

Ship RBF neural network sliding mode PID heading control

Jiaming Yu*, Renxiang Bu, and Liangqi Li

Collage of Navigation, Dalian Maritime University, Dalian, China

Abstract. In view of the inherent non-linearity, complexity, susceptibility to external wind, wave, and current interference of under-driven ships, and the difficulty of adjusting and adjusting control parameters, to improve the performance of ship's autopilot, a kind of RBF neural network sliding mode variable structure PID controller is designed. Traditional PID control is sensitive to parameter changes, online tuning is difficult, and easy to overshoot. In order to solve this problem, combining the variable structure characteristics of PID, a differential compensation term is added to the integral term to convert the PID control parameters into three parameters with more obvious physical meanings, and then combined with the RBF neural network learning and identification function to realize online tuning and adaptive control of ship control parameters. Using MATLAB software to simulate the container ship "MV KOTA SEGAR" MMG model shows that the designed RBF neural network sliding mode PID controller can effectively eliminate the ship's lateral deviation caused by external interference such as wind, waves, currents, etc., with high control accuracy, robustness and strong adaptability.

Keywords: Underactuated ship, Heading control, Sliding mode PID differential compensation, RBF neural network.

1 Introduction

Since substantive progress was made in the development of gyro compasses in the early 1920s, ship heading control has always been an important research topic in the field of ship motion control [1]. The original course controller was only simple proportional control, and could only be used for course keeping control with low precision requirements, but it gave people the hope of realizing automatic control in the field of ship maneuvering. At present, there are a variety of methods used in the field of ship control systems, such as PID control, neural network adaptive control, fuzzy control, etc. The use of the above methods is to design a PID autopilot or adaptive autopilot and other products. In terms of PID autopilots used by more than 90% of ships, PID autopilot parameters are simple to adjust and do not depend on the prior knowledge of the controlled object, and its structure is simple and robust. Although PID autopilots have made great progress compared with the previous ones,

* Corresponding author: 767060595@qq.com

conventional PID autopilots still have shortcomings. This is because the design of PID autopilots of ships is based on a definite mathematical model. However, when the ship is sailing at sea, the speed of the ship is , Loading conditions, draught conditions will change the ship's motion state, changes in the external wind, waves, currents and other sea conditions will also cause changes in external interference, resulting in ship control cannot reach the ideal state, while the conventional PID autopilot does not have self-adjusting parameters The ability of the ship cannot adapt to the various changing sea conditions when the ship is sailing. At the same time, it is difficult for the pilot to accurately adjust the PID parameters according to the ship's sailing conditions. Therefore, the performance of the ship's PID autopilot will be greatly affected. In actual navigation, conventional PID autopilots can only adopt the "dead zone" method in the face of real-time changing sea conditions. This method reduces the accuracy of ship control while also increasing energy consumption. Therefore, how to design ship control systems Ship controller, and accurately optimize and adjust its PID parameters, so that the ship achieves the ideal control state will be particularly important. The basic problem of PID control is the contradiction between integral overshoot and steady-state error, and the ship motion state has the characteristics of large time delay, large inertia, and nonlinearity, which makes it more difficult to select the parameters of the ship's PID controller. In recent decades, many PID optimization methods have been proposed, which are mainly divided into two types: one is traditional methods, such as Ziegler-Nichols rule, critical ratio rule, and simplex rule; the other is intelligent algorithm, Such as genetic algorithm, particle swarm algorithm, ant colony algorithm, artificial neural network, fuzzy theory, etc. Traditional methods are usually empirical, and it is difficult to meet the requirements in the face of complex control situations, while intelligent algorithms can better meet the parameter optimization of PID complex control situations. In recent years, intelligent algorithms have been widely used in PID parameter optimization, and certain research results have been obtained. Mao Min [2] and others have used genetic algorithms to optimize PID parameters without the need to differentiate the objective function, but it is complex in optimization. The problem has shortcomings that cannot be ignored, and the basic genetic algorithm has a slow convergence speed and is prone to premature maturity, which leads to a greatly reduced optimization ability. Tan Guanzheng [3], etc., in view of the PID parameter optimization problem, combined with the ant colony algorithm to design an optimal PID controller with incomplete differentiation. Simulation shows that compared with the traditional PID controller design, this method has better control performance and Robust performance can be used to control many different objects and processes. Wang Jingzhi [4] and others used the BP neural network to have the characteristics of arbitrary non-linear expression ability, and proposed a PID control algorithm based on the BP neural network to achieve the best PID control effect through the learning of the system performance. Obviously the PID controller is the core of the entire control system, and its PID parameters play a vital role in its control quality. At present, there are many optimization methods for PID parameter tuning. Both traditional methods and intelligent algorithms are not omnipotent. Each has its own strengths and weaknesses, and further research is needed.

