/10.1016/j.eswa.2018.08.008

Ning, J., Chen, H., Li, T., Li, W., Li, C., 2020. COLREGs-Compliant Unmanned Surface Vehicles Collision Avoidance Based on Multi-Objective Genetic Algorithm. IEEE Access 8, 190367–190377. https://doi.org/10.1109/ACCESS.2020.3030262

Ntakolia, C., Iakovidis, D.K., 2021. A swarm intelligence graph-based pathfinding algorithm (SIGPA) for multi-objective route planning. Comput. Oper. Res. 133, 105358. https://doi.org/10.1016/j.cor.2021.105358

Ntakolia, C., Lyridis, D.V., 2022. A comparative study on Ant Colony Optimization algorithm approaches for solving multi-objective path planning problems in case of unmanned surface vehicles. Ocean Eng. 255, 111418. https://doi.org/10.1016/j.oceaneng.2022.111418

Öztürk, Ü., Akdağ, M., Ayabakan, T., 2022. A review of path planning algorithms in maritime autonomous surface ships: Navigation safety perspective. Ocean Eng. 251, 111010. https://doi.org/10.1016/j.oceaneng.2022.111010

Sathiya, V., Chinnadurai, M., Ramabalan, S., 2022. Mobile robot path planning using fuzzy enhanced improved Multi-Objective particle swarm optimization (FIMOPSO). Expert Syst. Appl. 198, 116875. https://doi.org/10.1016/j.eswa.2022.116875

Shah, B.C., Gupta, S.K., 2020. Long-Distance Path Planning for Unmanned Surface Vehicles in Complex Marine Environment. IEEE J. Ocean. Eng. 45, 813–830. https://doi.org/10.1109/JOE.2019.2909508

Silva, J.A.R., Grassi, V., 2018. Clothoid-Based Global Path Planning for Autonomous Vehicles in Urban Scenarios, in: 2018 IEEE International Conference on Robotics and Automation (ICRA). Presented at the 2018 IEEE International Conference on Robotics and Automation (ICRA), IEEE, Brisbane, QLD, pp. 4312–4318. https://doi.org/10.1109/ICRA.2018.8461201

Smith, K.L., Sherman, A.D., McGill, P.R., Henthorn, R.G., Ferreira, J., Connolly, T.P., Huffard, C.L., 2021. Abyssal Benthic Rover, an autonomous vehicle for long-term monitoring of deep-ocean processes. Sci. Robot. 6, eabl4925. https://doi.org/10.1126/scirobotics.abl4925

Song, R., Liu, Y., Bucknall, R., 2019. Smoothed A* algorithm for practical unmanned surface vehicle path planning. Appl. Ocean Res. 83, 9–20. https://doi.org/10.1016/j.apor.2018.12.001

Tan, G., Zou, J., Zhuang, J., Wan, L., Sun, H., Sun, Z., 2020. Fast marching square method based intelligent navigation of the unmanned surface vehicle swarm in restricted waters. Appl. Ocean Res. 95, 102018. https://doi.org/10.1016/j.apor.2019.102018

Tanabe, R., Ishibuchi, H., 2019. A niching indicator-based multi-modal many-objective optimizer. Swarm Evol. Comput. 49, 134–146. https://doi.org/10.1016/j.swevo.2019.06.001

Vahid, S., Dideban, A., 2022. Optimal path planning for unmanned surface vehicle using new modified local search ant colony optimization. J. Mar. Sci. Technol. 27, 1207–1219. https://doi.org/10.1007/s00773-022-00898-2

Wang, F., Zhao, L., Bai, Y., 2022. Path Planning For Unmanned Surface Vehicles Based On Modified Artificial Fish Swarm Algorithm With Local Optimizer. Math. Probl. Eng. 2022, 1–15. https://doi.org/10.1155/2022/1283374

Wang, N., Xu, H., 2020. Dynamics-Constrained Global-Local Hybrid Path Planning of an Autonomous Surface Vehicle. IEEE Trans. Veh. Technol. 69, 6928–6942. https://doi.org/10.1109/TVT.2020.2991220

Wang, N., Zhang, Y., Ahn, C.K., Xu, Q., 2022. Autonomous Pilot of Unmanned Surface Vehicles: Bridging Path Planning and Tracking. IEEE Trans. Veh. Technol. 71, 2358–2374. https://doi.org/10.1109/TVT.2021.3136670

Wang, R., Ma, W., Tan, M., Wu, G., Wang, L., Gong, D., Xiong, J., 2021. Preference-inspired coevolutionary algorithm with active diversity strategy for multi-objective multi-modal optimization. Inf. Sci. 546, 1148–1165. https://doi.org/10.1016/j.ins.2020.09.075

