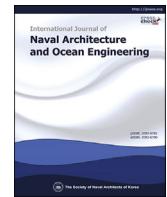




Contents lists available at ScienceDirect

## International Journal of Naval Architecture and Ocean Engineering

journal homepage: <http://www.journals.elsevier.com/international-journal-of-naval-architecture-and-ocean-engineering/>



# Data-driven model identification and predictive control for path-following of underactuated ships with unknown dynamics

Le Wang <sup>a</sup>, Shijie Li <sup>a</sup>, Jialun Liu <sup>b, c, \*</sup>, Qing Wu <sup>a</sup>

<sup>a</sup> School of Transportation and Logistics Engineering, Wuhan University of Technology, Wuhan, PR China

<sup>b</sup> Intelligent Transportation Systems Research Center, Wuhan University of Technology, Wuhan, PR China

<sup>c</sup> National Engineering Research Center for Water Transport Safety, Wuhan, PR China



### ARTICLE INFO

#### Article history:

Received 7 September 2021

Received in revised form

27 February 2022

Accepted 28 February 2022

Available online 10 March 2022

#### Keywords:

Model identification

AMBS-P

Elastic net regression

Path following

### ABSTRACT

With the trends towards autonomous shipping, advanced ship motion control methods have received increased attention in recent years. The validity of ship models is crucial in designing motion controllers and directly affects their performances. However, accurate models that could reflect true ship dynamics are highly nonlinear, complex and complicated to identify, especially in situations when the experimental conditions are limited. This paper proposes a data-driven predictive control method for path-following of under-actuated cargo ships with unknown dynamics, which makes use of data gathered during operation to improve the model and the path-following performance. Based on the ship navigation data set, the relations between the heading angle and the rudder angle of the ship are fitted with seven typical regression algorithms, which acts as the prediction model in the path-following controller. Simulation study is carried out to choose the most suitable regression algorithm, among which elastic net regression is selected. The Antenna Mutation Beetle Swarm Predictive (AMBS-P) algorithm is introduced to find the optimal weights in the model identification process. A Line-of-Sight (LOS) algorithm is used as the guidance law to transform reference way-points into reference heading angles, and the path-following controller is designed also based on and the AMBS-P algorithm. Simulation results show that the proposed data-driven control method performs well in the path-following task without having prior knowledge regarding the hydrodynamic coefficients and ship parameters.

© 2022 Society of Naval Architects of Korea. Production and hosting by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

## 1. Introduction

With the trends towards autonomous shipping, advanced ship motion control techniques are being developed to ensure that ships can independently control their own action. Surface ships can be divided into fully-actuated ships and under-actuated ships, which can be distinguished by whether the control input dimension is equal to the degree of freedom of the ship. Most marine ships are under-actuated systems, only the yaw and surge movements can be actuated directly, which means that there are more variables to be controlled than the number of control actuators.

While the validity of ship models is crucial in designing motion controllers and directly affects their performances, accurate models

that could reflect true ship dynamics are highly nonlinear, complex and complicated to identify, especially in situations when the experimental conditions are limited. Data-driven control provides another way to deal with situations in which the system model of the ship are unknown or difficult to obtain, which uses on-line or off-line data to achieve optimal control performance. In this paper, an integrated method that combines model identification with a predictive control scheme is proposed for the path-following control problem of underactuated ships with unknown ship dynamics.

Model identification determines a model equivalent to the measured system from a set of given data, and machine learning algorithms such as least squares method, neural network, support vector machine (SVM), Gaussian process regression, have been applied in identifying the key parameters of ships. The least square methods have the characteristics of simplicity and strong universality, and have been used in ship model parameter identification (Koyama et al., 1973; Zhang et al., 2015). In Zhang et al. (2015), the transformed multi innovation least squares (TMILS) algorithm for the 4°-of-freedom ship maneuvering model identification. Neural

\* Corresponding author.

E-mail addresses: [wanglexile@whut.edu.cn](mailto:wanglexile@whut.edu.cn) (L. Wang), [lishjie@whut.edu.cn](mailto:lishjie@whut.edu.cn) (S. Li), [jialunliu@whut.edu.cn](mailto:jialunliu@whut.edu.cn) (J. Liu), [wq@whut.edu.cn](mailto:wq@whut.edu.cn) (Q. Wu).

Peer review under responsibility of The Society of Naval Architects of Korea.

network is used to estimate the dynamic characteristics of ship propeller and deal with its hydrodynamic uncertainty, so as to implement the motion control and trajectory tracking of ship (Haseltalab and Negenborn, 2019). To estimate the shaft power and main engine fuel consumption of container ships sailing under any conditions, a multi-layer neural network algorithm is used to train the model of ship fuel consumption and speed loss based on the high sampling frequency data sets (Karagiannidis and Themelis, 2021). Compared with neural network, support-vector-machine (SVM) has the advantages of high generalization performance, global optimal solution and overcoming “dimension disaster”. To determine the parameters in SVM, meta-heuristic optimization algorithm can be introduced. In Zhu et al. (2017), the artificial bee colony algorithm is used to optimize the super parameters in SVM, so as to identify a simplified model of a cargo ship based on the simulation data of ship rudder and heading angles. The  $\epsilon$ -SVM is applied to identify the ship maneuvering motion with the Abkowitz model using the training data without noise (Zhang and Zou, 2011). The  $\epsilon$ -SVM have also been used in Wang et al. (2015) to tests the prediction ability of white-box modeling, gray-box modeling and black-box modeling of ship maneuvering motion. In order to reduce the problem of parameter drift or overfitting caused by noise on the reliability of the model, based on  $\epsilon$ -SVM,  $\nu$  (“Nu”)-SVM is designed to identify ship maneuvering model (Wang et al., 2019b). Gaussian process regression has been used in Xue et al. (2020) to identify the ship dynamic model with input noise.

The identified system model is usually combined with a predictive control scheme to design controllers. In Ugo et al. (2018), a data-driven stochastic model predictive control strategy is established, in which the system disturbance is identified with Gaussian process regression, so as to improve the vision-based mobile robot path tracking. This could reduce the demand for accurate process model and specific parameter observer, while maintaining the advantages of multi-point control such as predictive behavior and constraint processing (Ostafew et al., 2016). Gaussian process of off-line data and extended Kalman filter are used to improve the model of a robotic arm, which is combined with model predictive control to improve the trajectory tracking performance (Caron et al., 2019). In Zhu et al. (2018), a cloud model based on probability theory and fuzzy mathematics is used to design a ship course-keeping controller, in which the optimized SVM is used to identify the ship dynamics.

This paper investigates and compares the model identification performances of seven regression algorithms, and finally selects the elastic network regression algorithm, which is combined with a heuristics-based control algorithm to realize ship path-following control. Meta-heuristic optimization algorithm is often used in ship controller parameter optimization and ship model parameter identification (Wang and Z, 2012; Shin et al., 2017). Based on the requirement of fast and accurate convergence, this paper chooses a bio-based metaheuristic algorithm, namely, the beetle antenna search (BAS) algorithm (Jiang and Li, 2017). We introduced BAS into mutation operator to obtain adaptive mutation beetle particle swarm (AMBPS) algorithm, which is combined with PID to adjust ship control parameters (Wang, Wu, Liu, Li and Negenborn, 2019a). In this paper, we add prediction idea to AMBPS algorithm to design an antenna mutation beetle swarm predictive (AMBS-P) algorithm, such that AMBS-P can be directly used as a control algorithm. It has been used to find the optimal weights in the model identification process, as well as been to design the path-following controller. A line-of-sight (LOS) algorithm is used as the guidance law to transform reference way-points into reference heading angles.

Experimental results show that the proposed data-driven control method performs well in the path-following task without having prior knowledge regarding the hydrodynamic coefficients and ship parameters.

The main contributions of this paper are as follows:

- The proposed data-driven control strategy allows the users to exploit the approximate model knowledge captured from data without having prior knowledge regarding the hydrodynamic coefficients and ship parameters, as well as exploiting the advantage of the predictive control scheme that it can be integrated with meta-heuristics optimization algorithms in controller parameter selections.
- A quantitative analysis of the model identification performances of seven typical regression algorithms is carried out, which could provide guidance for practitioners in selecting the most suitable algorithms.
- The Antenna Mutation Beetle Swarm Predictive (AMBS-P) algorithm is proposed and applied in finding the optimal weights in the model identification process and in designing the path-following controller with a LOS guidance principle. This could save the efforts spend in weight selection and controller parameter adjustments. It is also fast and easy to implement, which could facilitate its use in practice. Simulation results show that compared with PID control, the heading tracking error is reduced by 2.02% and the path following error is reduced by 2.98%.

The rest of the paper is organized as follows. Section 2 introduces the characteristics of ship dynamics, the basic principles of regression algorithms and the AMBS-P algorithm. Section 3 describes the model identification steps of ship dynamics, upon which the prediction model of ship motion based on elastic net regression is formulated. Section 4 gives the controller design steps based on the prediction model, in which AMBS-P algorithm is introduced to ship model weight parameters search and motion control. Section 5 presents simulation results, in which the model identification performances of different regression algorithms are compared and analyzed. The path-following performance is also evaluated. Section 6 summarizes the research results and gives future research prospects.

