

## Article

# Collision Avoidance Using Finite Control Set Model Predictive Control for Unmanned Surface Vehicle

Xiaojie Sun , Guofeng Wang \*, Yunsheng Fan , Dongdong Mu  and Bingbing Qiu 

School of Marine Electrical Engineering, Dalian Maritime University, Dalian 116026, China; xjsun.phd@gmail.com (X.S.); yunsheng@dlmu.edu.cn (Y.F.); ddmu.phd@gmail.com (D.M.); bbqiu.dmu@gmail.com (B.Q.)

\* Correspondence: gfwangsh@163.com; Tel.: +86-0411-8472-5623

Received: 21 April 2018; Accepted: 1 June 2018; Published: 4 June 2018



**Abstract:** In recent years, with the development of unmanned platforms, unmanned surface vehicles (USV) are attracting more and more attention. Compared to ordinary ships, USV have a smaller volume and faster speed, so their collision avoidance system (CAS) should have better responsiveness and stability. The paper describes a method that is based on finite control set model predictive control (FCS-MPC). A finite control set is generated by more practical control commands: the thruster speed and propulsion angle of the USV. The method is conceptually and computationally simple and yet quite versatile, as it can account for the dynamics of the USV, steering and propulsion system. Based on the theory of FCS-MPC, a safe and fast CAS is proposed, and it is verified in different static and dynamic environments. The real environment model for collision avoidance is established by extracting the environment data from the electronic chart. The result shows that the method is effective and can control the USV to sail safely and quickly in complex real scenarios with multiple dynamic obstacles.

**Keywords:** unmanned surface vehicle; collision avoidance; finite control set; model predictive control

## 1. Introduction

As a device for monitoring the marine environment, protecting marine rights and modern military weapons, the unmanned surface vehicle (USV) has a wide range of application prospects and has become a hot research topic for intelligent ships [1]. In the complex marine environment conditions, dynamically autonomous collision avoidance is the premise and foundation for the USV to navigate safely and successfully accomplish the assigned tasks [2]. It is also one of the key research contents of the automatic decision-making in USV intelligent control technology.

In recent years, some scholars's research interest in the development of USV has increased. Collision avoidance is one of the major areas in this field. With the development of research, more and more papers rely on the real environment. Yang et al. used the finite angle A-Star algorithm (FAA\*) to solve the path planning problem of USV in real satellite thermal images [3]. Kim et al. suggested a grid map-based on the Theta\* algorithm to create paths in real-time, considering both the yaw rate and course angle of USV [4]. Combining global path planning with local path planning, the shortcut Dijkstra algorithm and modified artificial potential field (APF) algorithm were used for the vessel's navigation in [5]. After, [6] improved the global path planning, that is the artificial potential field-ant colony optimization (APF-ACO) obstacle-avoidance algorithm. Based on the Voronoi diagram and Fermat's spiral, [7] produced the path in real time, which steered the vessel course by an indirect adaptive line-of-sight (LOS) guidance algorithm. Campbell integrated obstacle detection, risk assessment and path planning (avoidance) tasks. The detection system is based on a vision-LiDAR (light detection and ranging) system, and the Rule-based Repairing A-Star (R-RA\*) algorithm is adopted in path

planning [8]. Kuwata et al. [9] completed the task of safe maritime autonomous navigation by using the velocity obstacles (VO) method. The environmental information of the works in [3–6] was obtained from satellite images, and that of [8,9] from cameras or sonar. As previously mentioned, although many papers choose satellite images, some invisible information, such as reefs, ocean depth, and so on, does not exist. The observing ability of cameras and sonar will also be affected by light and shocks, resulting in their small observation range and low accuracy. Therefore, the paper is based on the environmental information from the electronic chart.

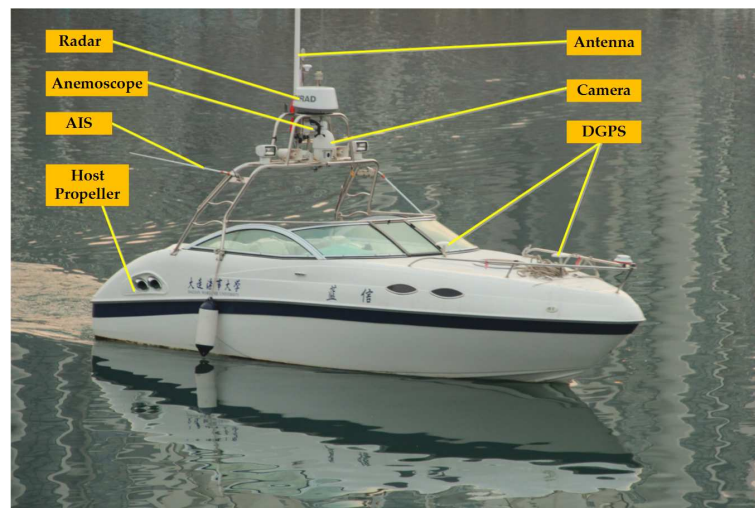
Generally, the collision avoidance system (CAS) requires the vessel to finish the task safely, stably and rapidly, i.e., attainability, safety, stability and rapidity [2]. For high speed USV, [10] proposed a local reaction avoidance method. However, we find that the realization of CAS is divided into path planning and the control system. If the control and planning are combined into an algorithm, the real-time performance of the algorithm will be greatly improved to meet the needs of collision avoidance for USV. Therefore, we need an approach that employs ideas from optimization-based control and that can take into account the motion characteristics of the vessel. Model predictive control (MPC) is increasingly being applied to vehicle navigation problems [11,12]. It has been combined with path planning to achieve online stability and robustness [13,14]. This was accomplished by the constant re-planning path process at each time, then applying the initial control related to the chosen trajectory to the vehicle [15]. In the literature on robust MPC, the concept of optimization of a finite control set is well known, e.g., [15,16]. The finite control set model predictive control (FCS-MPC) algorithm has the advantages of direct control and easy handling of system constraints [17–19]. In [18], the collision avoidance of a ship was realized with MPC. Compared with ordinary ships, USV have a smaller volume and faster speed, so their CAS should have better responsiveness and stability.

In summary, the contributions of this paper are that, firstly, the theory of FCS-MPC is introduced into the collision avoidance of the USV, and the combination of path planning and the control system is realized to improve the safety and rapidity of collision avoidance for the USV; secondly, combining the characteristics of USV and the theory of FCS-MPC, a safe and fast CAS is proposed; thirdly, it is more practical as the output of CAS is the thruster speed and propulsion angle, instead of force and torque; finally, the CAS is verified in different static and dynamic environments, and the real environment model of collision avoidance is established from the electronic chart.

The rest of the paper is organized as follows. Section 2 introduces the hierarchical structure and technology of the USV system. Section 3 establishes the mathematical model of the USV and applies the FCS-MPC method to CAS. Section 4 is the simulation result verification. Finally, Section 5 concludes the paper, summing up comments and future works about the CAS of USV.

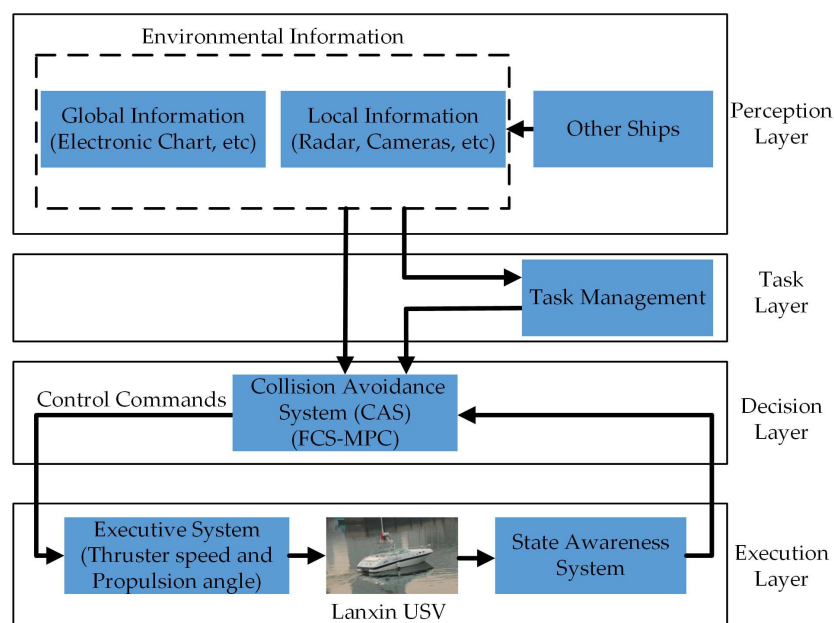
## 2. The Overview of the System Structure and Technology

The Lanxin USV is a small unmanned mobile platform above water of Dalian Maritime University, which has many functions such as maritime rescue, marine monitoring, water sampling, and so on [20]. As a new type of intelligent marine equipment, it is equipped with many types of equipment, such as radar, camera, differential global positioning system (DGPS) navigation sensor, automatic identification system (AIS), etc. The radar, DGPS and AIS were provided by Simrad; the camera is by Hikvision; and the supplier of the host propeller was Mercury Marine. The information of these devices will be transmitted to STM32 (32-bit microcontroller integrated circuits by STMicroelectronics) and a computer, in which the CAS is implemented. The Lanxin USV is shown in Figure 1.



