An Applicable Multi-Ship Collision Avoidance Control Method Based on Velocity Obstacle

Yong Xiong[1,2], Liwen Huang[1,2], Xiaochun Wu[1,2]

[1] *School of Navigation*, Wuhan University of Technology, 190 Peace Avenue, Wuhan ,430063, China
[2]. Hubei Key Laboratory of Inland Shipping Technology,Wuhan University of Technology,Wuhan 430063,China

***Abstract* — Automatic collision avoidance from obstacles is important, but the conventional methods for ship collision avoidance emphasize how to design the feasible routes for collision avoidance basing on a ship's real time position, not to consider the dynamics or control for ship. . In this paper, an applicable multi-ship collision avoidance control method on the basis of the directions of relative velocity is presented, which applies to the routing plan of collision avoidance and ship motion control simultaneously. This method consists of three main steps. The first step is extracting/calculating basic motion and position parameters of the own ship and target ships. The second step is calculating suitable real time directions of velocity of the own ship basing on the method of velocity obstacle. The third step is controlling the variable transformation and algorithm design. This method is simple and flexible. It can be applied to the real time multi-ship collision avoidance, and is very feasible. Simulation results show that the algorithm has good effects.**

Keywords - multi-ship collision avoidance, velocity obstacle zone, PID control, Limited Collision Avoidance Distance

I. INTRODUCTION

Ship collision avoidance is an important problem for automatic navigation of water surface vessels, as more and more vessels are navigating in the water channels, especially when the channels are narrow and busy. So the possibility of collision is increased for all the ships. Ship collision avoidance has been researched for a long time, but early researches only focused on the routing plan of ship collision avoidance and assessment of collision risks [1-7], which assumes that the ship and its target ships all have uniform linear motion, but the kinetics of the own ship is not considered. So these methods are called GACM (geometric collision avoidance method). GACM is a traditional method that make collision avoiding decisions depending on ship's position information and captain's experiences. In the last decade, some modern methods were introduced to solve this problem. Especially, non-analytical methods were introduced in the area of artificial intelligence. For example, the genetic algorithm was used to search for optimal routes for collision avoidance [8-11], and the expert system combined with fuzzy logic method was used to create a risk assessment model of ship collision[12-14], etc. The routing plans of collision avoidance are important in the conventional methods. But how to control the ship to move along the planned trajectory of collision avoidance is not involved in the above methods. Therefore, these methods are not real time methods for collision avoidance.

Artificial intelligence was being applied very early in the robot motion control, especially in collision avoidance, path planning and so on. But, there is less practical application in the field of ship collision avoidance. Due to the ship motion has strong nonlinear characteristics, so it is difficult to establish mathematical model from the theory of precise dynamic, which leads the difficulties of the ship motion control, and the inaccuracy of motion control，so ship may not complete the collision avoidance. Therefore artificial intelligence method is introduced to the study of ship collision avoidance. Any animal knows to avoid the moving obstacles, because they can feel and estimate the relative velocity of the object through eyes and brain, so the basic principle of animal is researched to avoid the movement of static obstacle, and that is bionic intelligence, which is an important area in artificial intelligence. Relative velocity obstacle was represented, which is the general method of avoiding obstacle arisen from animals' behavior of avoiding collision. The ship automatic collision avoidance can be promoted by this method [15-16].

The relative velocity obstacle method was introduced by Fiorini P and Shiller Z first [17]. It is a straightforward and easy method to solve collision avoidance problems of moving objects. The relative velocity obstacle method has been used to deal with robot collision avoidance, vehicle automatic navigation, and tracking problems of plane motion, but has not been applied to the problems of ship collision avoidance. In this paper, a multi-ship collision avoidance control method basing on relative velocity obstacle is represented. The basic idea of this method is as follows: first, calculating the relative velocity of the own ship with respect to all target ships; second, controlling the direction of the relative velocity of the own ship not to point to any target ship at any time, which can make the own ship avoid possible obstacles no matter whether they are still or moving. An alert index is necessary in ship collision avoidance system, which can be used to assess collision risks of the own ship with respect to any target ship in time, so the own ship knows which target ship is the most risky one in terms of collision and then decides to avoid it at any time. In this paper, Limited Collision Avoidance Distance (LACD) is used to measure ship collision risks, and it is more appropriate than Distance to Closest Point of Approach （DCPA） because the dynamics of the own ship in the process of collision avoidance is taken into account. So it is more accurate to measure the danger level of ship collision than DCPA.

As stated above, this method can automatically evaluate collision risks and choose a target ship to avoid collisions at any time. The process will be continued until the own ship gets out of the collision area and gets back to the planned route. This is automatic and doesn't need human interference if all the parameters of this method have been adjusted to the appropriate values. Simulate results will verify the effectiveness of this method.

This paper is arranged as follows. Section 1 is the introduction, and section 2 describes the ship collision avoidance problem. Section 3 creates the ship dynamic model. Section 4 introduces the ship collision avoidance method based-on velocity obstacle. Section 5 presents the ship course control and recovery, and section 6 introduces the ship alert system with LACD. Section 7 shows the simulation results, analyses and discussions.

II. PROBLEM DESCRIPTION

On figure 1, at time t, the own Ship O moves at velocity V_o according to the planned route OB, and the target ships P_1, P_2,… P_i appear randomly, which move on the water surface at unified line velocities V_{P_1}, V_{P_2}, V_{P_i} respectively. Now, it is needed to design a method based-on velocity obstacle to meet the following objectives:

(1)Alert the level of collision risks, and suggest the own ship when to avoid collisions.

(2)Give warnings in advance so that the own ship O can avoid collisions with the dangerous target ship P_i, if collision is inevitable.

(3)Suggest the own ship O recovering to the planned route after finishing the process of collision avoidance.

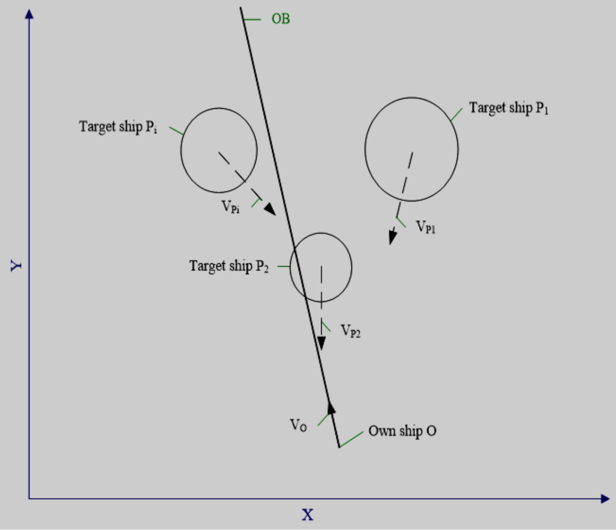

Fig.1ship avoid collision problem.