## 2. Theoretical background

This section introduces the basic characteristics of ship motion, the principles of regression analysis, as well as the AMBS-P algorithm.

### 2.1. Basic ship dynamics

Fig. 1 illustrates the space-fixed coordinate system  $o_0 - x_0y_0z_0$  and the moving ship fixed coordinate system  $o - xyz$  used in this paper. The fixed coordinate system is set at the center of gravity of the ship. Among them,  $\psi$  is defined as the angle between  $x_0$  and  $x$  axes, which is referred as heading angle.  $U$  is the total velocity.  $\delta$  represents rudder angle. Symbols  $x$  and  $y$  represent the ship positions along axis  $x$  and  $y$ , respectively.  $u$  and  $v$  denote the velocity components in  $x$  and  $y$  directions respectively.  $r$  is the yaw rate.

The establishment of ship motion model is crucial for effective controller design. To build an accurate model of ship motion, implicit hydrodynamic parameters need to be identified. A three degrees of freedom (3-DOF) ship model is usually expressed as Eq. (1):

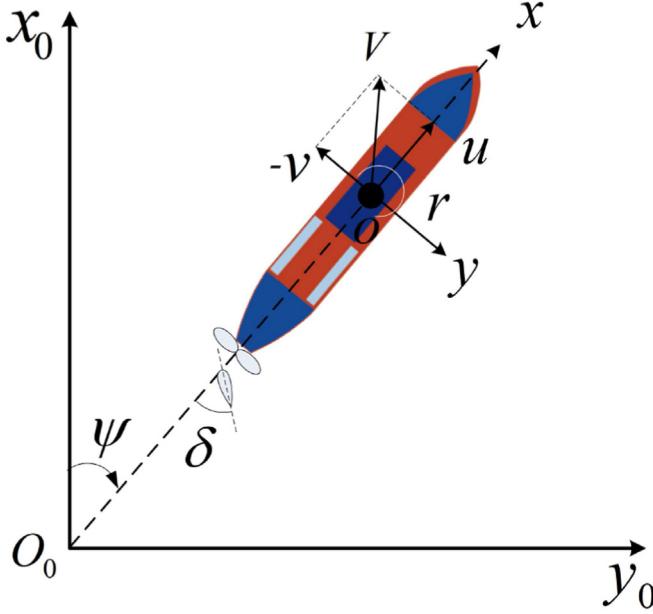


Fig. 1. Ship and coordinate systems.

$$\begin{cases} (m + m_x)\dot{u} - (m + m_y)v - x_Gmr^2 = X_H + X_P + X_R, \\ (m + m_y)\dot{v} + (m + m_x)ur + x_Gmr = Y_H + Y_P + Y_R, \\ (I_z + x_G^2m + J_z)\dot{r} + x_Gm(\dot{v} + ur) = N_H + N_P + N_R, \end{cases} \quad (1)$$

where  $H$ ,  $R$  and  $P$  represent the hull, rudder and propeller, respectively.  $m$ ,  $m_x$  and  $m_y$  are ship mass, added mass in  $x$ -direction, and added mass in  $y$ -direction.  $u$  and  $v$  are ship longitudinal and lateral speed,  $r$  is ship yaw rate around midship.  $I_z$  and  $J_z$  are moment of inertia and added moment of inertia around the  $z$ -axis.

To simplify model (1), this paper mainly considers the yaw movement, in which the evolution of ship heading can be expressed in Eq. (2) (Li et al., 2020):

$$\begin{cases} \dot{\psi} = r, \\ \dot{r} = \frac{m_u - m_v}{m_r}uv - \frac{f_r(v)}{m_r} + \frac{F_r(\cdot)}{m_r}\delta + \frac{t_{dr}}{m_r}, \end{cases} \quad (2)$$

where  $m_u$  and  $m_v$  are the sum of ship mass and added mass in  $x$ -axis and  $y$ -axis, respectively.  $m_r$  is the sum of moment of inertia and added moment of inertia around the  $z$ -axis.  $f_r(v)$  represent the high-order fluid dynamics item.  $F_r(\cdot)$  is the yaw moment, which are generated by the rudder angle  $\delta$ .  $t_{dr}$  represent the disturbances on the  $z$ -axis.

However, in the case of limited measurement tools and conditions,  $f_r(v)$  and  $F_r(\cdot)$  are unknown. Therefore, data-driven modeling fitting methods provide another effective way to approximate ship motion characteristics. This paper adopts regression techniques to construct the prediction model of ship dynamics, which can be expressed as follows.

$$S_{STATE}(\lambda + 1) = f(S_{STATE}(\lambda), \delta_E(\lambda + 1)), \quad (3)$$

where  $\delta_E$  is defined as rudder angle command,  $S_{STATE}$  is the ship state.  $\lambda$  is the step.

## 2.2. Elastic net regression

Regression analysis is a statistical analysis method used to determine the quantitative relationship between two or more variables. The purpose of the regression model is to find the fitting curve which minimizes the sampled variance. Although this algorithm is simple, there may be overfitting phenomenon.

The loss function of a general regression algorithm is defined as:

$$L(\mathbf{W}) = \arg \min_{\mathbf{W}} \|\mathbf{XW} - \mathbf{Y}\|_2^2, \quad (4)$$

where  $\mathbf{W}$  is the partial regression coefficient,  $\mathbf{X}$  represents the system input, and  $\mathbf{Y}$  represents the system output. The commonly used methods to minimize the loss function parameter include gradient descent method and least square method. These methods may lead to overfitting or under fitting, therefore lasso regression, ridge regression and elastic net regression are developed. The difference between ridge regression and general linear regression is that an  $L2$  regularized term is added to the loss function.

The expression of the loss function of ridge regression (Hoerl and Hoerl, 1962) is given as:

$$L(\mathbf{W}) = \arg \min_{\mathbf{W}} \|\mathbf{Y} - \mathbf{XW}\|_2^2 + \alpha \|\mathbf{W}\|_2^2, \quad (5)$$

in which  $\alpha$  is a constant coefficient, representing the regularization coefficient, which needs to be tuned. If  $\alpha$  is too small, it will lose the ability to deal with overfitting. If  $\alpha$  is too large, it will appear under fitting.  $\|\mathbf{W}\|_2^2$  is  $L2$  norm.

Ridge regression (Hui and Hastie, 2004) reduces the regression coefficient without abandoning any feature, making the model relatively stable. It is simple to solve, and the least square method is usually used.

Lasso regression uses  $L1$  regularization, and its loss function is formulated using the following equation:

$$L(\mathbf{W}) = \arg \min_{\mathbf{W}} \|\mathbf{Y} - \mathbf{XW}\|_2^2 + \alpha \|\mathbf{W}\|_1, \quad (6)$$

where  $\|\mathbf{W}\|_1$  is  $L1$  norm.

Lasso regression makes some coefficients smaller, or even some coefficients with smaller absolute values become zero directly. It is especially suitable for parameter number reduction and parameter selection, so it is used to estimate sparse parameters in linear models. But the problem of lasso regression is that its loss function is not continuously differentiable. Because  $L1$  norm uses the sum of absolute values, the loss function has non-differentiable points.

Elastic net regression (Hui and Hastie, 2004) combines the advantages of ridge regression and lasso regression. The model obtained by this method is as sparse as lasso regression. It also has the same regularization ability as ridge regression. Elastic net regression is very useful when many features are interconnected. Lasso may only randomly consider one of these features, while elastic net regression prefer to choose two. The loss function expression of elastic net regression is given as:

$$\begin{aligned} L(\mathbf{W}) = \arg \min_{\mathbf{W}} & \|\mathbf{XW} - \mathbf{Y}\|_2^2 + (1 - R_w)\alpha \|\mathbf{W}\|_2^2 \\ & + R_w\alpha \|\mathbf{W}\|_1, \end{aligned} \quad (7)$$

in which,  $R_w$  is the regression weight of elastic net regression.

## 2.3. AMBS-P algorithm

To determine the weights in Eq. (7), AMBS-P algorithm is used. The steps of this algorithm is as follows.

Suppose there are  $n$  beetles  $B = (B_1, B_2, \dots, B_n)$  in  $N$  dimensional space. The position of the  $i - th$  beetle is described as  $B_i = (b_{i1}, b_{i2}, \dots, b_{iN})$  and the velocity is expressed as  $V_i = (v_{i1}, v_{i2}, \dots, v_{iN})$ . The individual and global extreme points of beetles are respectively assumed as:  $Pb_i = (Pb_{i1}, Pb_{i2}, \dots, Pb_{iN})$  and  $Gb_i = (Gb_{i1}, Gb_{i2}, \dots, Gb_{iN})$ . The individual extreme point  $Pb$  refers to the optimal value in the search process, and the global extreme point  $Gb$  refers to the optimal value obtained by comparing multiple individual extreme points after the end of this search.