Aiming at the problem of integral overshoot caused by conventional PID control, this paper adds differential compensation to its integral term, and combines the theoretical characteristics of sliding mode variable structure to design a sliding mode PID controller. Aiming at the problem of ship navigation in time-varying sea conditions, the RBF neural network learning algorithm is introduced into the design of the sliding mode PID controller, which eliminates the need to estimate external disturbances such as wind, waves, and currents to achieve adaptive control of under-driven ships. And use MATLAB simulation environment to verify.

2 Basic description

2.1 Ship steering motion equation

When describing the maneuvering movement of a ship, two different coordinate systems are usually used: One is the inertial (fixed) coordinate system $O_E \rightarrow x_E y_E z_E$, used to describe the position of the ship relative to the ground; the other is the attached (on board) coordinate system $o \rightarrow xyz$, the origin of the coordinates is generally at the center of gravity or midship of the ship, the positive x-axis points to the bow, the positive y-axis points to starboard, the positive z-axis points to the bottom keel.

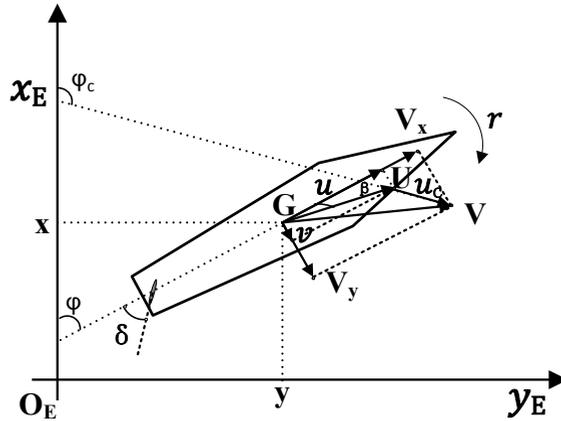


Fig. 1. Schematic diagram of ship plane coordinates.

Choose the center of the ship as the origin of the appendage coordinate system, and the ship's operational motion equation is [5]:

$$\begin{cases} \dot{x} = u \cos \varphi - v \sin \varphi + u_c \cos \varphi_c \\ \dot{y} = u \sin \varphi + v \cos \varphi + u_c \sin \varphi_c \\ \dot{\varphi} = r \\ \dot{r} = (N_s + N_p + N_r + N_e) / (I_{zz} + J_{zz}) \\ \dot{u} = [X_s + X_p + X_r + X_e + (m + m_y)vr] / (m + m_x) \\ \dot{v} = [Y_s + Y_p + Y_r + Y_e - (m + m_x)ur] / (m + m_y) \\ \dot{\delta} = K_E (\delta_r - \delta) / T_E \end{cases} \quad (1)$$

where: x and y are the abscissa and ordinate of the ship in a fixed coordinate system; u and v are the longitudinal speed and lateral speed of the ship in the appendage coordinate system, respectively; φ and r are respectively the heading angle turning speed; φ_c and u_c are the flow direction and velocity of the uniform flow received by the ship; δ_r and δ are the command rudder angle and the actual rudder angle respectively; T_E is the time constant of the steering gear, generally take 2.5~3s; K_E is the steering gear control gain, generally take 1 in experimental research; m , m_x and m_y are the ship's mass, longitudinal additional mass and horizontal additional mass, respectively; I_{zz} and J_{zz} are the ship's moment of inertia and additional moment of inertia, respectively; X and Y are the longitudinal and

transverse forces acting on the hull, respectively; N is the moment about the vertical axis; the subscripts S, P, R and E represent fluid viscosity, propeller, rudder blade and external interference, respectively.