The method mentioned above has the following assumptions or considerations:

1.There are no environmental interference forces, for example: wave, wind or current force.

2. International Regulations for Preventing Collisions at Sea 1972(COLREGS) are not considered.

This method just focuses on the main part of the problem, and the emphasis is to explore how to control the own ship to avoid the moving target ship on the calm water surface, so it does not seek to meet the requirements in COLREGS at this time.

III. SHIP DYNAMIC MODEL

The own ship motion is determined by its dynamics, so the ship motion mathematic model can be used to calculate its velocity and position. From this point of view, MMG model is more popular and practical than other ship dynamic models. In this paper, a three degrees-of-freedom maneuvering MMG model is used to represent the own ship motion, which can represent surge, yaw and sway motions under body-fixed coordinates, and the superscript prime represents non-dimensional form of variable.

$$\begin{cases} \left(m'+m'_x\right)\dot{u}'-\left(m'+m'_y\right)v'r'=X'_H+X'_p+X'_R \\ \left(m'+m'_y\right)\dot{v}'+\left(m'+m'_x\right)u'r'=Y'_H+Y'_R \\ \left(I'_{zz}+i'_{zz}\right)\dot{r}'=N'_H+N'_R \end{cases} \tag{1}$$

ρ is liquid's density. L , d are ship length and draft ,respectively.

$$m' = \frac{m}{0.5\rho L^2 d}, \quad m'_x = \frac{m_x}{0.5\rho L^2 d}, \quad m'_y = \frac{m_y}{0.5\rho L^2 d},$$

so m, m_x, m_y are the own ship's mass, added mass along x-axis, added mass along y-axis of the body-fixed coordinates respectively.

$$I'_{zz} = \frac{I_{zz}}{0.5\rho L^4 d}, \quad J'_{zz} = \frac{J_{zz}}{0.5\rho L^4 d},$$

so I_{zz}, J_{zz} are the own ship's moment of inertia, added moment of inertia accord to z-axis of the body-fixed coordinates.

$$X' = \frac{X}{0.5\rho L U^2 d}, \quad Y' = \frac{Y}{0.5\rho L U^2 d},$$

so X, Y are the external forces along x-axis and y-axis respectively.

$$N' = \frac{N}{0.5\rho L^2 U^2 d}, \quad r' = \frac{rL}{U},$$

N, r are yaw moment around z-axis and yaw angular velocity respectively.

$$u' = \frac{u}{U}, \quad v' = \frac{v}{U}, \quad U = \sqrt{v^2 + u^2},$$

u, v are yaw angular velocity, velocity along x-axis, velocity along y-axis, respectively, U is total ship velocity at earth-fixed coordinates.

Then, propeller force X_P is computed using the following equation,

$$X'_P = c_{t_P}(1 - t_{P0}) n^2 D_P^4 \frac{K_T(J_P)}{0.5 L d U^2} \tag{2}$$

And

$$K_T(J_P) = c_1 + c_2 J_P + c_3 {J_P}^2 \tag{3}$$

$$J_P = \frac{u(1 - w_P)}{n D_P} \tag{4}$$

c_{t_P}, t_{P0}, c_1, c_2, c_3 and w_P are constants, n is rotational speed of the propeller, D_P is the diameter of the propeller.

Computing the rudder force X_R, Y_R and moment N_R,

$$X'_R = -(1 - t_r) F'_n \sin\delta \tag{5}$$

$$Y'_R = -(1 + a_h) F'_n \cos\delta \tag{6}$$

$$N'_R = -(x'_r + a_h x'_h) F'_n \cos\delta \tag{7}$$

δ is the rudder angle, t_r, a_h, x'_r, x'_h are the constants. F'_n is the main rudder force.

The forces X_H, Y_H acting on the ship hull are along x-axis and y-axis, respectively. N_H is the external moment acting on ship, which is calculated using Inoue model.

$$X'_H = X'_{uu} u'^2 + X'_{vv} v'^2 + X'_{vr} v' r' + X'_{rr} r'^2 \tag{8}$$

$$Y'_H = Y'_v v + Y'_r r + Y'_{v'|v'|} v'|v'| + Y'_{r'|r'|} r'|r'| + Y'_{v'|r'|} v'|r'| \tag{9}$$

$$N'_H = N'_{r'} r' + N'_{v'} v' + N'_{r'|r'|} r'|r'| + N'_{v'|v'|} v'|v'| + N'_{v'v'r'} v'^2 r' + N'_{r'r'v'} r'^2 v' \tag{10}$$

Then transform the ship body-fixed coordinates to earth-fixed coordinates

$$\begin{cases} \dot{x} = u'\cos\psi - v'\sin\psi \\ \dot{y} = v'\cos\psi + u'\sin\psi \\ \dot{\psi} = r' \end{cases} \tag{11}$$

Then combine (1) and (11). The ship motion position can be determined at any time point. And we assume other ships have unified linear motion, which meet the motion equation:

$$\begin{cases} y_f = a_1 t + b_1 \\ x_f = a_2 t + b_2 \end{cases} \tag{12}$$

And the shape of the obstacle is assumed to be a circle, which can be defined by:

$$(x_c - x_f)^2 + (y_c - y_f)^2 = R^2 \tag{13}$$

IV. SHIP COLISION AVOIDANCE METHOD BASED-ON RELATIVE VELOCOTY OBSTACLE

Ship collision avoidance is a fundamental problem of automatic ship navigation, which can be seen as a special trajectory tracking problem, because its planned trajectory could be changed continuously, depending on different situations that the ship might meet, so that the own ship can avoid collisions with target ships at every time point. In this section, the concept of velocity obstacle will be introduced, and then a very straightforward and simple method of collision avoidance will be used, which will transform the collision avoidance problem of moving object to a collision avoidance problem of stationery object through relative velocity.