**Figure 1.** The Lanxin USV. AIS, automatic identification system.

In the paper, the Lanxin USV is used as the research object. Figure 2 describes the hierarchical structure of the USV system. The structure consists of four layers, which are the perception layer, task layer, decision layer and execution layer.



**Figure 2.** The hierarchical structure of the Lanxin USV system. FCS, finite control set.

The perception layer acquires environmental information including global and local information from the electronic chart and radar. The task layer arranges the mission of the USV based on the requirements. In general, a task can be defined as a set of path points in environmental information. According to the task and environmental information, the third layer, that is the decision layer, updates the appropriate control commands to complete the mission by FCS-MPC. The generated control commands will be passed to the execution layer, to calculate the specific trajectory for the USV. In order to minimize system error, the real-time state information (velocity, position) is fed back to the decision layer to modify the control commands by the state awareness system. The structure is a closed loop system to ensure the safety of the USV.

From the whole structure of the USV to analyze, the decision layer is very important, and it serves as a link between the other three layers. In the decision layer, the most important task of CAS is to avoid collisions. In addition to the collision avoidance task, the CAS also needs to reach the required target point according to the mission. There are multiple objectives that need to be optimized, such as distance, speed, etc. The ultimate goal is to make the USV accomplish tasks as quickly and safely as possible.

Although many research works have also been carried out on collision avoidance of some unmanned platforms (unmanned ground vehicle [21,22], unmanned aerial vehicle [23], autonomous robot [24], etc.), in contrast, only a few articles have studied USV. This is possibly due to the relatively complex ocean environment and the robustness of the USV control system. Therefore, the challenge of CAS research is to propose a practical collision avoidance algorithm for USV in the real marine environment.

### 3. Collision Avoidance Based on the Finite Control Set Model Predictive Control Method

#### 3.1. Mathematical Model of USV Plane Motion

The mathematical model of the Lanxin USV plane motion is described by the nonlinear equations of motion for a three-degree of freedom (3-DOF) planar model about a surface vessel, where the motions in heave, roll and pitch are neglected [25]. The kinematics of the vessel can be written as:

$$\begin{cases} \dot{x} = u \cos \psi - v \sin \psi \\ \dot{y} = u \sin \psi + v \cos \psi \\ \dot{\psi} = r \end{cases} \quad (1)$$

where  $(x, y)$  denote the position of the center of mass,  $\psi$  is the course angle of the vessel in the inertial reference frame,  $(u, v)$  are the surge and sway velocities, respectively, and  $r$  denotes the yaw rate of the vessel in the body-fixed frame.

Assume that:

- The sea environment is calm, so the disturbances caused by environmental factors such as wind, wave and current are neglected;
- The USV is symmetrical, and its barycenter coincides with the geometric center, that is the inertia, added mass and damping matrices are diagonal.

In the process of collision avoidance, the propulsion angle  $\delta$  sometimes requires a larger value, so the influence of sway thrust  $\tau_v$  should be considered [26]. The simplified dynamic model of the surface vessel is used to describe [27,28]:

$$\begin{cases} \dot{u} = \frac{m_{22}}{m_{11}}vr - \frac{d_{11}}{m_{11}}u + \frac{1}{m_{11}}\tau_u \\ \dot{v} = -\frac{m_{11}}{m_{22}}ur - \frac{d_{22}}{m_{22}}v + \frac{1}{m_{22}}\tau_v \\ \dot{r} = \frac{m_{11}-m_{22}}{m_{33}}uv - \frac{d_{33}}{m_{33}}r + \frac{1}{m_{33}}\tau_r \end{cases} \quad (2)$$

where  $m_{ii}$  are given by the vessel inertia and the added mass effects,  $d_{ii}$  are given by the hydrodynamic damping and  $(\tau_u, \tau_v, \tau_r)$  are the surge thrust, sway thrust and yaw moment.

Additionally [26,29]:

$$\begin{cases} \tau_u = T \cos(\delta) \\ \tau_v = T \sin(\delta) \\ \tau_r = T \sin(\delta)L/2 \end{cases} \quad (3)$$

$$T = (1 - t_p) \rho n^2 D_p^4 K_T$$

where  $\delta$  is the propulsion angle,  $T$  is the thrust produced by the thruster and  $L$  is the vessel length. In the calculation of thrust,  $t_p$ ,  $\rho$ ,  $n$ ,  $D_p$  and  $K_T$  are the reduction coefficient, sea water density, thruster speed, propeller diameter and thrust coefficient, and other quantities except thruster speed  $n$  are constant. Therefore, the  $\delta$  and  $n$  are the inputs of the model.

In general, the response model of the steering process is considered as a first-order inertia link [29]. However, the thruster steering of the surface vessel is completed by the stern control system with high sensitivity and quick action. After analyzing the actual data collected from the experimental process, the steering gear response model can be regarded as a second-order system [30].

$$\ddot{\delta} + 2\zeta\omega_n\dot{\delta} + \omega_n^2\delta = K\omega_n^2\delta_r \quad (4)$$

Here,  $(\omega_n, \zeta, K)$  are the natural frequency, damping ratio and proportionality coefficient of the second-order system,  $\delta_r$  is the command propulsion angle and  $|\delta| \leq 30^\circ$  is the propulsion angle range.

### 3.2. Finite Control Set Model Predictive Control of CAS

The FCS-MPC is an optimal control algorithm [19]. Based on the USV input function combination (control set) being limited, through the prediction model, the output of the CAS is calculated by using the traversal method in a control cycle, and the combination of the input function that makes the designed evaluation function optimal is applied to the actuator of USV.

Figure 3 shows the principle of the CAS based on FCS-MPC. The USV prediction model  $f_p\{X, S\}$  is constructed according to the relationship between the state quantity  $X$  and input the function combination  $S$  of USV. At time  $t_k$ , use the state quantity  $X(t_k)$  and the  $i$ -th input function combination  $S_i$  to calculate the predicted value of the state quantity  $X_{pi}(t_{k+j})$  at the next prediction horizon under the action of  $S_i$ , that is  $X_{pi}(t_{k+j}) = f_p\{X(t_k), S_i\}$ ,  $i = 1, 2, \dots, c$ , where  $c$  is the number of all valid input function combinations for the USV,  $j = 1, 2, \dots, t_p$ , and  $t_p$  is the prediction horizon. The subscript  $pi$  of the variable symbol refers to the predicted value under the action of the  $S_i$ . According to the requirements of CAS, i.e., attainability, safety, stability and rapidity, the evaluation function  $f_e\{X_{pi}, S_i\}$  of the system is constructed. Finally, the control behavior combination  $S_o$  ( $o \in \{1, \dots, c\}$ ) that makes the function  $f_e$  optimal is applied to the vessel.

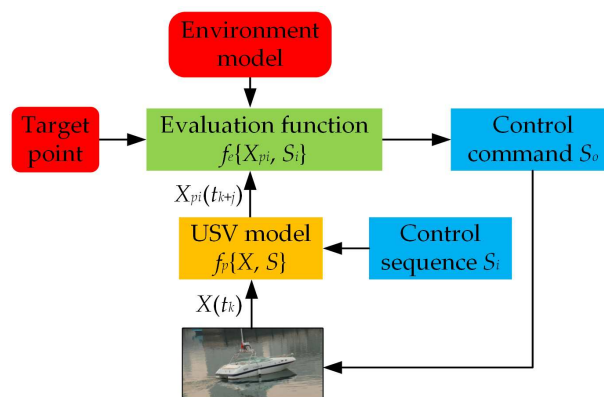


Figure 3. The CAS based on FCS-MPC.