So the velocity  $V_i$  and original position  $B_i$  of the beetle are as Eq. (8a) and Eq. (8b).

$$\begin{aligned} V_i^k &= \omega^k \cdot V_i^{k-1} + c_1 \cdot r_1 \cdot (Pb_i^{k-1} - B_i^{k-1}) \\ &\quad + c_2 \cdot r_2 \cdot (Gb_i^{k-1} - B_i^{k-1}), \end{aligned} \quad (8a)$$

$$B_i^k = B_i^{k-1} + V_i^k. \quad (8b)$$

Among them,  $k$  is the current number of iterations.  $c_1$  and  $c_2$  are location coefficients,  $r_1$  and  $r_2$  are 0–1 random variables. The “Linearly Decreasing Weight (LDW)” is used to set  $\omega$ , which makes adaptive adjustment according to the number of iterations, that is,

$$\omega_l^k = \omega_{l\min} + (\omega_{l\max} - \omega_{l\min}) \cdot (K - k) / K, \quad (9)$$

where,  $\omega_{l\max}$  and  $\omega_{l\min}$  are the maximum and minimum weight coefficients,  $K$  represents the maximum number of iterations.

The antennae of beetle is considered as two optional directions. The motion path can be further determined according to this. In this paper,  $B_L$  and  $B_R$  are defined as the coordinates of left and right antennae respectively, shown in Eq. (10). Besides,  $D_0$  is initial antennae length.

$$\begin{cases} B_{Li}^k = B_i^k + D_0^k \cdot \tilde{D}^k / 2, \\ B_{Ri}^k = B_i^k - D_0^k \cdot \tilde{D}^k / 2. \end{cases} \quad (10)$$

Antennae orientation can be described as  $Dir^k = randn(N, 1)$ , and define  $\tilde{D}^k$  as:

$$\tilde{D}^k = Dir^k / norm(Dir^k). \quad (11)$$

$B_i$  represents the beetle centroid coordinates. The calculation formula of  $B_i$  is shown in Eq. (12), where  $L_{step}$  is beetle step length. Then the cost function value  $F(B_i^k)$  is calculated according to  $B_i$  as the following:

$$B_i^k = B_i^{k-1} - L_{step}^k \cdot \tilde{D}^k \cdot sign(F(B_{Li}^k) - F(B_{Ri}^k)). \quad (12)$$

The optimal cost function is calculated based on the  $B_i^k$  in each iteration. Then the current  $Pb_i^k$  and  $F_{best}$  are compared and updated according to Eq. (13).

$$\begin{cases} F_{best}^k = F(B_i^k) \\ Pb_i^k = B_i^k \end{cases}, F(B_i^k) < F_{best}. \quad (13)$$

Update  $L_{step}$  and  $D_0$ .  $L_{step}$  and  $D_0$  require to be updated in time, and proportion control method is selected. Proportional control means that the current updated value is the multiple of the previous step. The condition for  $L_{step}$  attenuation is that the optimal

fitness value does not decrease. This means that the current  $L_{step}$  and  $D_0$  can no longer obtain a better position. They are required to be updated until the set optimal cost function value is met or the control is closed. On the contrary, only the  $B_i^k$  needs to be updated. Update rules are shown in Eq. (14).

$$\begin{cases} L_{step}^k = bl_1 \cdot L_{step}^{k-1} \\ D_0^k = L_{step}^{k-1} / bl_2 \end{cases}, F(B_i^k) < F_{best}. \quad (14)$$

The prediction part is introduced, and the total cost function is set as  $F$ .

$$\begin{aligned} F(B|j) &= \sum_{j=1}^{N_p} q_j \cdot F_{best}^j, \\ &= \sum_{j=1}^{N_p} q_j \cdot (R_{ref}^j - S_{state}^j), \\ &= Q \cdot (R_{REF} - S_{STATE}), \end{aligned} \quad (15)$$

where  $Q$  is the total weight factor,  $Q = diag[q_1, q_2, \dots, q_{N_p}]$ ,  $q$  is the weight factor.  $N_p$  is the number of prediction steps and  $F_{best}$  is the single step fitness function value.  $R_{REF}$  is target state,  $R_{REF} = [R_{ref}(j), R_{ref}(j+1), \dots, R_{ref}(j+N_p)]^T$ .  $S_{STATE}$  is  $j - th$  state,  $S_{STATE} = [S_{state}(j), S_{state}(j+1), \dots, S_{state}(j+N_p)]^T$ .

With the progress of iteration, the concentration degree of beetles increased, and the difference between them decreased gradually. The algorithm may find the global optimal value, or fall into the local optimum. To further avoid local optimum, this paper first calculate the variance of the average cost function and the spatial location aggregation degree of the beetle swarm, and then judge whether the threshold selection needs adaptive mutation to increase the diversity of beetle swarm.

At the  $k$  iteration,  $F_i^k = F(B_i^k)$  is the cost function, and  $F_{ave}^k$  is the average cost function. The fitness variance  $\Theta^k$  is used to reflect the aggregation degree of beetle swarm, as shown Eq. (16).

$$\Theta^k = \sum_{i=1}^n \left( \frac{F_i^k - F_{ave}^k}{\bar{F}^k} \right)^2, \quad (16)$$

where  $\bar{F}^k = \max\{\max\{|F_i^k - F_{ave}^k|\}, 1\}, i \in [1, n]$ .

Set the  $\Phi^k$  to indicate the spatial location aggregation degree between beetle individuals, as described in Eq. (17). The smaller  $\Phi^k$  indicates that the beetles are more concentrated and the higher the probability of variation.

$$\Phi^k = \frac{\max_{1 \leq i \leq n} \{ \|B_i^k - Pb_i^k\| \}}{\max_k \left\{ \max_{1 \leq i \leq n} \{ \|B_i^k - Pb_i^k\| \} \right\}}. \quad (17)$$

Define mutation probability  $p_m^k$  as Eq. (18). The algorithm may fall into local convergence when  $\Theta^k$  is less than the set threshold and  $\Phi^k$  decreases, and Eq. (18) needs to be executed.

$$p_m^k = \begin{cases} e^{-\Phi^k} / 5, & \Theta^k < \Theta_0, \\ 0, & \Theta^k \geq \Theta_0. \end{cases} \quad (18)$$

$\Theta_0$  is a threshold set according to the actual research object.

Finally, further judgment is made according to this probability, and then the individual extreme value of each beetle is adjusted, such as Eq. (19). Among them,  $p_{rand}$  is a random number, and  $\eta$  is a random vector obeying the standard normal distribution.

$$Pb_i^{k+1} = \begin{cases} Pb_i^k \cdot (1 + 0.5\eta), & p_{randn}^k < p_m^k, \\ Pb_i^k, & p_{randn}^k \geq p_m^k. \end{cases} \quad (19)$$

The pseudo-code of whole algorithm is proposed as [Algorithm 1](#).

#### Algorithm 1

AMBS-P algorithm.

---

```

1 Initialize:  $B(0), V(0), D_0^0, L_{step}^0, bl_1, bl_2, Pb^0 =$ 
    $Gb^0 = B(0), F_{best} = F(B(0)) ;$ 
2 while  $k < K$  do
3   for  $i = 1 \rightarrow n$  do
4     Calculate  $\omega^k$  according to Eq. (9);
5     Update the  $V_i^k$  and  $B_i^k$  of beetle according to
       Eq. (8a) and 8b;
6     Generate  $\tilde{D}^k$  according to Eq. (11);
7     Calculate  $B_{Li}^k$  and  $B_{Ri}^k$  according to Eq. (10);
8     Calculate  $F(B_{Li}^k)$  and  $F(B_{Ri}^k)$  according to
       Eq. (15);
9     According to Eq. (12), calculate the  $B_i^{k+1}$ ;
10    According to Eq. (10), update the  $F(B_i^{k+1})$ ;
11    if  $F(B_i^{k+1}) < F_{best}(B_i^*)$  then
12       $Pb_i^{k+1} = B_i^{k+1};$ 
13       $F_{best}(B_i^*) = F(B_i^{k+1});$ 
14    Update the  $D_0^k$  and  $L_{step}^k$  of beetle according to
       Eq. (14);
15     $Gb^k = \min(Pb^k);$ 
16     $F_{best} = \min(F_{best}(B^*(k)));$ 
17    Calculate  $\Theta$  and  $\Phi$  separately according to
       Eq. (16) and (17);
18    Calculate  $p_m^k$  according to Eq. (18);
19    Update the  $Pb^k$  according to Eq. (19) .

```

---

### 3. Model identification steps

[Fig. 2](#) gives proposed model identification scheme, which including data acquisition, rolling fitting, and weight adjustments. Firstly, the target ship is selected and the training data is collected. Secondly, the target ship model is fitted based on elastic net regression model, in which the weight of the model is obtained via AMBS-P algorithm. Then, combined with the  $S_{STATE}$  of each step and the rudder angle command obtained according to the control target  $R_{ref}$ , the training model is imported to get prediction of the next step. As environmental interference may affect the accuracy of training data and ship state prediction, a rolling fitting scheme is proposed.