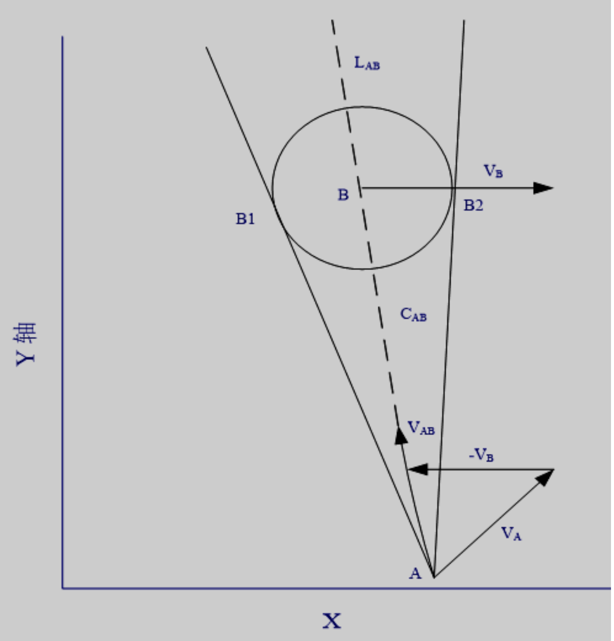

Fig.2 Velocity obstacle.

The basic idea is described as follows. In Figure.2, A and B are the own ship and the target ship respectively. $V_A(t)$, $V_B(t)$ are the velocity vectors at time point t respectively, and $V_{AB}(t)$ is the relative velocity of A to B at time point t. If B is a static object, $V_B(t)$ =0. A conflicts with B if $V_A(t)$'s direction falls into $\angle B_1AB_2$ zone. If B is a moving object, $V_B(t)$ ≠0. If $V_{AB}(t)$, s direction falls into $\angle B_1AB_2$ zone , then A will inevitably collide with B. So first, the velocity obstacle should be defined to judge what the necessary conditions of ship collision avoidance are, and then a collision avoidance algorithm is designed depending on the complication of the ship meeting situation.

Definition 1: Velocity obstacle zone $C_{AB}(t)$ of A relative to B:

$$C_{AB}(t) = \{V_{AB}(t) | L_{AB} \cap B \neq \phi\} \quad (14)$$

So the velocity obstacle zone $C_{AB}(t)$ is the triangle zone B_1AB_2 in figure 2, and in this area, the relative velocity line L_{AB} across ship B, so A would collision with B.

Definition 2: Velocity obstacle $VO(t)$

$$VO(t) = C_{AB}(t) \otimes V_B(t) \quad (15)$$

$\otimes$ is the Minkowski vector sum operator.

When the own ship meets n different target ships, this will be a multi-ship collision avoidance problem. If the target ships move in the unified linear motion, the i-th target ship should satisfy the following equations.

$$\begin{cases} x_i(t) = x_{i0} + u_i t \\ y_i(t) = y_{i0} + v_i t \end{cases} \quad (16)$$

(x_{i0}, y_{i0}) is the original position of the i-th target ship. u_i, v_i are the initial velocities of the i-th target ship in x- and y-axes respectively. Then $VO_i(t)$ is the velocity obstacle of the own ship relative to the i-th target ship. The total velocity obstacle $VO(t)$ is the union of n different target ship velocity obstacles $VO_i(t)$ if that is a multi-ship meeting situation.

$$VO(t) = \bigcup_{i=1}^{n} VO_i(t) \quad (17)$$

So if $V_A(t) \notin VO(t)$ at any time point, which means

$$V_A(t) = \overline{VO}(t) = \overline{\bigcup_{i=1}^{n} VO_i(t)} \quad (18)$$

equation (18) is a sufficient condition of ship collision avoidance. Although this equation is simple, it is the only formal expression. So how to compute $VO_i(t)$ according to different ship meeting situations is the key step. In figure 3, O is the own ship, and P_i represents the i-th target ship, then the ship collision avoidance method based-on velocity obstacle would be represented as follow.

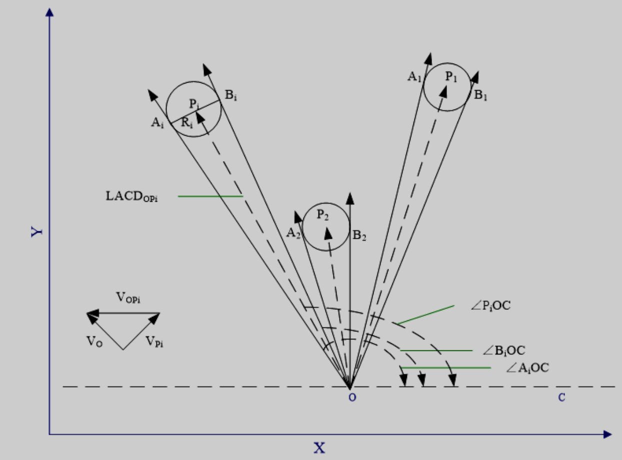

Fig.3.Multi-ship collision avoidance.

(1)Compute the coordinates of the own ship's position O $\left(x(t),y(t)\right)$ at time point t depending on ship motion equations (1) and (11) using Runge-Kutta method. Compute the coordinates of the real time position P_i of the i-th target ship, $\left(x_i(t),y_i(t)\right)$, using the unified linear equation.

(2)Calculate $\angle P_1OC$, $\angle P_2OC$ … $\angle P_nOC$, and $\angle A_1OC$, $\angle A_2OC$ … $\angle A_nOC$ and $\angle B_1OC$, $\angle B_2OC$ … $\angle B_nOC$ depending on the position coordinates of the own ship and target ship.

$$\angle P_iOC=\begin{cases}\operatorname{arc}tg\left(\dfrac{y(t)-y_i(t)}{x(t)-x_i(t)}\right) & if \quad x(t)-x_i(t)>0\\ \pi+\operatorname{arc}tg\left(\dfrac{y(t)-y_i(t)}{x(t)-x_i(t)}\right) & if \quad x(t)-x_i(t)<0\end{cases} \tag{19}$$

$$\angle A_iOC=\angle P_iOC+\arcsin\left(\frac{R_i}{\sqrt{\left(x(t)-x_i(t)\right)^2+\left(y(t)-y_i(t)\right)^2}}\right) \tag{20}$$

$$\angle B_iOC=\angle P_iOC-\arcsin\left(\frac{R_i}{\sqrt{\left(x(t)-x_i(t)\right)^2+\left(y(t)-y_i(t)\right)^2}}\right) \tag{21}$$

$i=1,2...n$, R_i is i-th obstacle radius.

(3)Sort $\angle P_iOC$ in the ascending order, so $\angle P_{i_1}OC<\angle P_{i_2}OC...<\angle P_{i_n}OC$. Then sort $\angle A_iOC$, $\angle B_iOC$ according to the same ascending subscripts i_1, i_2 … i_n.