### 3.3. Control Behaviors and Prediction of USV Trajectory

With the FCS-MPC method, the optimal control behavior can be obtained under the constraints of the environment and the dynamics. In order to predict vessel motion under different control behaviors, the 3-DOF planar dynamics model  $f_p\{X, S\}$  of the USV is adopted, and the state quantity  $X$  of the USV is:

$$X = [x, y, \psi, u, v, r]^T. \quad (5)$$

In order to improve the system performance, the finite control set  $S$  should be as wide as possible in the computing time. In the USV system, the thruster speed  $n$  and propulsion angle  $\delta_r$  form the finite control set  $S$ . The performance of the actuator of the USV is limited, such as the maximum propulsion angle  $\delta_m$ , the maximum thruster speed  $n_m$  and the change rate  $\dot{\delta}$  and  $\dot{n}$  of the propulsion angle and thruster speed. At time  $t_k$ ,  $\delta$  and  $\dot{n}$  are used to calculate the range of finite control set  $S$  within discrete time  $\Delta t$  as:

$$S_m = \{\delta_r \in [-\delta_m, \delta_m], n \in [0, n_m]\} \\ S = \left\{ (\delta_r, n) \left| \begin{array}{l} \delta_r \in [\delta - \dot{\delta}\Delta t, \delta + \dot{\delta}\Delta t] \\ n \in [n - \dot{n}\Delta t, n + \dot{n}\Delta t] \end{array} \right. \right\}. \quad (6)$$

The maximum number of control input combinations is  $c = (2\dot{\delta}\Delta t/\delta_d) \times (2\dot{n}\Delta t/n_d)$ , where  $\delta_d$  and  $n_d$  are the discrete propulsion angle and thruster speed. That is to say, the CAS has the most  $c$  kinds of effective control behaviors within discrete time  $\Delta t$ .

When reducing the value of  $\delta_d$  and  $n_d$ , the control behavior will be more detailed, but this will cause too much computation. Therefore, there is a balance between the discrete number of control behavior and the discretization of high fidelity time, prediction horizon, ship model complexity, control update interval and computation waiting time. In addition, a very useful feature of these control behaviors is that they represent meaningful actions when the control behavior remains constant throughout the prediction horizon.

### 3.4. Evaluation Function

According to the four requirements (attainability, safety, stability and rapidity) of the CAS, the paper proposes the evaluation function  $f_{ei}(X_{pi}, S_i)$  of CAS, where some sub-functions, i.e.,  $f_1(X_{pi}, S_i)$ ,  $f_2(X_{pi}, S_i)$ ,  $f_3(X_{pi}, S_i)$  and  $f_4(X_{pi}, S_i)$ , correspond to the requirements, respectively. Considering that the independent variables in evaluation function  $f_e$  are discrete variables, this situation will lead to a discontinuous evaluation function, and some sub-functions in the evaluation function are dominant. Hence, the normalized method is used to eliminate this problem, e.g., the first normalized sub-function is:

$$\bar{f}_1(X_{pi}, S_i) = \frac{f_1(X_{pi}, S_i)}{\sum_{i=1}^c f_1(X_{pi}, S_i)}. \quad (7)$$

Moreover, with the importance of the CAS requirements, different weights  $(\alpha, \beta, \gamma, \lambda)$  are assigned to each evaluation sub-function. The  $i$ -th evaluation function is:

$$f_{ei}(X_{pi}, S_i) = \alpha \cdot \bar{f}_1(X_{pi}, S_i) + \beta \cdot \bar{f}_2(X_{pi}, S_i) + \gamma \cdot \bar{f}_3(X_{pi}, S_i) + \lambda \cdot \bar{f}_4(X_{pi}, S_i). \quad (8)$$

CAS chooses the control command  $S_o$ , which can obtain the maximum value of the evaluation function. The sub-functions are defined as follows:

- Attainability Sub-function  $f_1(X_{pi}, S_i)$

The first requirement of CAS is to complete the task, i.e., one or more target points from the task layer. If the course keeps pointing to the target point, which is called the navigation angle, the vessel will eventually arrive at the target point with the shortest distance. Thus, we can use the difference angle  $\theta_{pi}$  between the navigation angle and the predicted course  $\psi_{pi}$  to evaluate the attainability of  $X_{pi}$ , which is illustrated in Figure 4.



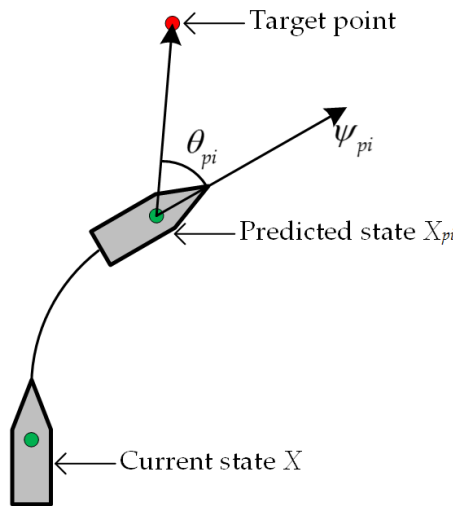


Figure 4. Description attainability sub-function.

However, with the increase of difference angle  $\theta_{pi}$ , the attainability will be reduced. This is contrary to the optimal value of the evaluation function, that is the  $\theta_{pi}$  can only describe the attainability indirectly. The attainability sub-function is:

$$f_1(X_{pi}, S_i) = \pi - \theta_{pi}. \quad (9)$$

- Safety sub-function  $f_2(X_{pi}, S_i)$ :

In order to ensure the safety of the system, the vessel runs away from all obstacles in the surrounding environment as far as possible. The distance between the vessel and the nearest obstacle can describe the security of the USV. The safety sub-function  $f_2(X_{pi}, S_i)$  is:

$$do_i^k = |X_{pi} - Xo_k|$$

$$f_2(X_{pi}, S_i) = \min_{k=1}^z \{do_i^k\} \quad (10)$$

where  $do_i^k$  is the distance between predicted USV state  $X_{pi}$  and the  $k$ -th obstruction  $Xo_k$ ,  $z$  is the number of obstructions within  $do_i^k \leq d_r$  and  $d_r$  is the distance range that takes security into account. If  $do_i^k > d_r$ ,  $do_i^k$  is considered equal to  $d_r$ .

Moreover, considering the motion characteristics of the vessel, stopping distance  $d_{stop}$  should also be considered, and it is the process of slowing down from the current speed  $V$  to stopping according to the acceleration  $\dot{V}$ . The distance is calculated by:

$$d_{stop}^i = V_{pi}^2 / (2 \cdot \dot{V}_{pi}). \quad (11)$$

Therefore, when  $d_{stop}^i > f_2(X_{pi}, S_i)$ , the  $i$ -th control behavior  $S_i$  can be excluded.

- Stability sub-function  $f_3(X_{pi}, S_i)$ :

Because the instability of the vessel is mainly caused by the sharp alteration of course, it is necessary to avoid the occurrence of a large propulsion angle  $\delta$ , that is the stability sub-function is:

$$f_3(X_{pi}, S_i) = \delta_m - |\delta_i|. \quad (12)$$

- Rapidity sub-function  $f_4(X_{pi}, S_i)$

The speed  $V$  of the vessel can be used to describe the rapidity, that is:

$$f_4(X_{pi}, S_i) = V_{pi}. \quad (13)$$

According to the importance of each sub-function, we adjust the weight value and select the optimal control command, so that the vessel can accomplish tasks safely, stably and quickly.

#### 4. Simulation Study and Results

In order to verify the feasibility of the proposed CAS, simulation experiments are carried out in two different environments. The first simulation experiment is carried out in a complex environment and tested in a static and a dynamic environment, respectively. The second experiment is based on the establishment of the real environment model with multiple dynamic obstacles. Table 1 is the parameter information of the USV and CAS. The dynamic obstacles are obtained by environmental awareness equipment, such as radar, etc., and the detection area is set to a circle with the USV as the center and a radius of 250 m.

All of the simulation results are presented by time-dependent motion sequences. The algorithm has been coded in MATLAB 2010(b), and simulations were run on a computer with a Core i7 3.6 GHz processor and 4 GB of RAM.