### 3.1. Training data and testing data acquisition

This paper deals with unknown ship dynamics. Therefore, a model-free controller is required to generate training data. A PID controller is designed and applied to the target ship to track the reference path, as given in:

$$\delta_E(\lambda + 1) = K_p \psi_{error}(\lambda) + K_i \sum_{j=1}^n \psi_{error}(\lambda) + K_d (\psi_{error}(\lambda) - \psi_{error}(\lambda - 1)). \quad (20)$$

As mentioned in Eq. (1), the rudder angle is directly related to the change of the ship heading angle. The input of the sample data is  $[\psi(\lambda), \delta(\lambda)]$  and the output is  $\psi(\lambda + 1)$  with a sampling interval of 1 s. To ensure the convergence and stability of target heading angle  $\psi_{ref}$ , the steps of each target heading are 200. The range of heading angle is  $[0^\circ, 360^\circ]$ , with 72000 steps of data for training, which is recorded as “A-train-data”. The rudder angle range is  $[-35^\circ, 35^\circ]$ . In order to avoid control instability caused by sudden change of heading angle, the second-order filter is used to soften the sharply changing target value heading angle  $\psi_{ref}$ . The second-order filter can be expressed as:

$$\frac{\psi_d}{\psi_{ref}} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}, \quad (21)$$

where  $\xi$  and  $\omega_n$  is the design parameter of the filter.  $\psi_d$  is the output heading of the filter. Then randomly arrange the order of reference heading within  $[0^\circ, 360^\circ]$  degrees to obtain the training data, as shown in [Fig. 3](#).

Randomly arrange the target heading within  $[0^\circ, 360^\circ]$  degrees once again. According to the principle of heading change with the rule of minimum angle rotation, the angle will appear within  $[-360^\circ, 0^\circ]$ . Therefore, randomly select any heading within  $[0^\circ, 360^\circ]$  and  $[-360^\circ, 0^\circ]$  as testing data set “A-test-data-a” and “A-test-data-b” respectively. The target heading is obtained as shown in [Fig. 4](#).

### 3.2. Model training

The elastic net regression algorithm is used to train the data collected by the ship based on Eq. (7), in which:  $\mathbf{X} = \mathbf{X}_{train} = \sum_{\lambda=1}^{\Lambda} [\psi(\lambda), \delta_E(\lambda + 1)]$ ,

$$\mathbf{Y} = \mathbf{Y}_{train} = \sum_{\lambda=1}^{\Lambda} \psi(\lambda + 1).$$

The partial derivative of (7) is obtained:

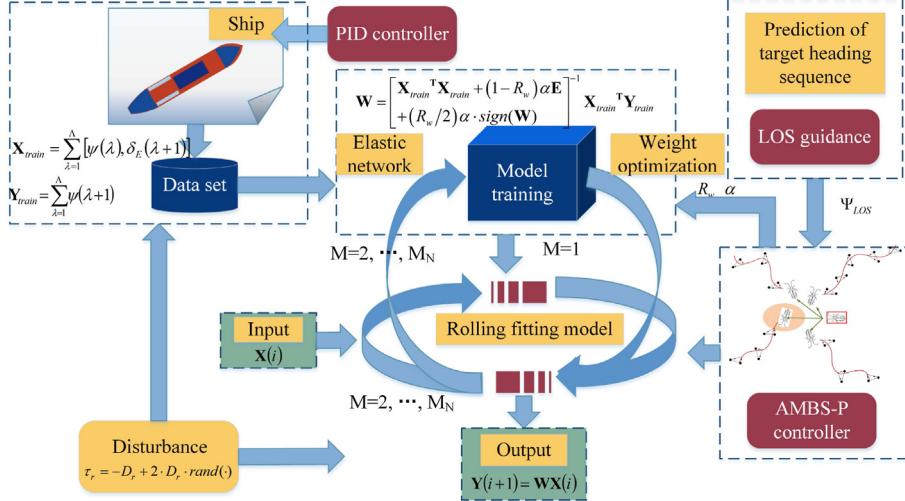
$$\mathbf{W} = \left[ \mathbf{X}^T \mathbf{X} + (1 - R_w) \alpha \mathbf{E} + \frac{R_w}{2} \alpha \cdot \text{sign}(\mathbf{W}) \right]^{-1} \mathbf{X}^T \mathbf{Y}, \quad (22)$$

where,  $E$  is the identity matrix. The optimal weights  $\alpha$  and  $R_w$  are solved iteratively by AMBS algorithm. Then the ship prediction model can be expressed as:

$$\mathbf{Y}(\lambda + 1) = \mathbf{W} \mathbf{X}(\lambda), \quad (23)$$

### 3.3. Disturbance data set

In order to verify the applicability of the control algorithm, the wind wave interference are added to obtain the training data with



**Fig. 2.** The proposed AMBS-P algorithm and model identification scheme.

disturbance. The first-order Markov process is used to describe the external low-frequency disturbance suffered by the ship:

$$\begin{cases} \tau_r = \mathbf{T}^T(\psi)d_r, \\ d_r = -T_r^{-1}d_r + \gamma_r G_{rs}, \end{cases} \quad (24)$$

where,  $\tau_r$  is the disturbance error sequence generated for the environment.  $d_r = -T_r^{-1}d_r + \gamma_r G_{rs}$  is the first-order Markov process of training disturbance data.  $T_r$  is a diagonal constant matrix of disturbance data.  $\gamma_r$  is a diagonal matrix scaling the amplitude of  $G_{rs}$ .  $G_{rs}$  is a vector of zero-mean Gaussian white noises. Since the model used in this paper is a one-dimensional “rudder-heading” model, Eq. (24) is simplified to the disturbance of heading control, taken as:  $\tau_r = \int_{\lambda_t}^{\lambda_{t+1}} (-T_r^{-1}d_r + \gamma_r G_{rs})$ . Here,  $\lambda_t$  is the sampling time.  $d_r(0) = 0.1$ ,  $T_r = 2D_r$ ,  $\gamma_r = D_r$ ,  $D_r$  is disturbance amplitude coefficient of train data. The results of training data are shown in Fig. 5, recorded as “A-train-data-dis”.

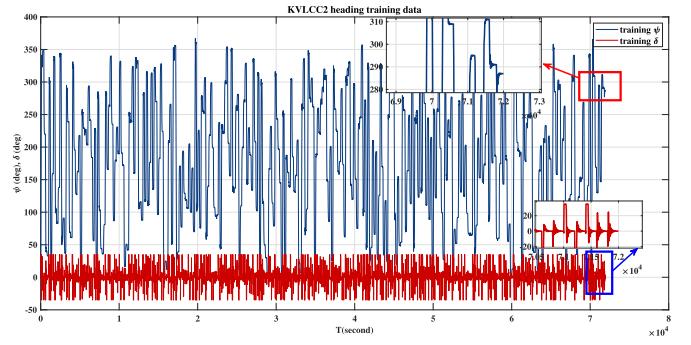
#### 4. Design steps of ship path-following controller

In this section, a path-following controller is designed, in which a line-of-sight (LOS) algorithm is introduced as the guidance law, and the identified data-driven model acts as the predictor of ship dynamics. This part introduces the process of LOS navigation, and proposes the design of path-following controller based on AMBS-P algorithm.

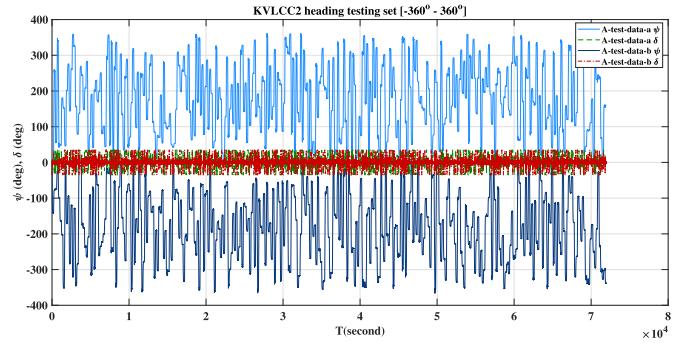
##### 4.1. Principle of LOS navigation

LOS navigation is a method to simulate the sight of experienced sailors to achieve the desired path (Fossen et al., 2015). According to the principle of LOS navigation, if the ship can gradually track the LOS angle, it could successfully track the target path. The principle of LOS navigation is shown in Fig. 6.