2.2 Control the target

Control the rudder angle δ by adjusting the parameters k_1, k_2, k_3 , make the ship sail on the desired course, that is, the course deviation $\varphi_e \rightarrow 0$.

3 Controller design

3.1 Sliding mode variable structure integral compensation PID control design

The basic form of PID control law is

$$u = -k_d \dot{x}_1 - k_p x_1 - k_i \int x_1 dt \quad (2)$$

In traditional PID control, the integral coefficient k_i is constant, In the actual control process, we hope that when the system deviation is too large, the control effect of the integral should be relatively weakened or even zero. When the system deviation is reduced, the integral control function should be appropriately strengthened at this time, so adjust the integral control reasonably speed according to the system deviation is the key to improving control quality [6]. In order to solve the problem of integral overshoot, a differential term is added to the integral term of PID control for compensation design:

$$u = -k_d \dot{x} - k_p x - k_i \int \left(x + \frac{k_d}{k_p} \dot{x} \right) dt \quad (3)$$

The controller parameters in formula (3) are still three, and while the system tends to be stable, it can effectively offset the integral of the PID integral term system output deviation, and when the system output deviates from the target, it can accelerate the stability of the system.

Using the idea of sliding mode variable structure [7][8][9] to analyze the formula (3), we can get:

$$\begin{aligned} u &= -k_p \left(x + \frac{k_d}{k_p} \dot{x} \right) - k_i \int \left(x + \frac{k_d}{k_p} \dot{x} \right) dt \\ &= -k_d \left(\frac{k_p}{k_d} x + \dot{x} \right) - k_i \frac{k_d}{k_p} \int \left(\frac{k_p}{k_d} x + \dot{x} \right) dt \end{aligned} \quad (4)$$

In formula (4), let: $k_1 = \frac{k_p}{k_d}$, $k_2 = k_d$, $k_3 = k_i \frac{k_d}{k_p}$ get:

$$u = -k_2 (k_1 x + \dot{x}) - k_3 \int (k_1 x + \dot{x}) dt \quad (5)$$

It can be seen from formula (5) that the physical meaning of converting the parameter control of the PID controller to the control of k_1, k_2, k_3 is more obvious, and it is more convenient to optimize its tuning.

3.2 RBF neural network sliding mode PID control design

RBF neural network can approximate arbitrary non-linear functions, can handle difficult-to-analyze regularities in the system, has good generalization ability, and has a fast learning convergence rate [10]. In the RBF network, $x = (x_i)^T$ is the input of the network, the hidden layer output of the network is $h = (h_j)^T$, h_j is the output of the j th neuron in the hidden layer:

$$h_j = \exp\left(-\frac{\|X - C_j\|^2}{2b_j^2}\right) \tag{6}$$

where C_j is the coordinate vector of the center point of the Gaussian function of the j th neuron in the hidden layer, b_j is the width of the Gaussian function of the j th neuron in the hidden layer. The network weight of RBF is w , the RBF network output is $y_m(k) = w^T h$. The performance index function of the identifier is:

$$J = \frac{1}{2}[y(k) - y_m(k)]^2 \tag{7}$$

In the formula, $y(k)$ is the output value of the system identified at time k , and $y_m(k)$ is the output value of the identification network.