**Table 1.** Parameter information about the USV and CAS.

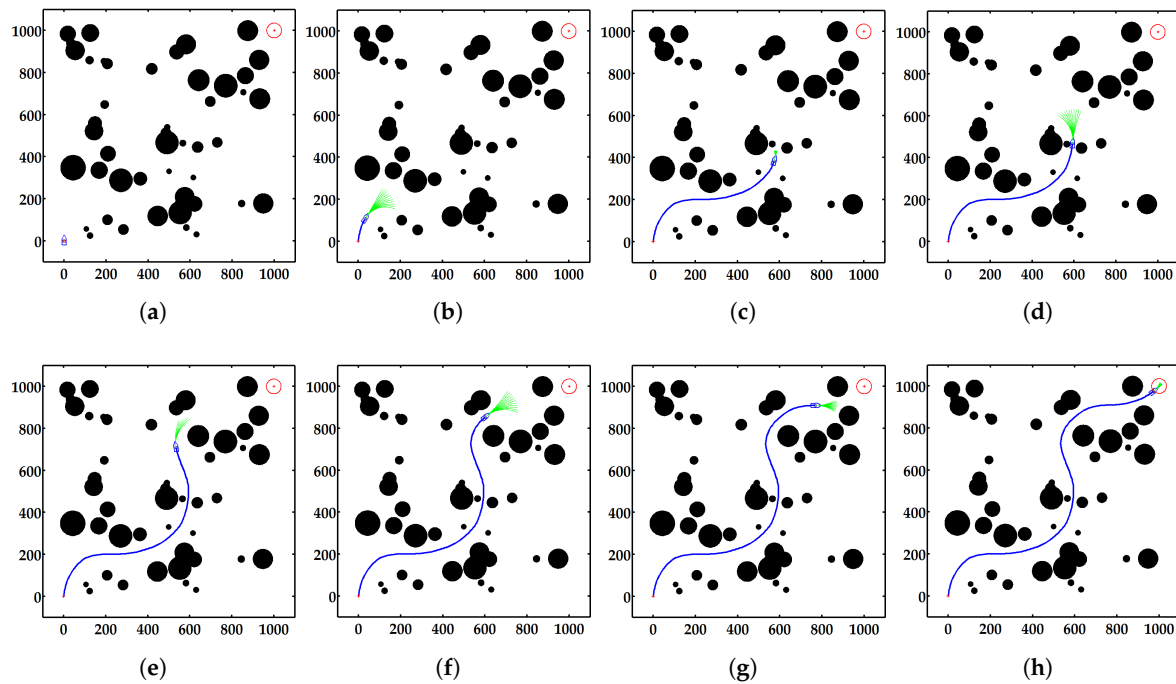
	Parameter	Value	Units
USV	Initial speed of USV	0.0 (0.0)	m/s (kn)
	Initial thruster speed	0.0	r/min
	Maximum thruster speed	66.7 (4000)	r/s (r/min)
	Change rate of thruster speed	6.7	r/s <sup>2</sup>
	Discrete thruster speed	1.7 (100)	r/s (r/min)
	Initial propulsion angle	0.0	°
	Maximum propulsion angle	30.0	°
	Change rate of propulsion angle	5.0	°/s
CAS	Discrete propulsion angle	1.0	°
	Detection area	250.0	m
	Distance range	200.0	m
	Prediction horizon	15.0	s
	Control update interval	1.0	s
	Weight of evaluation function	(0.70, 1.50, 1.50, 0.50)	–

##### 4.1. Collision Avoidance in a Complex Environment

In this test, in order to simulate an area having enough space for the USV to avoid collisions, the dimensions of the area constructed are 1100 m × 1100 m. In Figure 5, the static obstacles are 50 randomly generated circular obstacles, the radius of which is from 10 m to 60 m. The static environment is more complex than practice. The starting point for the USV is marked as a red '+' marker. Considering the fact that it is unreasonable to set the end point of the USV as a point, the end area is set to a red circular region with a radius of 35 m.

In Figure 5, the whole process of the USV from the starting point to the ending area is guided by the FCS-MPC algorithm according to the dynamics of the USV. Figure 5a,b describes the starting process of the USV. In Figure 5c,d, the USV reduces its speed by taking into account the risk of the region to be passed and accelerates after passing. In Figure 5e,f, in order to achieve reaching of the ending area, the yaw rate of the USV reaches a larger value, and the speed will decrease accordingly due to the influence of motion characteristics. Figure 5g,h describes the process of the USV arriving at the ending area.





**Figure 5.** Demonstration diagram of the USV in a static environment. (a) Time 1 s; (b) Time 18 s; (c) Time 83 s; (d) Time 93 s; (e) Time 113 s; (f) Time 126 s; (g) Time 143 s; (h) Time 168 s.

On the basis of static environment simulation, two dynamic obstacles (circles with a radius of 20 m) are added. The information of dynamic obstacles is as shown in Table 2, and the values of other parameters are consistent with the static environment.

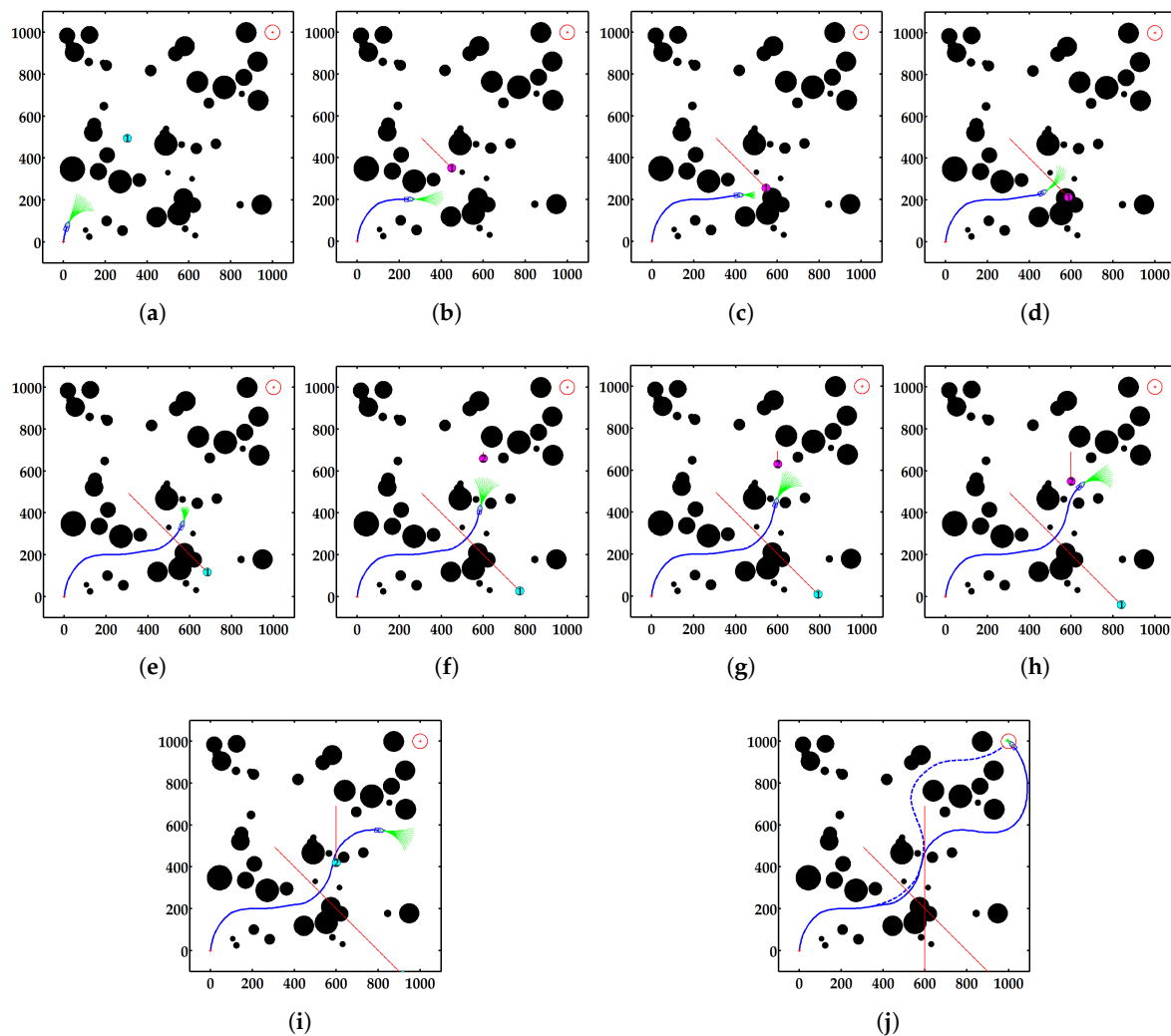
**Table 2.** The information of the dynamic obstacles.

	Start Point (m)	Start Time (s)	Velocity (m/s)	Speed (kn)
Dynamic Obstacle 1	(300, 500)	15	(6, −6)	16.5
Dynamic Obstacle 2	(600, 700)	40	(0, −10)	19.4

In Figure 6, the Number 1 and 2 cyan circles represent Dynamic Obstacles 1 and 2. Dynamic Obstacle 1 enters the detection area and becomes magenta at 39 s. When the time is 55 s, Dynamic Obstacle 1 and the USV form a crossing situation, forcing the USV to decelerate. The USV completely avoids Dynamic Obstacle 1 at 62 s. After 78 s, Dynamic Obstacle 1 leaves the detection area. At 93 s, Dynamic Obstacle 2 is detected by CAS in the detection area. Dynamic Obstacle 2 and USV form a head-on situation. During the time period 93–104 s, USV avoids Dynamic Obstacle 2, and at 174 s, it successfully reaches the ending area. In Figure 6j, the blue solid curve is the track in the dynamic environment, and the blue dashed curve is in the static environment.