(4)Calculate the relative velocity $V_{OP_{i_j}}$, and the angle $\angle V_{OP_{i_j}}=\operatorname{arc}tg\left(\dfrac{\dot{y}(t)-v_{i_j}}{\dot{x}(t)-u_{i_j}}\right)$, $j=1,2...n$.

(5)From $j=1$, compare $\angle V_{OP_{i_1}}$ and $\angle A_{i_1}OC$, $\angle B_{i_1}OC$. If $\angle B_{i_1}OC<\angle V_{OP_{i_1}}<\angle A_{i_1}OC$, the own ship will collision with the target ship P_{i_1}, and there will be four different meeting situations:

A1. Overlapping in the left side's velocity obstacle zone: $\angle A_{i_1}OC>\angle B_{i_1+1}OC$ and $\angle B_{i_1}OC>\angle A_{i_1-1}OC$

A2. Overlapping in the right side's velocity obstacle zone: $\angle A_{i_1}OC<\angle B_{i_1+1}OC$ and $\angle B_{i_1}OC<\angle A_{i_1-1}OC$

A3. Overlapping in both sides' velocity obstacle zones: $\angle A_{i_1}OC>\angle B_{i_1+1}OC$ and $\angle B_{i_1}OC<\angle A_{i_1-1}OC$

A4. Clearance on both sides: $\angle A_{i_1}OC<\angle B_{i_1+1}OC$ and $\angle B_{i_1}OC>\angle A_{i_1-1}OC$

Else, $j=j+1$, repeat the above process until $j=n$.

(6)Now the relative velocity angle $\angle V_{OP_{i_1}}$ must be adjusted to avoid the velocity obstacle zone, which will be solved depending on the relative velocity angle equation (22). θ_d has different values in different meeting cases.

$$\frac{V\sin\beta_d-v_i}{V\cos\beta_d-u_i}=tg\theta_d \tag{22}$$

$$V=\sqrt{\dot{x}(t)^2+\dot{y}(t)^2} \tag{23}$$

In case A1, $\theta_d=\angle B_{i_1}OC$; In case A2, $\theta_d=\angle A_{i_1}OC$; In case A4, if $\angle B_{i_1}OP_{i_1}<\angle A_{i_1}OP_{i_1}$, $\theta_d=\angle A_{i_1}OC$. Else, $\theta_d=\angle B_{i_1}OC$.

In case A3, consider ships P_{i_2}, P_{i_3} to P_{i_n}, then return to 5.

The value of θ_d should be a little bigger than the theoretic value for safely pass the obstacle.

Then solve equation (22), and set

$$\Delta=\sqrt{V^2(1+(tg\theta_d)^2)+2tg\theta_d u_i v_i-(tg\theta_d)^2u_i^2-v_i^2} \tag{24}$$

If $\Delta>0$, we can get two appropriate solutions for the own ship angle β_d.

$$\beta_d^{\,1}=\arccos\left(((tg\theta_d)^2u_i-v_itg\theta_d+\Delta)/\left(V(tg\theta_d)^2+V\right)\right) \tag{25}$$

$$\beta_d^{\,2}=\arccos\left(((tg\theta_d)^2u_i-v_itg\theta_d-\Delta)/\left(V(tg\theta_d)^2+V\right)\right) \tag{26}$$

Then, $$\beta_d=\begin{cases}\beta_d^{\,1}, if\,(\beta_d^{\,1}-\psi(t))^2<(\beta_d^{\,2}-\psi(t))^2\\ \beta_d^{\,2}, if\,(\beta_d^{\,1}-\psi(t))^2>(\beta_d^{\,2}-\psi(t))^2\end{cases} \tag{27}$$

V. SHIP COURSE ALTERATION AND RESUME

There are many methods for ship course control, for example, PID control, predictive control, adaptive control, neural network control and so on, some of which can be used to deal with this problem. In terms of generality and simplicity, PID control which is the most popular method can be used to solve this problem, while of course any other ship control method can be also used to deal with it. First, steering-wheel dynamics must be considered before applying a ship course control algorithm. Usually, the steering gear satisfies the equation below:

$$\dot{\delta} = -\frac{1}{T_e}\delta + \frac{K_e}{T_e}\delta_e , \quad \delta(0) = \delta_0 \tag{28}$$

δ_e is the order angle, δ is the real time angle, T_e is the time constant, K_e is the steering gear control gain constant. Solve this equation,

$$\delta = K_e\delta_e - K_e(\delta_e - \delta_0)e^{-\frac{t}{T}} = K_e\delta_e(1 - e^{-\frac{t}{T}}) + K_e\delta_0 e^{-\frac{t}{T}} \tag{29}$$

$$e = \beta_d - \psi \tag{30}$$

e is the error between the set value and the real value.

Under normal conditions, the rudder angle is constrained as $-\delta_{\max} \le \delta \le \delta_{\max}$. In this paper,

$$\delta_{\max} = 35^0 = \frac{35}{180} * \pi = 0.61 \tag{31}$$

So, use the transformation of hyperbolic tangent function for the rudder angle constraint,

$$\delta_e = \delta_{\max}\frac{e^{\xi/\delta_{\max}} - e^{-\xi/\delta_{\max}}}{e^{\xi/\delta_{\max}} + e^{-\xi/\delta_{\max}}} \tag{32}$$

Then use the control variable $\xi \subset (-\infty, +\infty)$ to substitute the order rudder angle variable $\delta_e \subset (-\delta_{\max}, \delta_{\max})$, and an incremental PID control algorithm is designed for the own ship to keep course β_d.

$$\xi_{k-1} = K_P\left[e_{k-1} + \frac{T}{T_i}\sum_{j=0}^{k-1} e_j + T_d\frac{e_{k-1} - e_{k-2}}{T}\right] \tag{33}$$

$$\Delta\xi_k = \xi_k - \xi_{k-1} = K_P\left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right)e_k - K_P\left(1 + \frac{2T_d}{T}\right) e_{k-1} + K_P\frac{T_d}{T}e_{k-2} = Ae_k - Be_{k-1} + Ce_{k-2} \tag{34}$$

So in equation (39), the coefficient

$$A = K_P\left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right), \quad B = K_P\left(1 + \frac{2T_d}{T}\right), \quad C = K_P\frac{T_d}{T}$$

are all constants, where $\xi_k = \xi_{k-1} + \Delta\xi_k$.