In the RBF neural network, in order to realize the tuning of the control system parameters, the internal parameters that can be adjusted are: the center vector value of the Gaussian function, the base width parameter and the output weight value. In this paper, the gradient descent method is used to learn and train the internal parameters of the network. The iterative algorithm of network weights, node center values and base width parameters is as follows [11]:

$$\begin{cases} \Delta w_j = \eta[y(k) - y_m(k)]h_j \\ w_j(k) = w_j(k-1) + \Delta w_j + \alpha[w_j(k-1) - w_j(k-2)] + \beta[w_j(k-2) - w_j(k-3)] \\ \Delta b_j(k) = \eta[y(k) - y_m(k)]w_j h_j \frac{\|X - C_j\|^2}{b_j^3} \\ b_j(k) = b_j(k-1) + \Delta b_j + \alpha[b_j(k-1) - b_j(k-2)] + \beta[b_j(k-2) - b_j(k-3)] \\ \Delta c_{ji}(k) = \eta[y(k) - y_m(k)]w_j h_j \frac{x_j - c_{ji}}{b_j^2} \\ c_{ji}(k) = c_{ji}(k-1) + \Delta c_{ji} + \alpha[c_{ji}(k-1) - c_{ji}(k-2)] + \beta[c_{ji}(k-2) - c_{ji}(k-3)] \end{cases} \tag{8}$$

In the formula, η is learning efficiency, α and β are momentum factors, $\eta, \alpha, \beta \in (0, 1)$.

The Jacobian array reflects the sensitivity of the output of the controlled object to the change of the control quantity. Its value is identified by the RBF network. Its formula is as follows:

$$\frac{\partial y(k)}{\partial \Delta u(k)} \approx \frac{\partial y_m(k)}{\partial \Delta u(k)} = \sum_{j=1}^m w_j h_j \frac{c_{ji} - x_i}{b_j^2} \quad (9)$$

Whether it is k_p, k_i, k_d in traditional PID control or k_1, k_2, k_3 in sliding mode PID control in formula (5), its parameters are relatively difficult to tune in the control system, and traditional empirical methods and trial-and-error methods are far from satisfying the current control needs. In this paper, the RBF neural network identification system is applied to sliding mode PID control to realize online self-tuning of parameters k_1, k_2, k_3 in sliding mode PID control.

RBF neural network tuning sliding mode PID parameter control block diagram is shown in Figure 2:

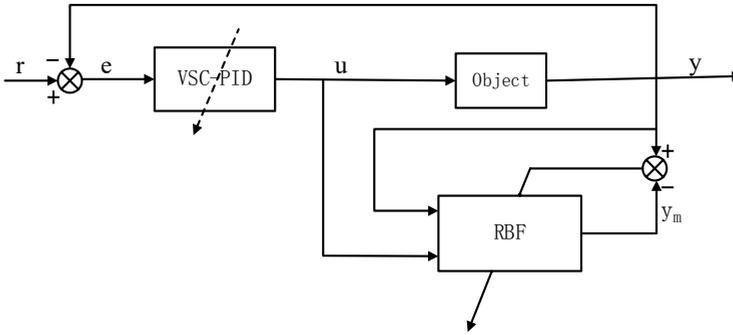


Fig. 2. RBF neural network tuning sliding mode PID control block diagram.

It can be seen from Figure 2 that the input quantity of the system control error signal is:

$$e(k) = y(k) - y_m(k) \quad (10)$$

The input items of the sliding mode PID controller are:

$$\begin{cases} x(1) = e(k) - e(k-1) \\ x(2) = e(k) \\ x(3) = e(k) - 2 \cdot e(k-1) + e(k-2) \end{cases} \quad (11)$$

According to the idea of incremental PID discretization, the sliding mode PID control algorithm is known from formula (5):

$$\begin{cases} u(k) = u(k-1) + \Delta u(k) \\ \Delta u(k) = (k_1 k_2 + k_3)(e(k) - e(k-1)) + k_2 e(k) + k_1 k_3 (e(k) - 2e(k-1) + e(k-2)) \end{cases} \quad (12)$$