Table 3 and Figure 7 display the evaluation index and compare the graph of the main parameters between static and dynamic environments. Figure 7a–d shows the control input (thruster speed and command propulsion angle) and state output (the speed and course) of the USV during the whole process. In the two tests, the USV and thruster should adjust their speed according to the risk level under the premise of keeping the maximum speed during sailing time. In addition, the maximum size of the command propulsion angle is  $16^\circ$  in static and dynamic environments, which is far less than  $30^\circ$ , so the system is stable. Figure 7e shows the computing time of CAS, and the maximum computation time is 0.22 s, far less than the update time of 1.0 s. Figure 7f displays the distance between obstacles and the USV in the distance range. In the static environment, the minimum and average of the distance

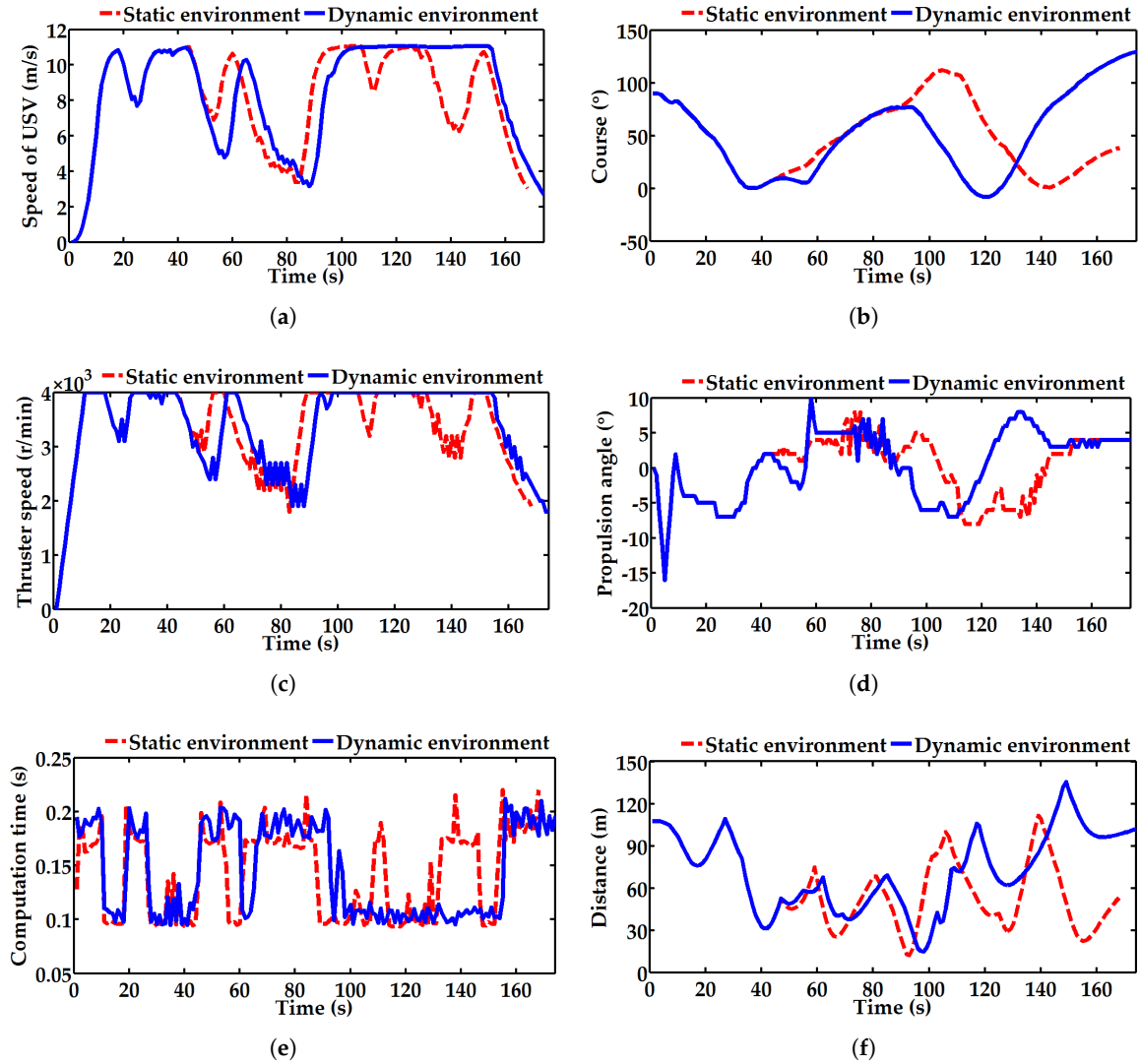
are 12.2 m and 60.2 m, and they are 14.9 m and 72.8 m in the dynamic environment. It can be seen that the system is safe in the two tests.



**Figure 6.** Demonstration diagram of the USV in the dynamic environment. (a) Time 15 s; (b) Time 39 s; (c) Time 55 s; (d) Time 62 s; (e) Time 78 s; (f) Time 93 s; (g) Time 96 s; (h) Time 104 s; (i) Time 117 s; (j) Time 174 s.

**Table 3.** Results of the evaluation index comprised of two tests in the simulation environment.

Environment	Index	Speed of USV (m/s)	Thruster Speed (r/min)	Propulsion Angle (°)	Computation Time (s)	Distance (m)
static	maximum	11.1	4000	8.0	0.22	111.5
	minimum	0.0	0	−16.0	0.09	12.2
	average	8.3	3375	−0.5	0.14	60.2
dynamic	maximum	11.1	4000	10.0	0.21	135.5
	minimum	0.0	0	−16.0	0.09	14.9
	average	8.4	3389	0.4	0.14	72.8



**Figure 7.** Evaluation results in the simulation environment. (a) Speed of the USV; (b) course of the USV; (c) thruster speed; (d) command propulsion angle; (e) computation time; (f) distance between obstacles and the USV.

## 4.2. Collision Avoidance in a Real Environment

### 4.2.1. Environmental Assumptions

The improved GoodWin ship domain model by Davis expands the USV into a circle with a radius  $R_{USV}$  of half ship length [31]. According to Equation (14), the radius  $R_{Obst}$  of the obstacle is further expanded to  $RR_{UO}$ .

$$RR_{UO} = R_{USV} + R_{Obst} \quad (14)$$

In this way, the USV can be regarded as a particle, as shown in Figure 8. Therefore, the collision avoidance problem of the USV is converted to the expanded obstacle avoidance of the particle.

In addition, the height information in the environment is removed, considering only the two-dimensional (2D) plane environment. Therefore, the motion of the USV can be regarded as the particle motion in the plane environment.

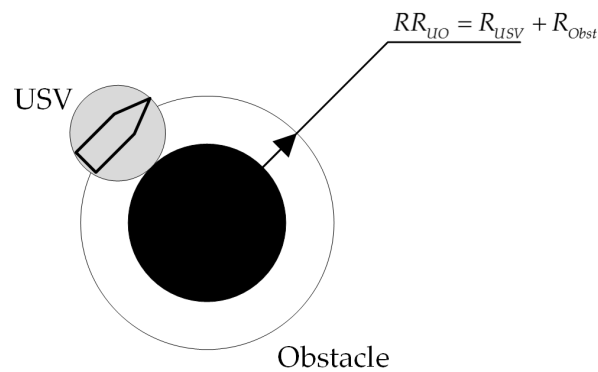


Figure 8. The expanded obstacle.

#### 4.2.2. Actual Marine Environment Model

The outer sea area of the harbor basin at Dalian Maritime University is selected as the marine environment modeling area. The marine environment in the electronic chart is shown in Figure 9. The electronic chart is provided by the Electronic Chart Display and Information System (ECDIS) of Dalian Maritime University.



Figure 9. The marine environment in the electronic chart.

There are islands and coastlines in the electronic chart. It is very necessary to deal with the coastline. To determine whether there is convex areas in the coastline, these areas are considered as islands. After excluding the convex region, the coastline is fitted as a line segment, and the farthest point of the coastline is calculated. The circumcircle of the farthest point and two ends of the fitted line segment is the approximate circle of the coastline, as shown in Figure 10.