As shown in Fig. 6, path points  $P_p$  ( $p = 0, 1, 2, \dots$ ) constitute the desired path, and the line between two adjacent points  $P_p P_{p+1}$  is a straight line. Among them, the straight line formed by  $P_p (y_p, x_p)$



**Fig. 3.** Ship motion control training data (“A-train-data”).



**Fig. 4.** Ship motion control testing data (“A-test-data-a” and “A-test-data-b”).

$P_{p+1}(y_{p+1}, x_{p+1})$  is the current desired path.  $P_{los}(y_{los}, x_{los})$  is the closest point between the LOS circle and the next point in the intersection of the target path.  $\psi_{los}$  is defined as an angle within  $[0^\circ, 360^\circ]$  degrees with the northward direction as the target point. It can be seen that no matter where the ship is currently located in  $P_s(y_s, x_s)$ , the solution formula of  $P_{los}(\lambda)(y_{los}(\lambda), x_{los}(\lambda))$  is as Eq. (25a). Then the calculation method of  $\psi_{los}(\lambda)$  is shown in Eq. (25b).

$$P_{los} = \begin{cases} \frac{x_{los}(\lambda) - x_p}{y_{los}(\lambda) - y_p} = \frac{x_{p+1} - x_p}{y_{p+1} - y_p}, \\ (y_{los}(\lambda) - y_s(\lambda))^2 + (x_{los}(\lambda) - x_s(\lambda))^2 = R_{AC}^2 \end{cases}, \quad (25a)$$

$$\psi_{los}(\lambda) = \arctan \frac{y_{los}(\lambda) - y_s(\lambda)}{x_{los}(\lambda) - x_s(\lambda)}. \quad (25b)$$

Besides, it can be seen from Fig. 6 that there is one or two intersection points, and there may be no real solution when the ship deviates from the desired path. To avoid this,  $R_{los}$  is usually set to  $\zeta$  times of captain  $L_{pp}$ . In order to ensure the adaptability of ship path-following, LOS circle is set as a dynamic circle.  $R_{los}$  can be defined as:

$$R_{los}(\lambda) = dl(\lambda) + \zeta L_{pp}. \quad (26)$$

Among them, the vertical distance  $dl(\lambda)$  from the ship's position point to the target path is:

$$\begin{cases} d_a = \sqrt{(y_s(\lambda) - y_{p+1})^2 + (x_s(\lambda) - x_{p+1})^2}, \\ d_b = \sqrt{(y_{p+2} - y_s(\lambda))^2 + (x_{p+2} - x_s(\lambda))^2}, \\ d_c = \sqrt{(y_{p+2} - y_{p+1})^2 + (x_{p+2} - x_{p+1})^2}. \end{cases} \quad (27a)$$

$$dl(\lambda) = \sqrt{d_a^2 - ((d_c^2 - d_b^2 + d_a^2)/(2d_c))^2}. \quad (27b)$$

When the ship approaches the target point  $P_{p+1}(y_{p+1}, x_{p+1})$ , the target path needs to be switched from  $P_p P_{p+1}$  to  $P_{p+1} P_{p+2}$ . When the ship enters the acceptance circle with  $P_{p+1}$  as the center and  $R_{AC}$  as the radius, that is,  $(y_s(\lambda) - y_{p+1})^2 + (x_s(\lambda) - x_{p+1})^2 \leq R_{AC}^2$ , the target path is switched. In order to ensure that point  $P_{los}(\lambda)$  may be located at inflection point  $P_{p+1}$ , this paper defines  $R_{AC}$  as:  $R_{AC} = R_{los}(\lambda)$ .

#### 4.2. Path-following controller design based on AMBS-P

Let  $\psi_{ref} = \psi_{los}$ , then a heading tracking controller is required. The aim of heading control is to calculate the rudder angle command of

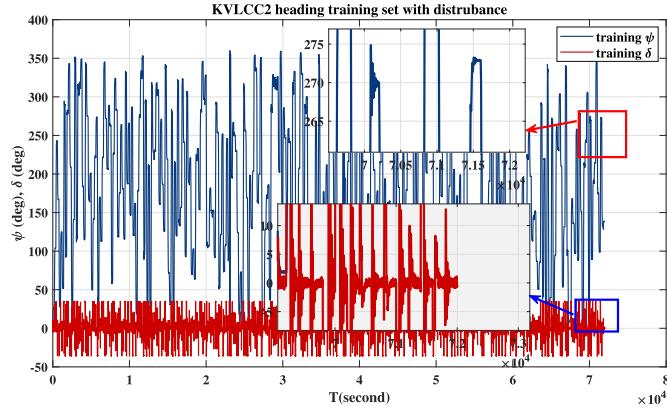


Fig. 5. Training data with errors ("A-train-data-dis",  $D_r = 1$ ).

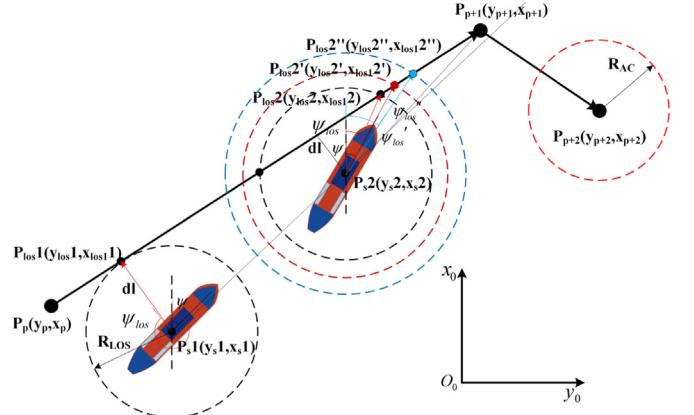


Fig. 6. LOS navigation schematic diagram.

each step, and then input it into the ship motion model to get the next step heading angles, so as to gradually realize the target heading. Based on the data-driven model, the rudder command  $\delta$  of the next time step is obtained by AMBS-P algorithm. In particular, for Eq. (15),  $S_{state}$  is the current heading and  $R_{ref}$  is the target heading  $\Psi_{ref}$ .

According to the type of ship, rudder angle has a range of constraints, set to  $[\delta_{E_{max}}, \delta_{E_{Rmax}}]$ . It needs to be judged after getting the  $\delta_E(\lambda + 1)$ , shown as Eq. (28).

$$\delta_E(\lambda + 1) = \begin{cases} \delta_{E_{max}}, & \delta_E(\lambda + 1) > \delta_{E_{max}}, \\ \delta_{E_{Rmax}}, & \delta_E(\lambda + 1) > \delta_{E_{Rmax}}. \end{cases} \quad (28)$$

In order to ensure the effect of heading control, the idea of predictive control is introduced. In this paper, the target heading of  $N_p$  step is obtained by adjusting the size of dynamic circle, as Eq. (29). Among it,  $n_{NP}$  is LOS circle scale coefficient.

$$R_{LOS}(\lambda + i) = n_{NP} R_{LOS}(\lambda), \begin{cases} i = 1, \dots, N_p \\ n_{NP} = 2, \dots, N_p \end{cases}, \quad (29a)$$

$$\Psi_{los}(\lambda) = [\psi_{los}(\lambda), \psi_{los}(\lambda + 1), \dots, \psi_{los}(\lambda + N_p)]. \quad (29b)$$

Reducing the distance between the ship and the target path is regarded as the primary target, and the speed of heading to the target path will inevitably be slower than before.

The control flow of ship path-following is shown in Fig. 7. The LOS navigation algorithm calculates the reference LOS angle sequence  $\Psi_{LOS}(\lambda)$  of  $\lambda \rightarrow (\lambda + N_p)$  steps according to the current path point. In AMBS-P algorithm,  $B = \delta_E(\lambda)$ . The values of left and right antennae  $\delta_{EL}(\lambda)$  and  $\delta_{ER}(\lambda)$  were calculated.

According to Eq. (15) and data-driven model Eq. (23), the total prediction errors  $f(\delta_{EL}(\lambda))$  and  $f(\delta_{ER}(\lambda))$  are obtained. According to Eq. (12), the next rudder command with the least prediction error is obtained, and the final optimal rudder command  $\delta_E(\lambda + 1)$  is obtained by iteration.  $\delta_E(\lambda + 1)$  and the current state  $S_{STATE}(\lambda)$  are input to the data-driven model to get the next state  $S_{STATE}(\lambda + 1)$ . The pseudo code of the whole control process is given in Algorithm 2.

**Algorithm 2**

Path-following controller design steps.

---

```

1 while  $\lambda < \Lambda$  do
2   The current ship status is  $S_{state}(\lambda)$ ; for
3      $i = 1 \rightarrow N_p$  do
4       Calculate  $R_{los}^i(\lambda)$  according to Eq. (26) and
        Eq. (27);
5       Calculate  $R_{los}^i(\lambda + 1)$  according to Eq. (29);
6   Generate
7      $\Psi_{los}(\lambda) [\psi_{los}(\lambda), \psi_{los}(\lambda + 1), \dots, \psi_{los}(\lambda + N_p)]$ 
      as target headings according to Eq. (25);
8   Initialize:  $B(0), V(0), D_0^0, L_{step}^0, bl_1, bl_2, Pb^0 =$ 
9      $Gb^0 = B(0), F_{best} = F(B(0))$ ;
10  while  $k < K$  do
11    In the Eq. (15) of Section 2.3, combining
        Eq. (23), we get  $F(B_L i^k)$  and  $F(B_R i^k)$ ;
12    Calculate  $\delta_E^k(i + 1)$  according to Eq. (12) to
        Eq. (19);
13  Under the constraint of Eq. (28), the rudder order
         $\delta_E(\lambda + 1)$  is obtained;
14  According to the current state and rudder order,
        the next state  $S_{state}(\lambda + 1)$  of the ship is
        obtained.