When the own ship O has passed DCPA from the target ship P_i, it should resume its original course. So a judge rule is needed to decide the beginning time to resume the course. If the joining line OP_i of the own ship and the target ship is perpendicular to the target ship P_i' s velocity V_{P_i}, which means the own ship arrives at DCPA point, then the own ship should begin to resume to the originally planned course, so the resume time point should satisfy:

$$\frac{\dot{y}(t) - v_i}{\dot{x}(t) - u_i} * \frac{x(t) - x_i(t)}{y(t) - y_i(t)} = -1 \tag{35}$$

VI. SHIP ALERT SYSTEM BASED-ON LACD

The own ship should scan all its nearby target ships in the counter-clockwise order, then the computational task will be very big if every nearby target ship is to be tracked and collision avoidance algorithm is applied at every time point. LACD (Limited Avoid Collision Distance) is a collision risk gauge of the own ship with target ship, and it is different from DCPA. LACD involves not only the real time ship position but also the ship dynamics, so the own ship can track and avoid collisions with the most dangerous target ship. Thus, the computation time will be reduced to an acceptable level at every time point.

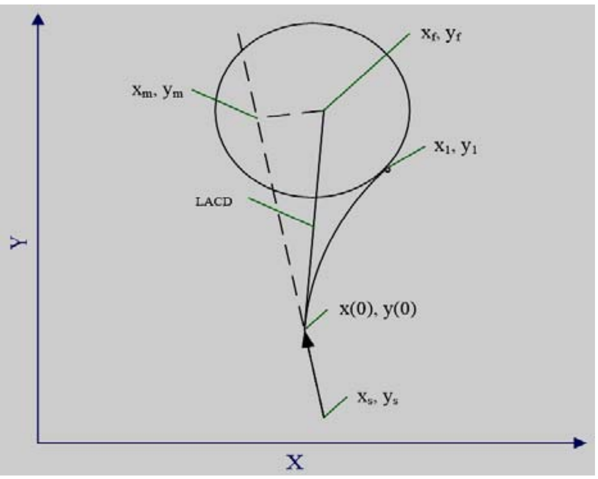

Fig.4 Limited avoid collision distance.

On figure 4, LACD can be defined as follows.

When a ship begins to move at point $\left(x(0), y(0)\right)$, and it has the biggest right turn rudder angle δ according to the motion equations (1) and (16), it is tangent to the obstacle circle. So

$$LACD=\sqrt{\left(x(0)-x_f\right)^2+\left(y(0)-y_f\right)^2} \tag{36}$$

So the key step is to decide the point $\left(x(0), y(0)\right)$, and the necessary condition is that the own ship's trajectory slope is equal to the obstacle's circle slope at point of tangency:

$$Dy/Dx(ship)=\frac{\dot{y}}{\dot{x}}=\frac{v\cos\psi+u\sin\psi}{u\cos\psi-v\sin\psi} \tag{37}$$

$$Dy/Dx(obstacle\quad circle)=\frac{-(y-y_f)}{(x-x_f)} \tag{38}$$

So,

$$\frac{v\cos\psi+u\sin\psi}{u\cos\psi-v\sin\psi}=-\frac{(y-y_f)}{(x-x_f)} \tag{39}$$

For an analytical solution of the ship trajectory curve, we must combine (1), (11), and (39). Unfortunately, it is very difficult and often impossible because it is a complicated nonlinear equation. So another choice is to design a numerical algorithm to calculate LACD.

The program is expressed in Figure .4.

(1) Set i=1. $\left(x_s, y_s\right)$, $\left(x_m, y_m\right)$ are the initial and final reference positions.

(2) Set an initial position $\left(x(0), y(0)\right)$, and compute ship trajectory $\left(x(i), y(i)\right)$ depending on ship motion equation (1) and (16) using Runge-Kutta method.

(3) Calculate the distance

$$D(i)=\sqrt{\left(x(i)-x_f(i)\right)^2+\left(y(i)-y_f(i)\right)^2},$$

if $D(i)\geq R$ and $D(i)\leq D(i-1)$, i=i+1.

Set

$$S=\left(x(i-1)-x_f\right)\left(y(i)-y_f\right)-\left(y(i-1)-y_f\right)\left(x(i)-x_f\right)$$

to judge the position of the target ship $\left(x_f, y_f\right)$ relative to the velocity direction of the own ship.

(4) If $S<0$, that means that the target ship is on the right of the own ship's trajectory, then reset $x(0)=\frac{x(0)+x_s}{2}$, $y(0)=\frac{y(0)+y_s}{2}$, and return to 2.

(5) If $S>0$, that means that the target ship is on the left of the own ship's trajectory, and $D(i)\geq R$ and $D(i)\geq D(i-1)$ and $D(i-1)\leq D(i-2)$, stop calculation. Reset $x(0)=\frac{x(0)+x_m}{2}$, $y(0)=\frac{y(0)+y_m}{2}$, and return to 2.

(6) If $D(i)\leq R$, stop calculation. Reset $x(0)=\frac{x(0)+x_s}{2}$, $y(0)=\frac{y(0)+y_s}{2}$, and return to 2.

(7) If $\sqrt{\left(D(i)-R\right)^2+\left(D(i-1)-R\right)^2}\leq\mu$, stop and output the final $\left(x(0), y(0)\right)$.

(8) Output the final results, $LACD=\sqrt{\left(x(0)-x_f\right)^2+\left(y(0)-y_f\right)^2}$, and end.

VII. SIMULATION EXPERIMENTS AND RESULT

The simulation experiments are carried out basing on Mariner classic vessel, whose features are as follows.

Overall length is 171.8m, the length between perpendiculars is 160.93m, maximum beam is 23.17m, designed draft is 8.23m, designed displacement is 18541 m3, and designed speed is 15 knot. Other details about the own ship' parameters can be found on Fosse's classic book [18].

The own ship Mariner's initial position is (0, 0) in the earth-fixed coordinates, and the initial velocity is 7.72m/s in the following simulation experiments. Without loss of generality, the own ship is assumed to be a mass point, and every target ship's radius is set to 200 m. Every target ship's initial position and velocity are set randomly depending on different ships' meeting situations, which will be shown in table I, table II, table III, table IV.