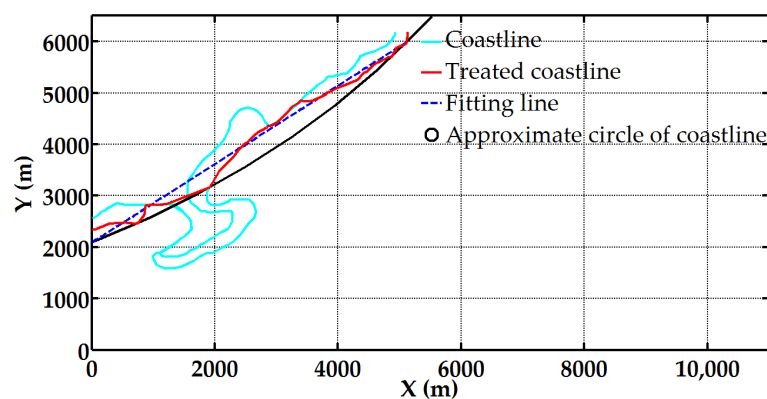


Figure 10. The coastline model after treatment.

These islands can be approximated as circles, for which the center point  $(r_x, r_y)$  and radius  $r_0$  are calculated by:

$$\begin{cases} r_x = r_0 + x_{\min} \\ r_y = r_0 + y_{\min} \end{cases}, \quad (15)$$

$$r_0 = \sqrt{[(x_{\max} - x_{\min})/2]^2 + [(y_{\max} - y_{\min})/2]^2},$$

where  $x_{\max}$ ,  $x_{\min}$  and  $y_{\max}$ ,  $y_{\min}$  are the maximum and minimum values of the islands on the  $x$  and  $y$  axes of the image coordinate system, respectively.

The simulated marine environment is shown in Figure 11.

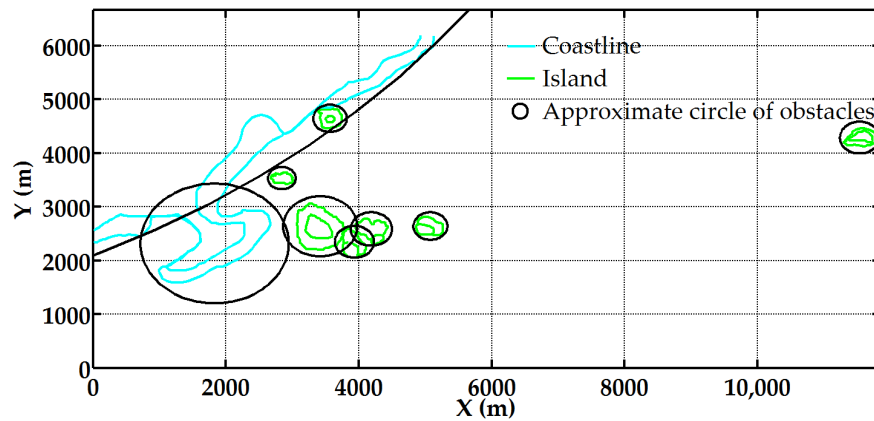


Figure 11. The expanded obstacle.

Now, the environment model is completed, which provides the foundation for the CAS in the real environment.

#### 4.2.3. Simulation Verification

Due to open water not being suitable to verify the effectiveness of the algorithm, in order to make the environment more complex, we reduce the map by five times. In Figure 12, the static obstacles include islands and coastlines, which expand as circular obstacles required by the algorithm. The starting and ending (goal) points for USV are marked as red 'S' and 'G' markers. In order to better simulate the real situation, the starting point and ending area are on the coastline. With the same algorithm parameters of the first simulation experiment, the USV starts sailing in the static sea area and safely enters the ending area. Figure 12 is the final result.

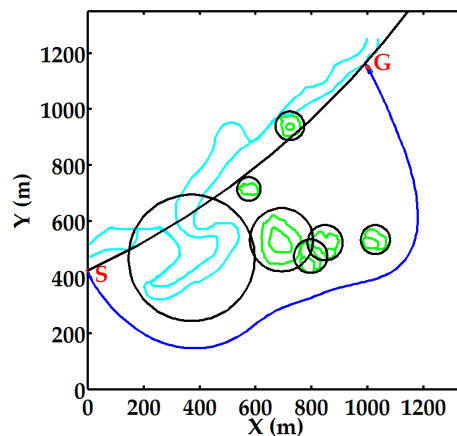


Figure 12. The simulation final result in the static sea environment.

Then, five dynamic obstacles are added into the environment, and their information is in Table 4. Compared to the simulation environment, the real environment is relatively simple. Therefore, we increase the number of dynamic obstacles with different velocities.

**Table 4.** The information of dynamic obstacles.

	Start Point (m)	Start Time (s)	Velocity (m/s)	Speed (kn)
Dynamic Obstacle 1	(600, 100)	20	(−10, 0)	19.4
Dynamic Obstacle 2	(550, 250)	40	(−5, −5)	13.7
Dynamic Obstacle 3	(1000, 50)	50	(−3, −10)	20.3
Dynamic Obstacle 4	(1150, 700)	100	(−1, 3)	6.1
Dynamic Obstacle 5	(1050, 1100)	130	(5, −5)	13.7

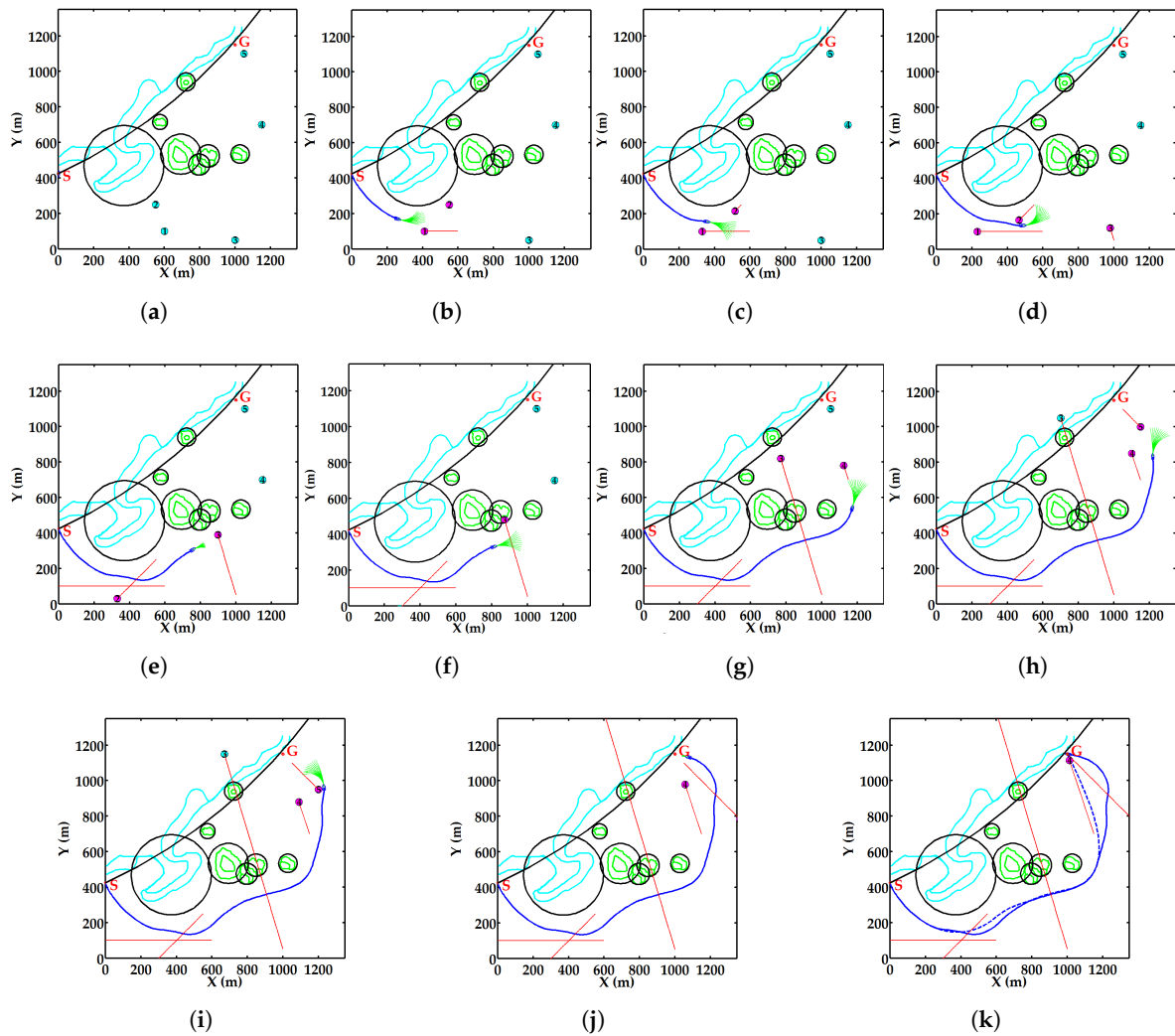
In Figure 13, from 38 s–56 s, the USV continuously evades Dynamic Obstacles 1 and 2. At 83 s, by reducing its speed, the USV avoids collision with Dynamic Obstacle 3. When the time is 126 s, because the speed of the USV is far greater than Dynamic Obstacle 4, they form the overtaking situation, and the CAS takes the option of steering without slowing down. Then, at 149 s, Dynamic Obstacle 5 and USV form the head-on situation, and the USV continues to alter about its course and avoids the obstacle. After 159 s, the USV avoids all obstacles, and at 237 s, it successfully reaches the ending area. Figure 13k shows the comparison of the final trajectories in the two environments.