```

---

## 5. Simulation results

This section analyzes the model identification performance and path-following control performance.

### 5.1. Simulation setup

A KVLCC2 tanker ship is selected as the target ship and the parameters are given in [Appendix A](#). All simulations were carried out on a laptop computer with a 2.6 GHz i7-10750 CPU and 16 GB RAM. The hydrodynamic parameters can be found in [Yasukawa and Yoshimura \(2015\)](#). We use a modular type ship maneuvering model which was proposed in our earlier work in [Liu et al. \(2016\)](#) as the simulation model. This model was validated by comparing simulated and tested results ([Lee et al., 2007](#); [Yasukawa and Yoshimura, 2015](#)) to reflect the actual characteristics of ship motion. It is noted that this model is only used to generate input, output data and disturbance data.

### 5.2. Analysis of model identification performance

This section first introduces the evaluation index of regression algorithm. Then seven typical regression algorithms are compared. After selecting the appropriate regression algorithm, AMBS-P algorithm is used to obtain the appropriate weight. The influence of the number of training data on the model identification performance is analyzed.

#### 5.2.1. Evaluation index

The commonly used fitting evaluation indexes include root mean square error (RMSE), root mean square (RMS), standard deviation (STD), coefficient of determination ( $R^2$ ), error sum of

squares (SSE), mean absolute percentage error (MAPE), maximum permissible error (MPE), mean absolute error (MAE), mean squared error (MSE), etc. ([Doris and Ernst, 2019](#); [Wang et al., 2020](#)). In order to compare the different methods, the RMSE, STD and  $R^2$  are selected for evaluation of model identification performance, and Taylor diagrams are drawn for display.

RMSE is used to calculate the deviation between predicted value and real value. The smaller the value, the better the prediction effect.

$$RMSE(D_{pre}, D_{real}) = \sqrt{\frac{1}{N_d} \sum_{i=1}^{N_d} (D_{pre} - D_{real})^2}. \quad (30)$$

$R^2$  is the fitting degree of regression, and the maximum value is 1. The closer the value to 1, the higher the degree of explanation of the independent variable to the dependent variable, the higher the percentage of the change caused by the independent variable to the total change, the better the fitting degree. Among it,  $D_{pre}$  is the mean value of the forecast.

$$R^2(D_{pre}, D_{real}) = 1 - \frac{\sum_{i=1}^N (D_{pre} - D_{real})^2}{\sum_{i=1}^N (D_{pre} - \bar{D}_{pre})^2}. \quad (31)$$

STD is the average of the distance of each data from the average. It can reflect the degree of dispersion of a data set.

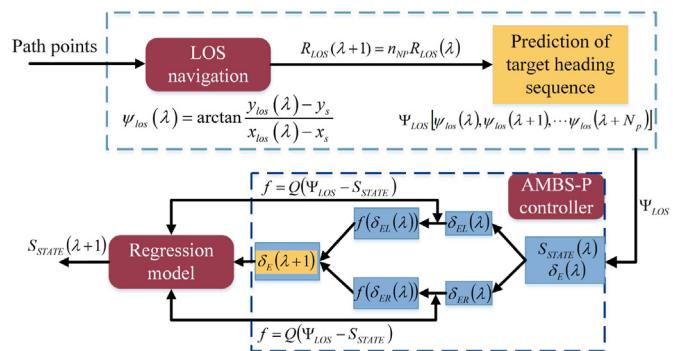
$$STD(D_{pre}, D_{real}) = \sqrt{\frac{1}{N_d} \sum_{i=1}^{N_d} (D_{pre} - \bar{D}_{pre})^2}. \quad (32)$$

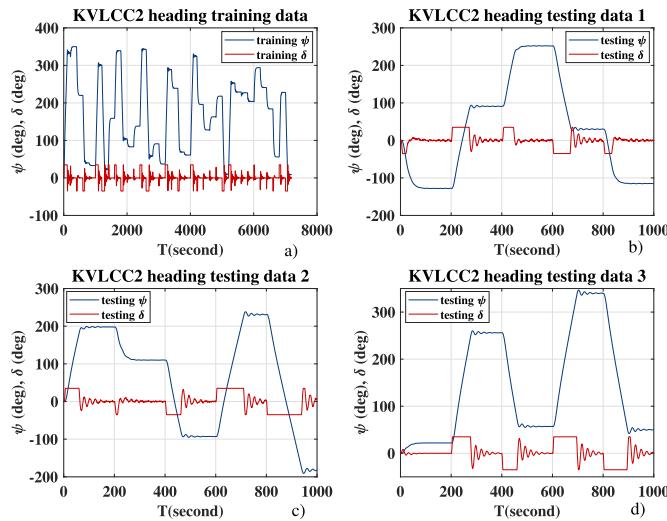
#### 5.2.2. Comparison of regression algorithms

This paper selects seven similar machine learning algorithms for comparison, including decision tree, bagging, random forest, Gaussian process regression, ridge regression, lasso regression and elastic net regression.

Without detailed appropriate weight test, the initial setting  $R_w = 0.2$ . 7200 steps of "A-train-data" are randomly selected for training and recorded as training data "B-train-data". 1000 steps of testing data are used for prediction, and the positive and negative heading sequences are selected as testing data "B-test-data-a", "B-test-data-b", "B-test-data-c". The comparison of regression algorithms and prediction results are shown in [Fig. 8](#), [Fig. 9](#) and [Table 1](#).

It can be seen from [Fig. 9](#) and [Table 1](#) that the prediction effect is better when the heading of three groups of testing data is positive. When the testing data is negative, the first four algorithms (decision tree, bagging, random forest, Gaussian process) have poor





**Fig. 8.** Training data and testing data, (a) Training data ("B-train-data"), b) Testing data ("B-test-data-a"), c) Testing data ("B-test-data-b"), d) Testing data ("B-test-data-c").

effect. It shows that the first four regression algorithms are highly dependent on the training data and have narrow adaptability. They are only suitable for the situation similar to the training data, namely  $[0^\circ, 360^\circ]$ . The latter three algorithms (lasso regression, ridge regression and elastic net regression) can well adapt to the large-scale changes of  $[-360^\circ, 360^\circ]$ , and are more suitable for model prediction.

Based on the above results, the Taylor diagrams are drawn as shown in Fig. 10. Through the Taylor diagram, it can be seen the advantages and disadvantages of each regression algorithm. Combined with the calculation speed in the actual process, we can finally choose the elastic net regression model as the regression algorithm, which has the characteristics of high accuracy and fast calculation speed.

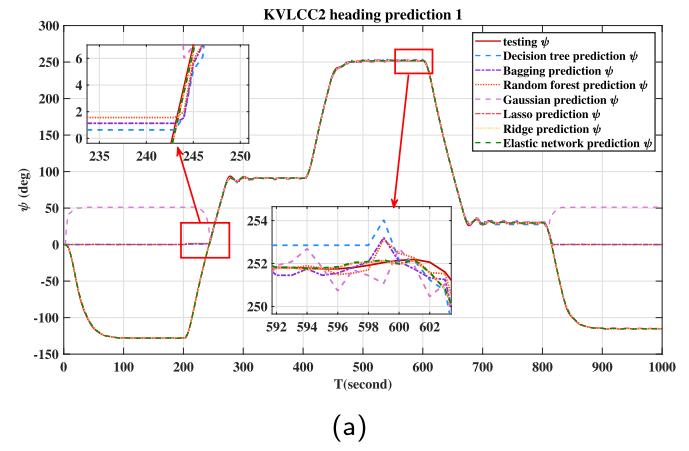
### 5.2.3. wt and parameters selection

In order to obtain the suitable weight value for the ship, AMBS-P algorithm is used. Since it does not involve overshoot, prediction and other factors, only AMBS part other than Eq. (15) is used. The inputs are weight parameters  $\alpha$  and  $R_w$ . Output is RMSE of model identification. The constraints are  $\begin{cases} 0 < \alpha \leq 1 \\ 0 < R_w \leq 1 \end{cases}$ . Take the testing data ("B-test-data-b") as an example, the iterative search results are shown in Fig. 11.

As can be seen from Fig. 11,  $\alpha = 0.8934$ ,  $R_w = 0.3454$ . The RMSE is the minimum, and  $R^2 = 0.999968$ , RMSE = 0.734886.