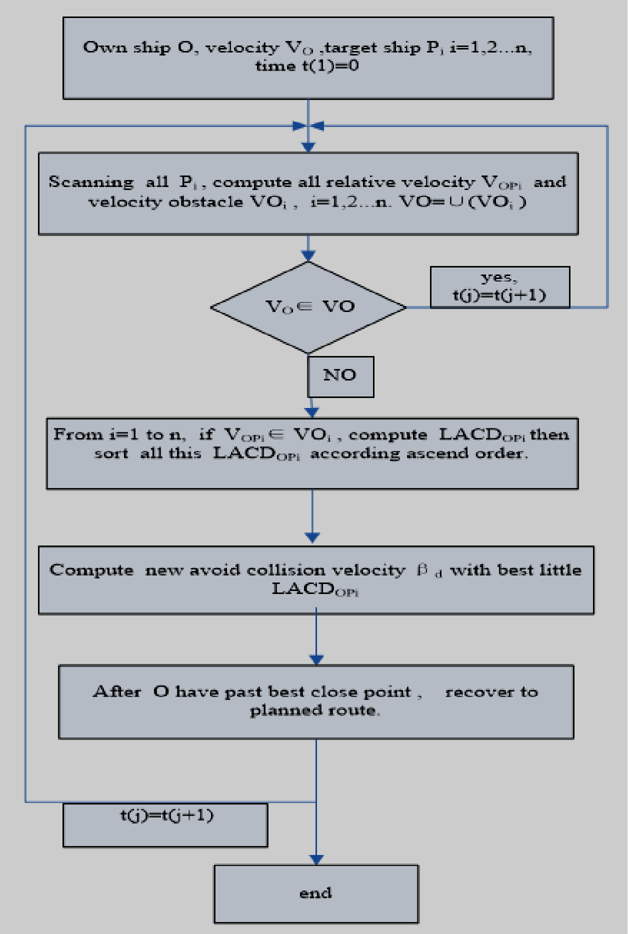

Fig.5 Flow chart.

The own ship uses PID control to realize course keeping, and the P, I, D parameters are KP=6, KI=400, KD=6.

Figure 5 shows the flowchart for the complete steps and process of the simulation experiments depending on the above sections. Simulation results will be analyzed and discussed.

According to the ascending order, the one with smallest collision avoidance velocity has passed, and resume to the simulation results which can be expressed depending on four ship meeting situations, A1, A2, A3, A4.

TABLE. I. MEETING SITUATION A1

	initial x-axis position (m)	initial y-axis position (m)	x-axis velocity (m/s)	y-axis velocity (m/s)
Target Ship 1	402.6	0	3.2	0.1
Target Ship 2	782.7	572.5	-6.4	-0.1
Target Ship 3	902.6	-804.4	-3.3	0.2

TABLE II. MEETING SITUATION A2.

	Initial x-axis position (m)	initial y-axis position (m)	x-axis velocity (m/s)	y-axis velocity (m/s)
Target Ship 1	400.5	0	-4.3	0.1
Target Ship 2	700.3	-446.5	-4.8	1.1
Target Ship 3	900.5	803.0	-2.1	-1.0

TABLE III. MEETING SITUATION A3.

	initial x-axis position (m)	initial y-axis position (m)	x-axis velocity (m/s)	y-axis velocity (m/s)
Target Ship 1	353.7	0	-4.7	0.01
Target Ship 2	649.8	636.7	-4.8	-1.0
Target Ship 3	788.6	-1186.9	-4.4	0.3

TABLE. IV. MEETING SITUATION A4

	initial x-axis position (m)	initial y-axis position (m)	x-axis velocity (m/s)	y-axis velocity (m/s)
Target Ship 1	356.0	0	-2.4	0.01
Target Ship 2	655.4	-422.3	-2.9	1.0
Target Ship 3	756.6	481.2	-3.0	-1.1