The evaluation index and compared graph of the main parameters are shown in Table 5 and Figure 14. In the two environments, the evaluation index is almost the same, except for the propulsion angle and distance in Table 5. In the two environments, before encountering dynamic obstacles, the speeds of the USV and thruster are the same, and they can maintain a high speed, as in Figure 14a,c. From Figure 14d, the propulsion angles are also the same before collision avoidance, and as the ending area is in the rear side of the USV initial state, the propulsion angle reaches the maximum value of 20°, which is the adjustment process of the algorithm. During the collision avoidance, the size of the command propulsion angle is less than 10°, and the static environment is less than 7°. Figure 14e displays the computing time of CAS. The maximum computing time is 0.21 s, which is far less than the update time of 1 s. Figure 14f displays the distance between the obstacles and USV in the distance range. Because the starting point and ending area are on the edge of the obstacles, the distance is smaller at the start and end of the simulation. When calculating the distance index, we exclude the start and end stage, that is 1–18 s and 150–217 s in the static environment and 1–18 s and 172–237 s in the static environment. During the collision avoidance, the minimum distances from the five dynamic obstacles are respectively 37.4 m, 19.2 m, 134.7 m, 99.4 m and 16.7 m, and sufficient security distance is guaranteed. In a real environment, the USV can avoid coastlines, islands and dynamic obstructions. This result is consistent with the simulation environment. The algorithm can guarantee the fast, stable and safe navigation of the USV in real time.

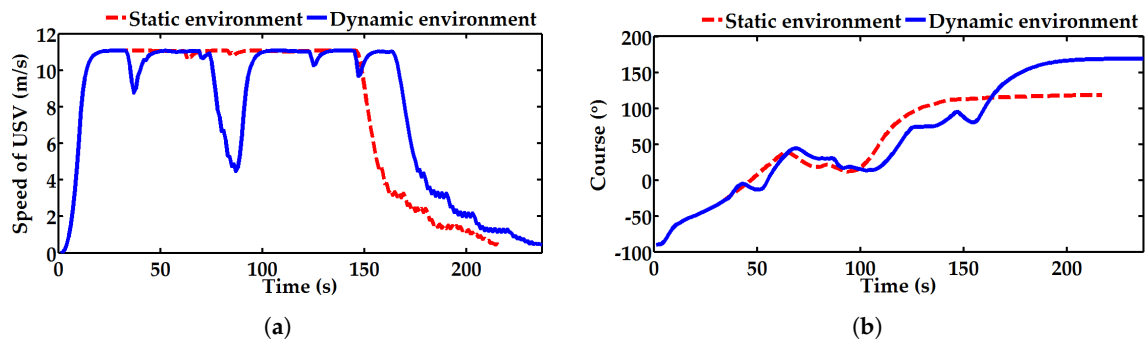
**Table 5.** Results of the evaluation index comprised of two tests in a real environment.

Environment	Index	Speed of USV (m/s)	Thruster Speed (r/min)	Propulsion Angle (°)	Computation Time (s)	Distance (m)
static	maximum	11.1	4000	20.0	0.21	200.0
	minimum	0.0	0	−4.0	0.09	75.1
	average	7.9	3153	1.7	0.13	116.3
dynamic	maximum	11.1	4000	20.0	0.21	172.4
	minimum	0.0	0	−10.0	0.09	16.7
	average	7.7	3117	2.1	0.14	97.6

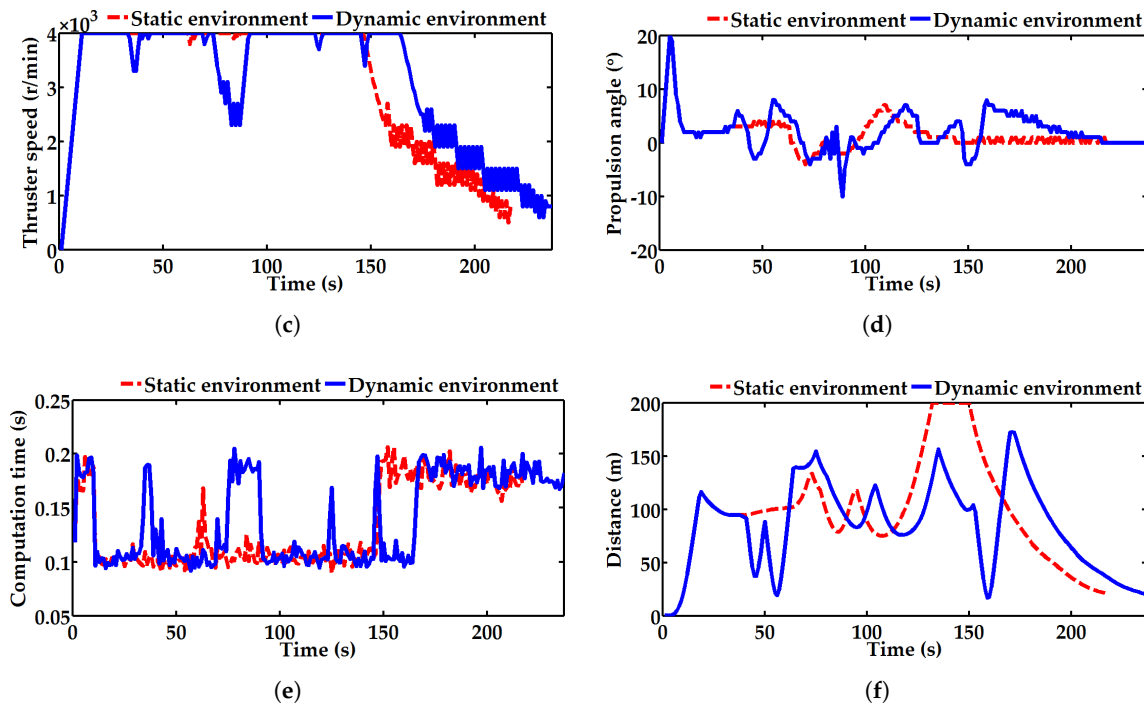




**Figure 13.** Demonstration diagram of the USV in a dynamic real environment. (a) Time 1 s; (b) Time 38 s; (c) Time 46 s; (d) Time 56 s; (e) Time 83 s; (f) Time 92 s; (g) Time 126 s; (h) Time 149 s; (i) Time 159 s; (j) Time 192 s; (k) Time 237 s.



**Figure 14.** Cont.



**Figure 14.** Evaluation results in a real environment. (a) Speed of the USV; (b) course of the USV; (c) thruster speed; (d) command propulsion angle; (e) computation time; (f) distance between the USV and the collision area.

## 5. Conclusions

The paper has proposed a collision avoidance control method for the autonomous navigation of a USV in complex and real environments. In previous research, the CAS was divided into path planning and a control system. However, in practical applications, this structure will cause a delay of collision avoidance due to information asynchrony, which is very serious in the problem of real-time collision avoidance. Therefore, it is very necessary to consider the integration of path planning and the control system into one system. To solve the problem, the theory of FCS-MPC is introduced into the CAS of the USV. The proposed CAS has some advantages such as a concise form, rapidity and easy handling of system constraints. In addition, the finite control set is generated by more practical control commands: the thruster speed and propulsion angle of the USV, which is more in accordance with the engineering requirements. The simulation experiments verify the computational speed and security of the proposed CAS in complex real scenarios with multiple dynamic obstacles.

In the application prospects, this research would provide more reliable CAS for the automatic sailing of a USV. In the next step, the practicability of CAS can be further increased for a real ship experiment. First, the environmental model directly adopts the actual shape of the obstacles, without assuming an ideal shape. Second, considering that the sea environment is complicated and changeable, it is very necessary to consider collision avoidance under the sea environmental interference. Thirdly, the Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) should be followed by CAS.