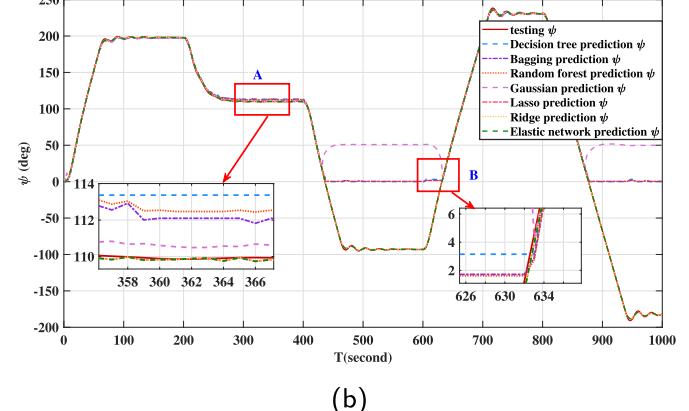
### 5.2.4. Quantity of selected training data

In order to further evaluate the accuracy of the data-driven model, the logic of "k-fold cross validation" method is used for analysis. This method has two functions: (1) Increase the diversity of samples. (2) The minimum number of training data needed for accurate prediction data can be obtained to increase the calculation speed. The training data sample "A-train-data" is divided into  $K_w$  parts on average, and then cyclic training is carried out. Finally, it is used to predict testing data "B-test-data-b", and the mean RMSE is chosen as the error. The evaluation results are shown in Table 2. For testing data "B-test-data-b", according to practical experience, the prediction error should be within  $\pm 1^\circ$ . The prediction effect with 95% confidence interval trained by training data "A-train-data" is shown in Fig. 12.



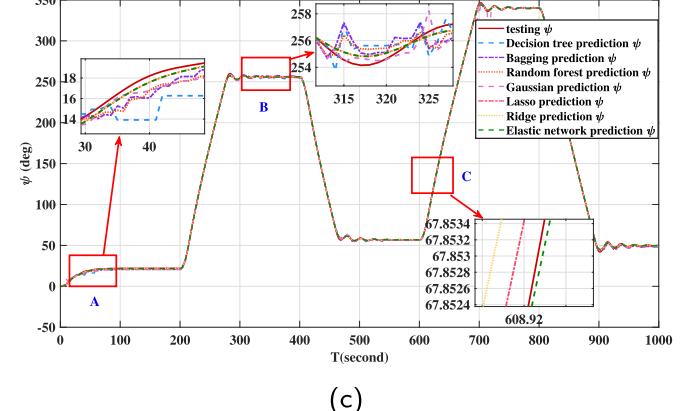
(a)

KVLCC2 heading prediction 2



(b)

KVLCC2 heading prediction 3



(c)

**Fig. 9.** Ship heading prediction for different regression algorithms, (a) Testing data ("B-test-data-a"), (b) Testing data ("B-test-data-b"), (c) Testing data ("B-test-data-c").

To sum up, it can be find that within  $K = 150$ , that is, when the training data number of each group is greater than 480, the prediction error is within  $1^\circ$ . Therefore, the minimum number of training data is more than 480.

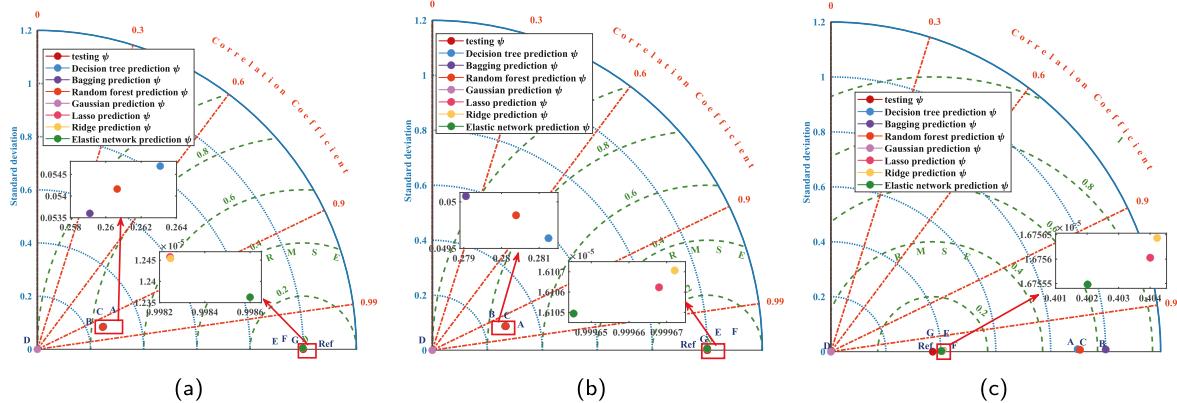
### 5.3. Analysis of path-following controller performance

Typical target trajectories to evaluate path-following control performance can be divided into straight trajectory and curved trajectory. The curved trajectory is divided into smooth curves and

**Table 1**

Analysis of fitting algorithm results.

Group	Evaluation index	Decision Tree	Bagging	Random forest	Gaussian process	Ridge	Lasso	Elastic net
"B-test-data-a"	$R^2$ (-)	0.7194	0.7189	0.7188	0.8523	0.9999	0.9999	0.9999
	RMSE (deg)	71.2230	71.2820	71.3010	103.2200	0.6790	0.6789	0.6742
	Computing speed (second)	0.0189	0.1097	1.0745	218.7980	0.0279	0.0020	0.0019
"B-test-data-b"	$R^2$ (-)	0.7553	0.7537	0.7540	0.8832	0.9999	0.9999	0.9999
	RMSE (deg)	64.0390	64.2450	64.2070	90.3872	0.7355	0.7355	0.7354
	Computing speed (second)	0.0150	0.1022	0.9963	190.3000	0.0803	0.0040	0.0020
"B-test-data-c"	$R^2$ (-)	0.9999	0.9999	0.9999	0.9995	0.9999	0.9999	0.9999
	RMSE (deg)	2.0640	1.8240	1.4440	2.6551	0.6158	0.6158	0.6155
	Computing speed (second)	0.0160	0.1037	1.0013	182.1362	0.0046	0.0020	0.0010

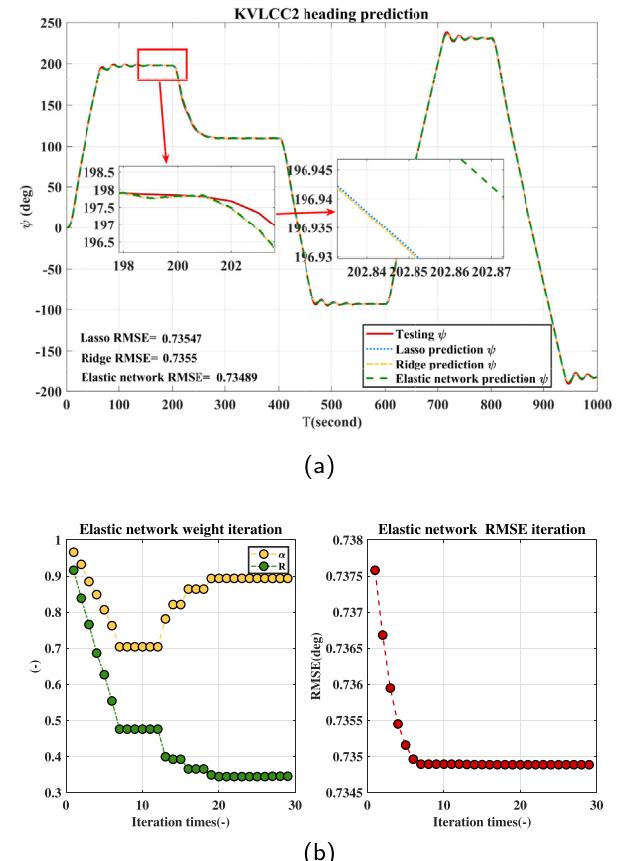
**Fig. 10.** Taylor diagram, (a) Testing data ("B-test-data-a"), (b) Testing data ("B-test-data-b"), (c) Testing data ("B-test-data-c").

non smooth curves. The smooth curves includes elliptic curve (Shen and Jing, 2019; Shen, Bi, Guo and Wang, 2019a), circular curves (Liu et al., 2019), sine and its similar trigonometric function curves (Ma et al., 2019; Liu, 2019b), spiral curves (Zhu and Du, 2020; Liu, 2019a), comb curve (Shen et al., 2019b), and Bessel curves (Sawada et al., 2021). The non smooth curves include random path points (Liu et al., 2018; Deng et al., 2019), regular hexagons (Yu, 2018), and closed curves (Zhao et al., 2020).

In this paper, to cover all aspects of straight trajectory and curve trajectory (smooth and non smooth), linear trajectory, Bessel curves and random path points are selected. The control parameters are shown in Appendix A. The control effect is shown in Fig. 13 and Table 3. As can be seen from Fig. 13, the effect is better whether it is tracking straight line or curve. Since the initial position of the ship is far from the starting point of the smooth curve, the RMSE calculated from the initial position of the ship is 24.1836m, which has a large error and can not reflect the real tracking effect. Starting from the first position point of the tracking reference track ( $y = 96.2359, x = 174.1618$ ) (as shown at the green star in Fig. 13), the RMSE is 0.3880m. Combined with Fig. 13, it can be concluded that the RMSE can reflect the ship path following effect at this time.