The parameters $k_1, k_2,$ and k_3 in the sliding mode PID controller are adjusted by the gradient descent method [12] in the neural network, and the tuning performance indicators are:

$$E(k) = \frac{1}{2} e^2(k) \quad (13)$$

Therefore, the calculation formula for parameter adjustment is:

$$\begin{cases} \Delta k_1 = -\eta \frac{\partial E}{\partial k_1} = -\eta_1 \frac{\partial E}{\partial y(k)} \frac{\partial y(k)}{\Delta u} \frac{\partial u}{\partial k_1} = \eta_1 e(k) \frac{\partial y(k)}{\Delta u} (k_2 x(1) + k_3 x(3)) \\ \Delta k_2 = -\eta \frac{\partial E}{\partial k_2} = -\eta_2 \frac{\partial E}{\partial y(k)} \frac{\partial y(k)}{\Delta u} \frac{\partial u}{\partial k_2} = \eta_2 e(k) \frac{\partial y(k)}{\Delta u} (k_1 x(1) + x(2)) \\ \Delta k_3 = -\eta \frac{\partial E}{\partial k_3} = -\eta_3 \frac{\partial E}{\partial y(k)} \frac{\partial y(k)}{\Delta u} \frac{\partial u}{\partial k_3} = \eta_3 e(k) \frac{\partial y(k)}{\Delta u} (x(1) + k_1 x(3)) \end{cases} \quad (14)$$

In formula (14), η_1, η_2, η_3 are the learning rates of the three parameters of sliding mode PID, $\frac{\partial y(k)}{\Delta u}$ is the Jacobian identification result of the neural network, the calculation process is as formula (9).

3.3 Schematic diagram of controller working principle

The working principle of RBF neural network sliding mode PID controller is shown in Figure 3.

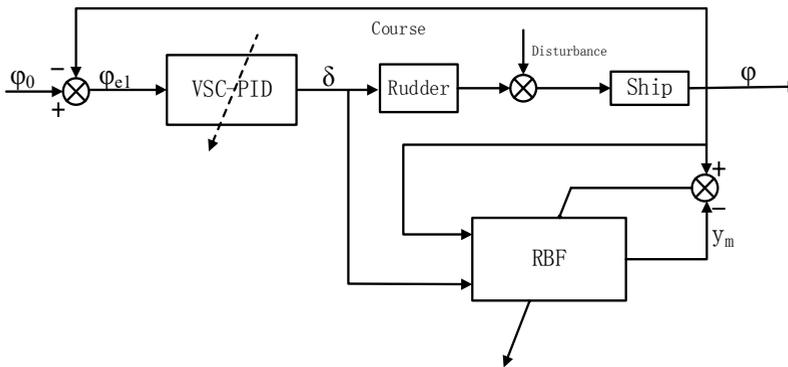


Fig. 3. Schematic diagram of controller working principle.

4 Simulation verification and analysis

In order to verify the effectiveness and feasibility of the designed controller, it is in line with the actual marine navigation conditions of the ship. The container ship "MV KOTA SEGAR" is used as the simulation object. The main parameters of the ship are 217m in length between two columns, 37.3m in width, and 12.5 in full load. m, the displacement is 14635t, and the square coefficient is 0.681 [13].

4.1 Simulation analysis of ship heading control under ideal sea conditions

Under ideal sea conditions without interference from the external environment such as wind, waves, currents, etc., set the ship's initial speed of 19.4 kn, rotation speed of 78.4 rpm, the ship's initial heading to 0° , and the ship's heading setting value of 60° , respectively, using the designed sliding mode PID controller and RBF neural network sliding mode PID controller are simulated, set sliding mode PID controller parameter $k_1 = 0.025, k_2 = 25, k_3 = 3$, set the RBF neural network sliding mode PID controller parameter $\eta = 0.8, \alpha = 0.5, \beta = 0.02$,

$k_1 = 0.025$, $k_2 = 25$, $k_3 = 3$, $\eta_1 = 0.03$, $\eta_2 = 0.09$, $\eta_3 = 0.05$ weight value $\omega_0 = [30, 40, 10]$, the number of hidden layer nodes $n = 6$. The simulation results of the ship are as follows:

4.1.1 Simulation of ship heading control under sliding mode PID controller:

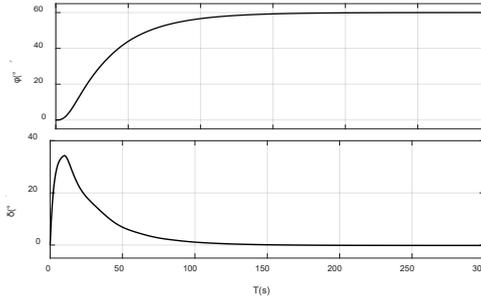


Fig. 4. Simulation diagram of ship heading under ideal sea conditions (sliding mode PID controller).

4.1.2 Ship heading control simulation under RBF neural network sliding mode controller (k_1 value magnification 100 times):

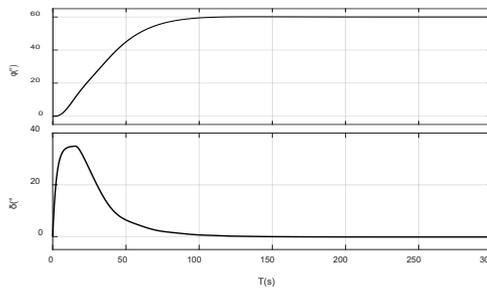


Fig. 5. Simulation diagram of ship heading under ideal sea conditions (RBF neural network sliding mode PID controller).

It can be seen from Figure 4 and Figure 5 that the sliding mode PID controller and RBF neural network sliding mode PID controller designed in this paper can achieve good control under ideal sea conditions without interference from external factors such as wind, waves, currents, etc. The effect is that the rise time is short, and the system basically has no overshoot. When the ship's heading changes from 0° to 60° , the adjustment time is about 150s under the action of the sliding mode PID controller, and about 110s under the action of the RBF neural network sliding mode PID controller. Compared with the two, RBF neural network sliding mode PID has better control effect, and the ship rudder angle changes smoothly under its control, which can effectively reduce the loss of ship steering gear and improve the economy of navigation.

4.2 Simulation analysis of ship heading control in interfering sea state

In order to verify the control performance of the designed ship controller when the ship is disturbed, on the basis of simulation under ideal sea conditions, interference conditions such as steady wind and random wind and waves are added to the simulation model. The ship parameters are set as: initial speed 19.4kn, main engine the rotation speed is 78.4rpm,

the wind speed is 10m/s, the wind direction is 30°, and the interference of sea waves and random wind current is expressed by the second-order oscillation link driven by white noise [14][15]:

$$H(s) = \frac{0.4198s}{s^2 + 0.3638s + 0.3675} \tag{15}$$

The simulation interference is shown in Figure 6:

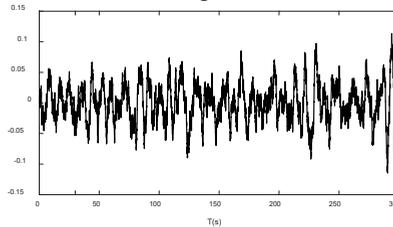


Fig. 6. Wave and random wind current interference diagram.

4.2.1 Simulation of heading control of sliding mode PID controller under disturbing sea conditions:

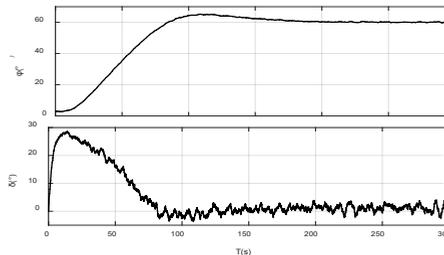


Fig.7. Simulation diagram of ship heading under interference sea conditions (sliding mode PID controller).