**Author Contributions:** The work presented here was performed in collaboration among all authors. X.S. designed, analyzed and wrote the paper. G.W. guided the full text. Y.F. conceived of the idea. D.M. and B.Q. analyzed the data. All authors have contributed to and given approval of the manuscript.

**Funding:** This research was funded by the Nature Science Foundation of China (Grant Number 51609033), the Nature Science Foundation of Liaoning Province of China (Grant Number 2015020022) and the Fundamental Research Funds for the Central Universities (Grant Numbers 3132014321, 3132016312 and 3132017133).

**Conflicts of Interest:** The authors declare no conflict of interest.

## Reference

1. Campbell, S.; Naeem, W.; Irwin, G.W. A review on improving the autonomy of unmanned surface vehicles through intelligent collision avoidance manoeuvres. *Annu. Rev. Control* **2012**, *36*, 267–283. [\[CrossRef\]](#)
2. Liu, Z.; Zhang, Y.; Yu, X.; Yuan, C. Unmanned surface vehicles: An overview of developments and challenges. *Annu. Rev. Control* **2016**, *41*, 71–93. [\[CrossRef\]](#)
3. Yang, J.M.; Tseng, C.M.; Tseng, P.S. Path planning on satellite images for unmanned surface vehicles. *Int. J. Naval Archit. Ocean Eng.* **2015**, *7*, 87–99. [\[CrossRef\]](#)
4. Kim, H.; Kim, D.; Shin, J.U.; Kim, H.; Myung, H. Angular rate-constrained path planning algorithm for unmanned surface vehicles. *Ocean Eng.* **2014**, *84*, 37–44. [\[CrossRef\]](#)
5. Xie, S.; Wu, P.; Liu, H.; Yan, P.; Li, X.; Luo, J.; Li, Q. A novel method of unmanned surface vehicle autonomous cruise. *Ind. Robot* **2016**, *43*, 121–130. [\[CrossRef\]](#)
6. Wu, P.; Xie, S.; Liu, H.; Li, M.; Li, H.; Peng, Y.; Li, X.; Luo, J. Autonomous obstacle avoidance of an unmanned surface vehicle based on cooperative manoeuvring. *Ind. Robot* **2017**, *44*, 64–74. [\[CrossRef\]](#)
7. Candeloro, M.; Lekkas, A.M.; Sørensen, A.J. A Voronoi-diagram-based dynamic path-planning system for underactuated marine vessels. *Control Eng. Pract.* **2017**, *61*, 41–54. [\[CrossRef\]](#)
8. Campbell, S.; Abu-Tair, M.; Naeem, W. An automatic COLREGs-compliant obstacle avoidance system for an unmanned surface vehicle. *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.* **2014**, *228*, 108–121. [\[CrossRef\]](#)
9. Kuwata, Y.; Wolf, M.T.; Zarzhitsky, D.; Huntsberger, T.L. Safe Maritime Autonomous Navigation With COLREGS, Using Velocity Obstacles. *IEEE J. Ocean. Eng.* **2014**, *39*, 110–119. [\[CrossRef\]](#)
10. Tang, P.; Zhang, R.; Liu, D.; Huang, L.; Liu, G.; Deng, T. Local reactive obstacle avoidance approach for high-speed unmanned surface vehicle. *Ocean Eng.* **2015**, *106*, 128–140. [\[CrossRef\]](#)
11. Fukushima, H.; Kon, K.; Matsuno, F. Model Predictive Formation Control Using Branch-and-Bound Compatible With Collision Avoidance Problems. *IEEE Trans. Robot.* **2013**, *29*, 1308–1317. [\[CrossRef\]](#)
12. Ji, J.; Khajepour, A.; Melek, W.W.; Huang, Y. Path Planning and Tracking for Vehicle Collision Avoidance Based on Model Predictive Control With Multiconstraints. *IEEE Trans. Veh. Technol.* **2017**, *66*, 952–964. [\[CrossRef\]](#)
13. Mayne, D.Q.; Raković, S. Model predictive control of constrained piecewise affine discrete-time systems. *Int. J. Robust Nonlinear Control* **2003**, *13*, 261–279. [\[CrossRef\]](#)
14. Richards, A.; How, J.P. Robust distributed model predictive control. *Int. J. Control* **2007**, *80*, 1517–1531. [\[CrossRef\]](#)
15. Michael, H.; Matveev, A.S.; Savkin, A.V. Algorithms for collision-free navigation of mobile robots in complex cluttered environments: A survey. *Robotica* **2015**, *33*, 463–497.
16. Sokaert, P.O.M.; Mayne, D.Q. Min-max feedback model predictive control for constrained linear systems. *IEEE Trans. Autom. Control* **1998**, *43*, 1136–1142. [\[CrossRef\]](#)
17. Ntousakis, I.A.; Nikolos, L.K.; Papageorgiou, M. Optimal vehicle trajectory planning in the context of cooperative merging on highways. *Transp. Res. Part C Emerg. Technol.* **2016**, *71*, 464–488. [\[CrossRef\]](#)
18. Johansen, T.A.; Perez, T.; Cristofaro, A. Ship Collision Avoidance and COLREGS Compliance Using Simulation-Based Control Behavior Selection With Predictive Hazard Assessment. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3407–3422. [\[CrossRef\]](#)
19. Formentini, A.; Trentin, A.; Marchesoni, M.; Zanchetta, P.; Wheeler, P. Speed Finite Control Set Model Predictive Control of a PMSM Fed by Matrix Converter. *IEEE Trans. Ind. Electron.* **2015**, *62*, 6786–6796. [\[CrossRef\]](#)
20. Mu, D.D.; Wang, G.F.; Fan, Y.S. Design of Adaptive Neural Tracking Controller for Pod Propulsion Unmanned Vessel Subject to Unknown Dynamics. *J. Electr. Eng. Technol.* **2017**, *12*, 2365–2377.
21. Zhang, R.; Li, K.; He, Z.; Wang, H.; You, F. Advanced Emergency Braking Control Based on a Nonlinear Model Predictive Algorithm for Intelligent Vehicles. *Appl. Sci.* **2017**, *7*, 504. [\[CrossRef\]](#)
22. Chen, S.L.; Cheng, C.Y.; Hu, J.S.; Jiang, J.F.; Chang, T.K.; Wei, H.Y. Strategy and Evaluation of Vehicle Collision Avoidance Control via Hardware-in-the-Loop Platform. *Appl. Sci.* **2016**, *6*, 327. [\[CrossRef\]](#)
23. Yao, P.; Wang, H.L.; Su, Z.K. Real-time path planning of unmanned aerial vehicle for target tracking and obstacle avoidance in complex dynamic environment. *Aerosp. Sci. Technol.* **2015**, *47*, 269–279. [\[CrossRef\]](#)

24. Zuhaib, K.M.; Khan, A.M.; Iqbal, J.; Ali, M.A.; Usman, M.; Ali, A.; Yaqub, S.; Lee, J.Y.; Han, C. Collision Avoidance from Multiple Passive Agents with Partially Predictable Behavior. *Appl. Sci.* **2017**, *7*, 903. [[CrossRef](#)]
25. Fossen, T.I. *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*; Marine Cybernetics: Trondheim, Norway, 2002.
26. Mu, D.; Wang, G.; Fan, Y.; Sun, X.; Qiu, B. Adaptive LOS Path Following for a Podded Propulsion Unmanned Surface Vehicle with Uncertainty of Model and Actuator Saturation. *Appl. Sci.* **2017**, *7*, 1232. [[CrossRef](#)]
27. Dong, W.; Guo, Y. Global time-varying stabilization of underactuated surface vessel. *IEEE Trans. Autom. Control* **2005**, *50*, 859–864. [[CrossRef](#)]
28. Fossen, T.I. *Guidance and Control of Ocean Vehicles*; John Wiley & Sons Inc.: Hoboken, NJ, USA, 1994.
29. Jia, X.L.; Yang, Y.S. *The Mathematical Model of Ship Motion Mechanism Modeling and Identification Modeling*; Dalian Maritime University Press: Dalian, China, 1999.
30. Sun, X.J.; Shi, L.L.; Fan, Y.S.; Wang, G.F. Online Parameter Identification of USV Motion Model. *Navig. China* **2016**, *1*, 39–43.
31. Wang, N.; Meng, X.; Xu, Q.; Wang, Z. A unified analytical framework for ship domains. *J. Navig.* **2009**, *62*, 643–655. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).