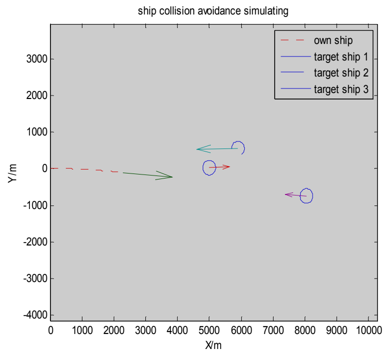

Fig.6

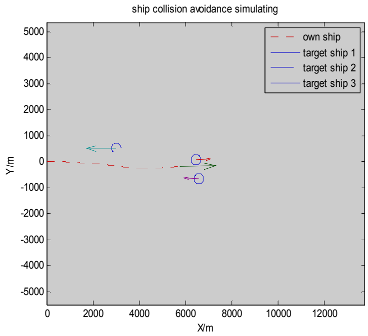

Fig.7

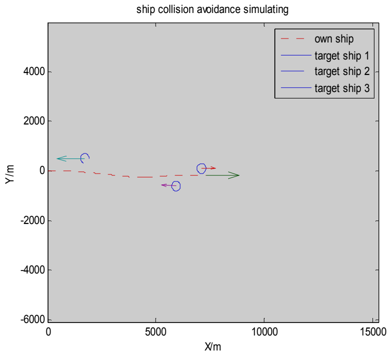

Fig.8

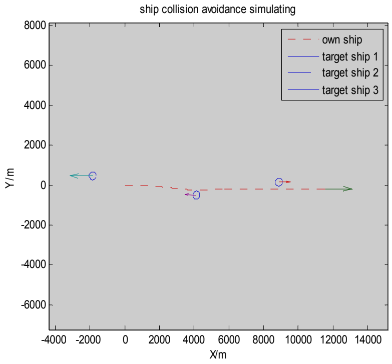

Fig.9

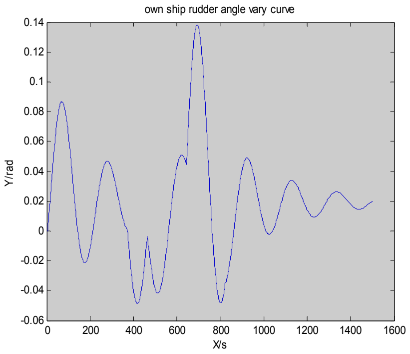

Fig.10

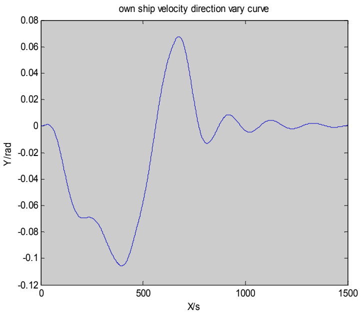

Fig.11

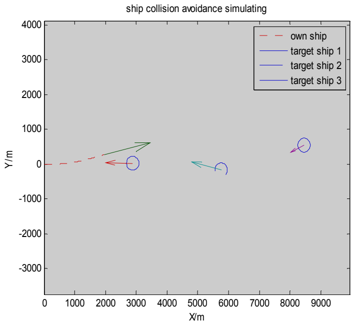

Fig..12

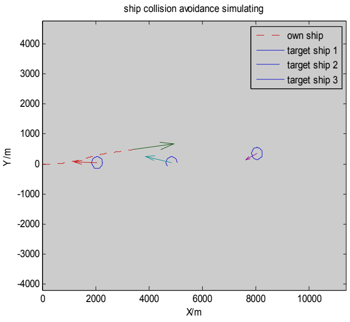

Fig.13

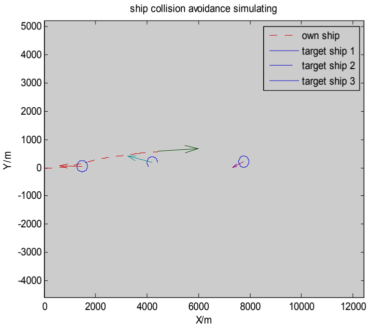

Fig.14

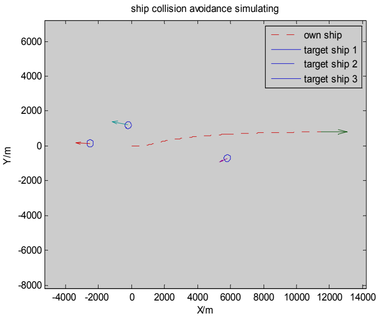

Fig.15

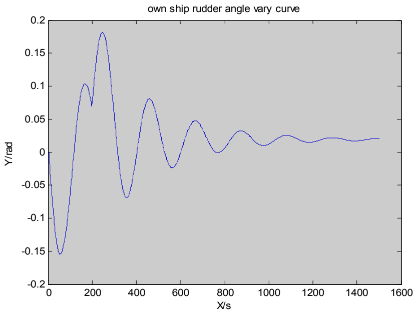

Fig.16

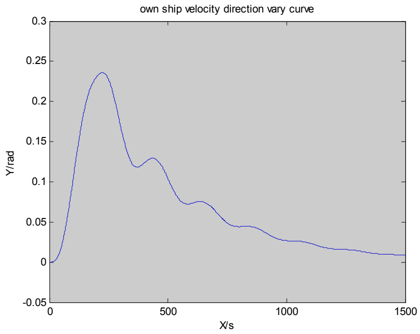

Fig.17

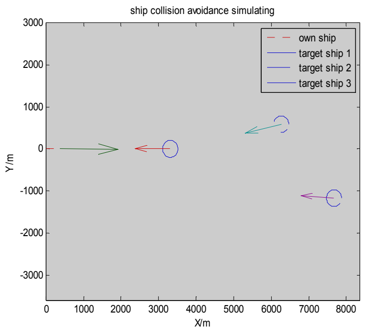

Fig.18

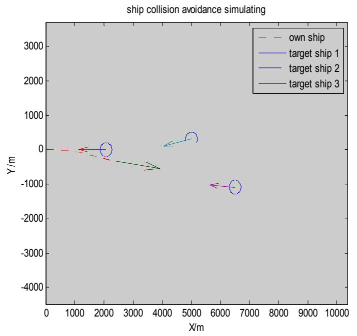

Fig.19

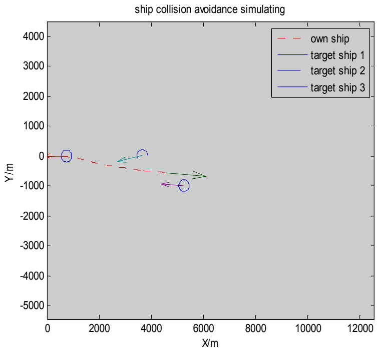

Fig.20

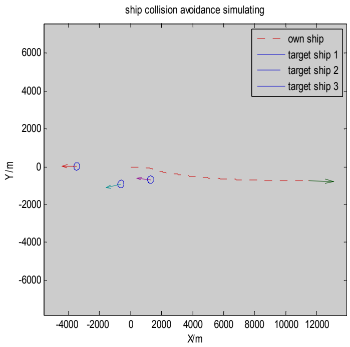

Fig.21

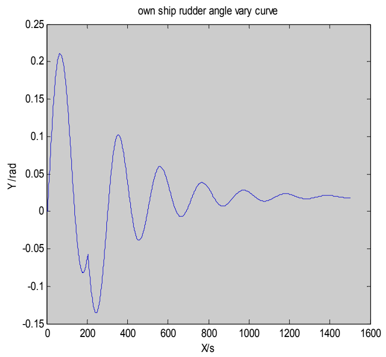

Fig.22

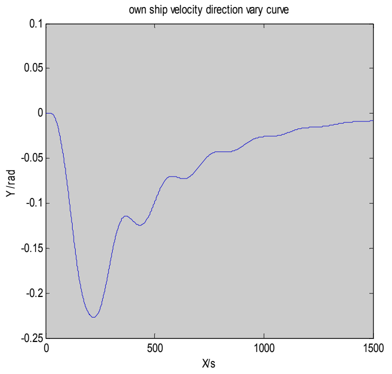

Fig.23

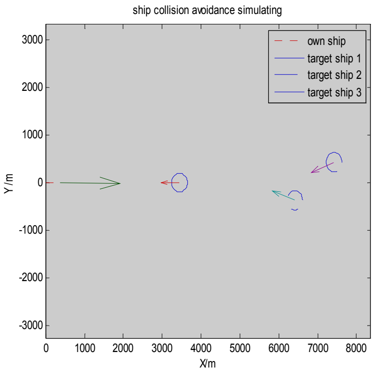

Fig.24

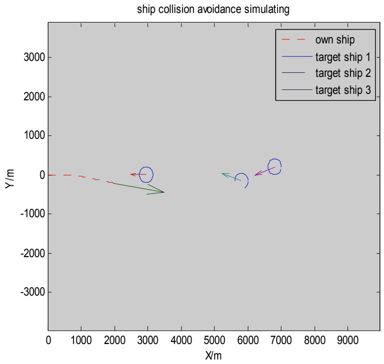

Fig.25

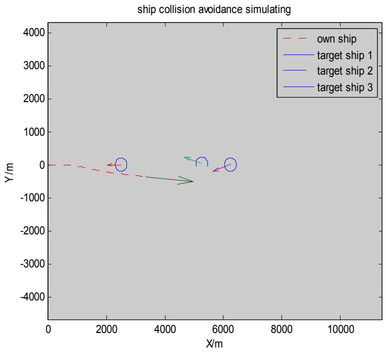

Fig.26

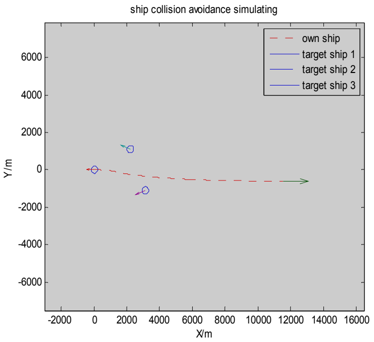

Fig.27

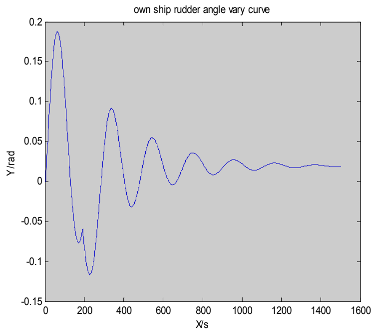

Fig.28

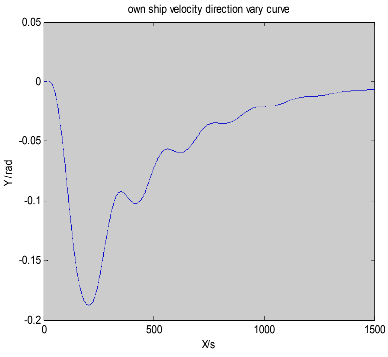

Fig.29

VIII. DISCUSION AND CONCLUSION

As it can be seen in the above simulation results, the own ship can avoid all the collisions with target ship in the four different ship meeting situations A1, A2, A3, A4. But the effects of the ship collision avoidance are affected by three factors.

1. Distance between the own ship and the target ships. When the distance is longer, the own ship is safer. When the distance is shorter than LACD, the own ship is impossible to complete collision avoidance with the target ships. LACD is a basic parameter to judge whether the action of collision avoidance can be executed successfully. So the own ship should begin the actions of avoiding collision earlier for better safety in real navigation. But the ship will consume too much fuel if collision avoidance is executed too often and early. So how to set an appropriate starting distance of collision avoidance is very important. Usually, it can be set 5 nm basing on the experience of navigation.
2. Ship meeting situations. The collision avoidance methods automatically judge the ship meeting situations according to all the ships' real time positions, which change continuously with time. The own ship will compute the directions of relative velocity continuously depending on the ship meeting situations, then solve relative motion equations to get the own ship's velocity direction, and put it to the PID control module.
3. Performance of the own ship PID control. In terms of ship collision avoidance, we need to consider not only how to get a right collision avoidance route but also how to control the own ship motion basing on the planned route. So if the performance of the own ship's course-keeping control is very bad, the performance of collision avoidance will also be bad. Different methods can be used in own ship's course-keeping control other than PID control. But no matter what control method is used, we need to adjust control parameters for good control.

ACKNOWLEDGMENT