4.2.2 RBF neural network sliding mode PID controller heading control simulation under disturbing sea conditions:

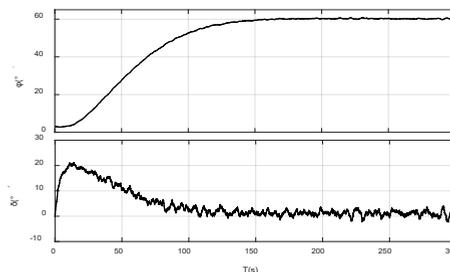


Fig. 8. Simulation diagram of ship heading under interference sea conditions (RBF neural network sliding mode PID controller).

It can be seen from Figure 7 and Figure 8 that the sliding mode PID controller and the RBF neural network PID controller can still achieve better control effects under the interference of steady wind and random wind, waves and currents, which also verifies the designed sliding mode PID control. The device is not sensitive to interference and has

strong robustness. The initial parameters of the two controllers are the same, the ship has a slight overshoot under the action of the sliding mode PID controller, and the adjustment time is about 200s, while the RBF neural network sliding mode PID controller adjusts the initial parameters by self-tuning, and the system adjusts the time It is about 160s and there is no overshoot, so the sliding mode PID controller designed based on RBF neural network has better anti-interference ability, robustness and strong adaptive ability. Finally, in order to suppress the interference of steady wind and current, the output rudder angles of the two controllers are maintained at about 3° to maintain the ship's heading control.

5 Conclusions

In this paper, differential compensation is added to the integral term of traditional PID control, and combined with the variable structure characteristics of PID control, a sliding mode variable structure PID control algorithm is designed. The physical meaning of the control parameters is more obvious, which is convenient for parameter tuning. In the face of the problem of ship control parameter changes in time-varying sea conditions, the RBF neural network learning algorithm is added to the sliding mode variable structure PID control to realize online tuning and adjustment of the control parameters of the ship when sailing under time-varying sea conditions. The MATLAB simulation environment is used to verify the effectiveness of the controller designed in this paper.

References

1. Xu Guoping and Zhang Xianku 2013 Summary of research on ship autopilot (China Shipbuilding) p 191-200
2. Mao Min and Yu Xinning 2002 PID parameter optimization method based on genetic algorithm (China Electric Power) p 51-54
3. Tan Guanzheng and Li Wenbin 2004 Design of optimal PID controller for intelligent artificial legs based on ant colony algorithm (Journal of Central South University) p 91-96
4. Wang Jingzhi , Ren Kaichun and Hu Bin 2011 PID control based on BP neural network tuning [J]. Industrial Control Computer (Industrial Control Computer) p 72-73
5. Jia Xinle and Yang Yansheng 1999 Mathematical model of ship motion: mechanism modeling and identification modeling (Dalian: Dalian Maritime University Press)
6. Zhu Yeqiu 2012 Research on optimization of PID control parameters of ship course (Dalian Maritime University)
7. Bu Renxiang 2008 Research on nonlinear feedback control of underdriven surface ships (Dalian Maritime University)
8. Liu Jinkun 2012 MATLAB simulation of sliding mode variable structure control (Beijing: Tsinghua University Press)
9. Wang Fengyao 1995 Sliding mode variable structure control (Beijing: Mechanical Industry Press)
10. Liu Jinkun 2018 RBF neural network adaptive control and MATLAB simulation (second edition) (Beijing: Tsinghua University Press)
11. Wang Xiaolan 2012 Research on ship motion control and its virtual reality simulation (Dalian Maritime University)
12. Lin Jiayu and Liu ying 2002 Learning step optimization in gradient descent training method of RBF neural network (Signal Processing) p 43-48

13. Sun Wuchen , Bu Renxiang and Liu Yong 2020 Ship PID differential compensation heading control with heel suppression function (Journal of Shanghai Maritime University) p 19-24
14. Feng Yongxiao and Zhang Xianku 2018 Comparison between nonlinear modification and fuzzy control of ship course keeping control (China Navigation) p 12-15
15. Zhang Xianku and Feng Yongxiao 2017 "Yupeng" wheel autopilot control algorithm based on tangent function nonlinear feedback (English) (Journal of Measurement Science and Instrumentation) p 78-83