Wang, X.D., Hirsch, C., Kang, Sh., Lacor, C., 2011. Multi-objective optimization of turbomachinery using improved NSGA-II and approximation model. Comput. Methods Appl. Mech. Eng. 200, 883–895.

https://doi.org/10.1016/j.cma.2010.11.014
Wang, Y., Yang, Z., Guo, Y., Zhu, J., Zhu, X., 2019. A novel multi-objective competitive swarm optimization algorithm for multi-modal multi objective problems, in: 2019 IEEE Congress on Evolutionary Computation (CEC). Presented at the 2019 IEEE Congress on Evolutionary Computation (CEC), IEEE, Wellington, New Zealand, pp. 271–278. https://doi.org/10.1109/CEC.2019.8790218
Xia, G., Han, Z., Zhao, B., Wang, X., 2020. Local Path Planning for Unmanned Surface Vehicle Collision Avoidance Based on Modified Quantum Particle Swarm Optimization. Complexity 2020, 1–15. https://doi.org/10.1155/2020/3095426
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 Eng. 184, 311–322. https://doi.org/10.1016/j.oceaneng.2019.04.055
Xue, H., 2022. A quasi-reflection based SC-PSO for ship path planning with grounding avoidance. Ocean Eng. 247, 110772. https://doi.org/10.1016/j.oceaneng.2022.110772
Yao, Y., Liang, X., Li, M., Yu, K., Chen, Z., Ni, C., Teng, Y., 2021. Path Planning Method Based on D* lite Algorithm for Unmanned Surface Vehicles in Complex Environments. China Ocean Eng. 35, 372–383. https://doi.org/10.1007/s13344-021-0034-z
Yu, J., Liu, G., Xu, J., Zhao, Z., Chen, Z., Yang, M., Wang, X., Bai, Y., 2022a. A Hybrid Multi-Target Path Planning Algorithm for Unmanned Cruise Ship in an Unknown Obstacle Environment. Sensors 22, 2429. https://doi.org/10.3390/s22072429
Yu, J., Yang, M., Zhao, Z., Wang, X., Bai, Y., Wu, J., Xu, J., 2022b. Path planning of unmanned surface vessel in an unknown environment based on improved D*Lite algorithm. Ocean Eng. 266, 112873. https://doi.org/10.1016/j.oceaneng.2022.112873
Yu, K., Liang, X., Li, M., Chen, Z., Yao, Y., Li, X., Zhao, Z., Teng, Y., 2021. USV path planning method with velocity variation and global optimisation based on AIS service platform. Ocean Eng. 236, 109560. https://doi.org/10.1016/j.oceaneng.2021.109560
Zhang, F., Ennasr, O., Litchman, E., Tan, X., 2016. Autonomous Sampling of Water Columns Using Gliding Robotic Fish: Algorithms and Harmful-Algae-Sampling Experiments. IEEE Syst. J. 10, 1271–1281. https://doi.org/10.1109/JSYST.2015.2458173
Zhao, L., Bai, Y., Wang, F., Bai, J., 2022a. Path planning for autonomous surface vessels based on improved artificial fish swarm algorithm: a further study. Ships Offshore Struct. 1–13. https://doi.org/10.1080/17445302.2022.2116765
Zhao, L., Mao, R., Bai, Y., 2022b. Observer Design for GNSS-aided Position Estimation of Autonomous Surface Vessels, in: 2022 IEEE International Conference on Unmanned Systems (ICUS). pp. 237–241. https://doi.org/10.1109/ICUS55513.2022.9987084
Zhao, L., Mao, R., Bai, Y., 2022c. Local Path Planning for Unmanned Surface Vehicles based on Hybrid A* and B-spline, in: 2022 IEEE International Conference on Unmanned Systems (ICUS). pp. 81–85. https://doi.org/10.1109/ICUS55513.2022.9986811
Zhao, L., Wang, F., Bai, Y., 2022d. Route planning for autonomous vessels based on improved artificial fish swarm algorithm. Ships Offshore Struct. 1–10. https://doi.org/10.1080/17445302.2022.2081423
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. Appl. Ocean Res. 111, 102658. https://doi.org/10.1016/j.apor.2021.102658
Zitzler, E., Laumanns, M., Thiele, L., 2001. SPEA2: Improving the strength pareto evolutionary algorithm (Report), TIK Report. ETH Zurich, Computer Engineering and Networks Laboratory, Zurich. https://doi.org/10.3929/ethz-a-004284029