The research was sponsored by grants from the Key Project in the National Science & Technology Pillar Program (Grant No.2015BAG20B05) funding.

REFERENCES

[1] Kwik K H. Calculation of ship collision avoidance manoeuvres: A simplified approach[J]. Ocean engineering, 1989, 16(5): 475-491.

[2] Large F, Vasquez D, Fraichard T, et al. Avoiding cars and pedestrians using velocity obstacles and motion prediction[J]. IEEE Intelligent Vehicle Symposium, 2004:375 - 379.

[3] Kluge B, Prassler E. Recursive Probabilistic Velocity Obstacles for Reflective Navigation.[M]// Field and Service Robotics. Springer Berlin Heidelberg, 2003:71-79.

[4] Zhang J, Zhang D, Yan X, et al. A distributed anti-collision decision support formulation in multi-ship encounter situations under COLREGs[J]. Ocean Engineering, 2015, 105:336–348.

[5] Bouchafa S, Zavidovique B. C-velocity: A cumulative frame to segment objects from egomotion[J]. Pattern Recognition & Image Analysis, 2009, 19(4):583-590.

[6] Tsoukalas A, Tzes A. Modelling and Control of Hyper-Redundant Micromanipulators for Obstacle Avoidance in an Unstructured Environment[J]. Journal of Intelligent & Robotic Systems, 2015, 78(3-4):517-528.

[7] Kim D, Hirayama K, Okimoto M. Ship Collision Avoidance by Distributed Tabu Search[J]. TransNav: International Journal on Marine Navigation and Safety of Sea Transportation, 2015, 9.

[8] Jatmiko W, Sekiyama K, Fukuda T. A pso-based mobile robot for odor source localization in dynamic advection-diffusion with obstacles environment: theory, simulation and measurement[J]. IEEE Computational Intelligence Magazine, 2007, 2(2):37-51.

[9] Liu Y H, Yang W L. The structure design of an intelligent decision support system for navigation collision avoidance[C]//Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on. IEEE, 2004, 1: 302-307.

[10] Szlapczynski R. Evolutionary sets of safe ship trajectories within traffic separation schemes[J]. Journal of Navigation, 2013, 66(01): 65-81.

[11] Kayano J, Nishizaki C, Imazu H, et al. A study on analysis of characteristics of ships navigators' look-out by using OZT[C]//System of Systems Engineering Conference (SoSE), 2015 10th. IEEE, 2015: 274-279.

[12] Huang S Q, Liu D Z, Gao G Q, et al. A novel method for speckle noise reduction and ship target detection in SAR images[J]. Pattern Recognition, 2009, 42(7):1533-1542.

[13] Lisowski J. Optimal and game control algorithms of ship in collision situations at sea[J]. Control and Cybernetics, 2013, 42(4): 1-20.

[14] Hasegawa K, Fukuto J, Miyake R, et al. An intelligent ship handling simulator with automatic collision avoidance function of target ships[J]. Proc. of INSLC, 2012, 17.

[15] Ding Z G, Hefei. An Adaptive Anti-Collision Algorithm Based on Search Matrix[J]. Pattern Recognition & Artificial Intelligence, 2008, 21(4):476-481.

[16] Xiong M, Lees M, Cai W, et al. Analysis of an efficient rule-based motion planning system for simulating human crowds[J]. Visual Computer, 2010, 26(5):367-383.

[17] Fiorini P, Shiller Z. Motion planning in dynamic environments using velocity obstacles[J]. The International Journal of Robotics Research, 1998, 17(7): 760-772.

[18] Fossen T I. Guidance and control of ocean vehicles[M]. New York: Wiley, 1994.