### 5.3.1. Analysis of $N_p$

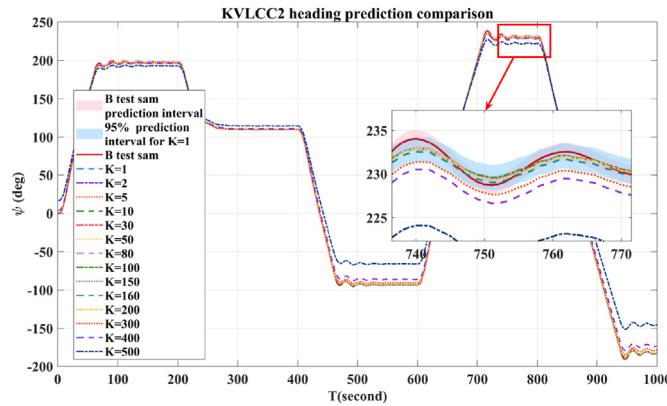
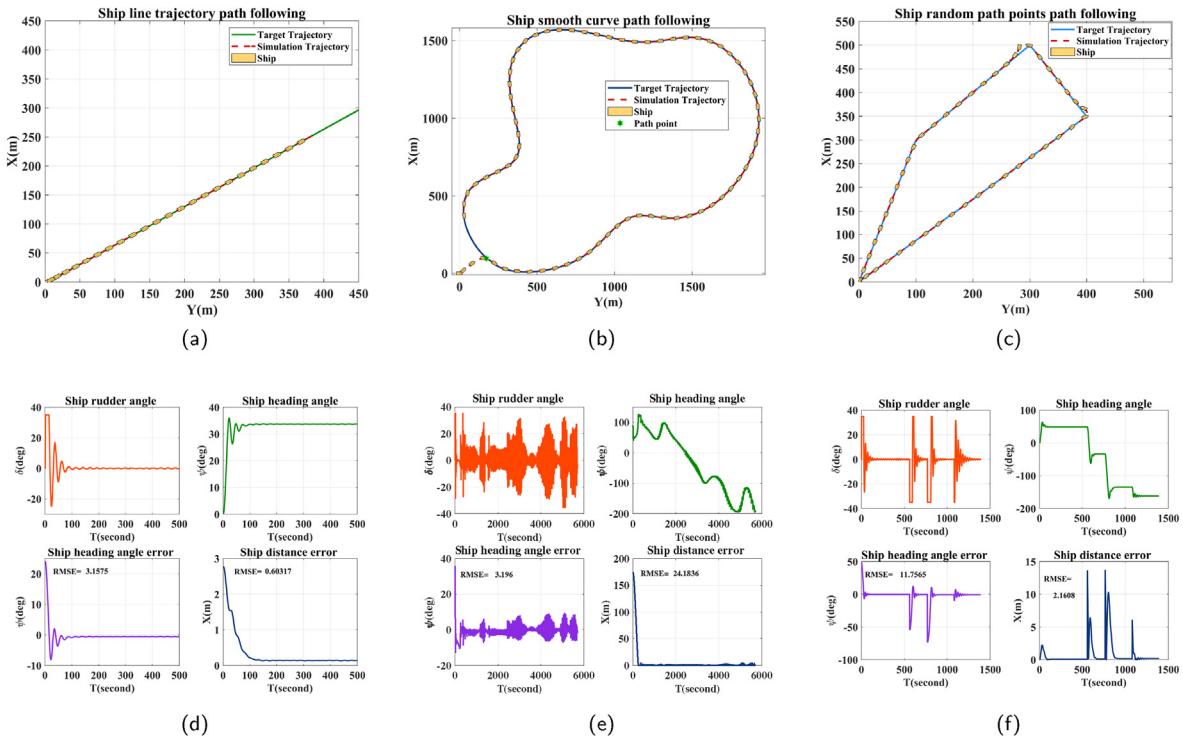
According to the principle of the algorithm, the predicted step number  $N_p$  has an important influence on the effect of ship motion control. Fig. 14 shows that when  $N_p = 1$ , the path following control deviates, and  $NaN$  is used here to indicate failure. When  $N_p = 5, 9$  or 10, the tracking effect is better. As the number of prediction steps increases, the single-step calculation time also increases. However, too slow calculation speed is not suitable for actual navigation control. Based on the above factors, the optimal prediction steps  $N_p = 5$  are obtained.

**Fig. 11.** Parameter and weight search results analysis, (a) Iterative prediction of final results, (b) Weight and coefficient.

**Table 2**

Evaluation results of k-fold cross validation method.

$K_w$	Number of each group	Mean RMSE (deg)
1	72000	0.7349
10	7200	0.7367
50	1440	0.7523
100	720	0.8160
150	480	1.0029
160	450	1.4631
400	180	6.7966
500	144	21.583

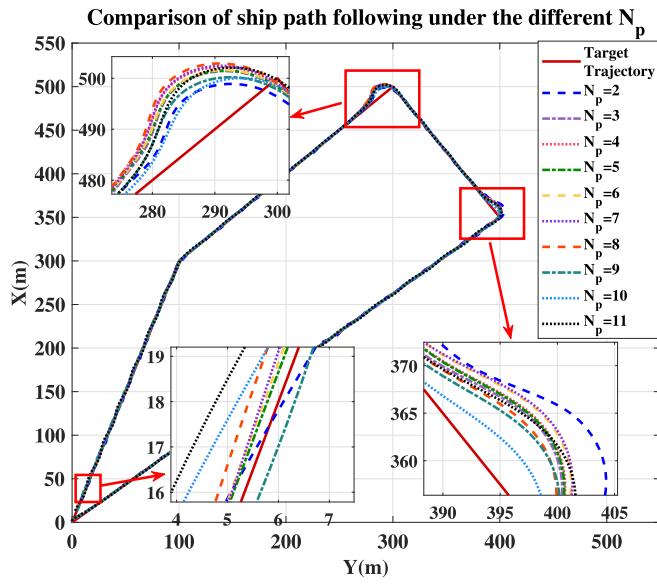
**Fig. 12.** Ship heading prediction under different  $K$ .**Fig. 13.** Ship path following situation, a) Linear target trajectory, b) Smooth curve trajectory, c) Trajectories composed of random path points, d) The change of rudder angles, heading angles and errors in ship line path following, e) The change of rudder angles, heading angles and errors in ship smooth curve path following, f) The change of rudder angles, heading angles and errors in ship random path points path following.**Table 3**

Path-following control error.

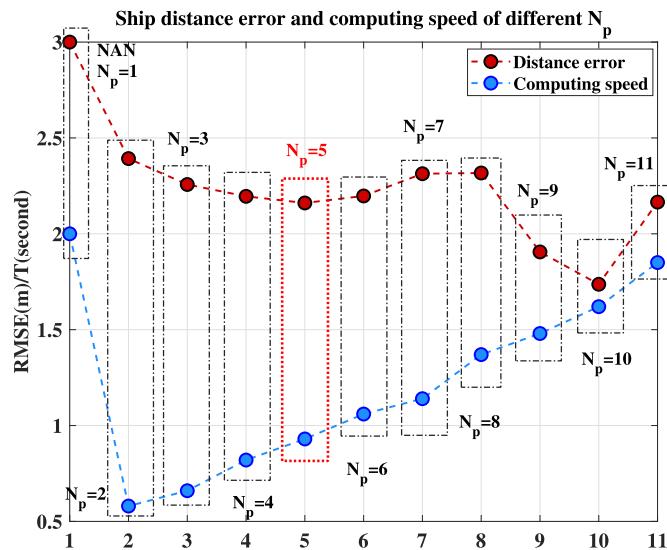
Trajectory type	RMSE	
	heading error (deg)	distance error(m)
Line	3.1575	0.6032
Smooth curve	3.1960	0.3880
Path points	11.7565	2.1608

### 5.3.2. Controller effect comparison

Fig. 15 and Table 4 give the comparison results of PID algorithm and AMBPS-P algorithm. Based on the principle of low rudder angle variation frequency and stable control effect, we set the PID parameters as  $K_B, K_I, K_D = [5, 0, 0]$ . Based on paper (Wang et al., 2019a), AMBPS-P algorithm is adopted, and  $K_B, K_I, K_D = [10, 0, 40]$ . Fig. 15 shows that AMBPS-P based on identification model has  $0.25^\circ$  less in heading error and  $0.07m$  less in distance than P control. Converted into percentage form, the heading tracking error is reduced by 2.10% and the path following error is reduced by 3.02%. Besides, AMBPS-P based on identification model controller is  $0.24^\circ$  less in heading error and  $0.06m$  less in distance than PID controller. In percentage form, the heading tracking error is reduced by 2.02%, and the path following error is reduced by 2.98%. It indicates that the whole control scheme is feasible and can achieve good control effect. For the research object with no model or inaccurate model, data-driven model and predictive algorithm can be established to control.



(a)



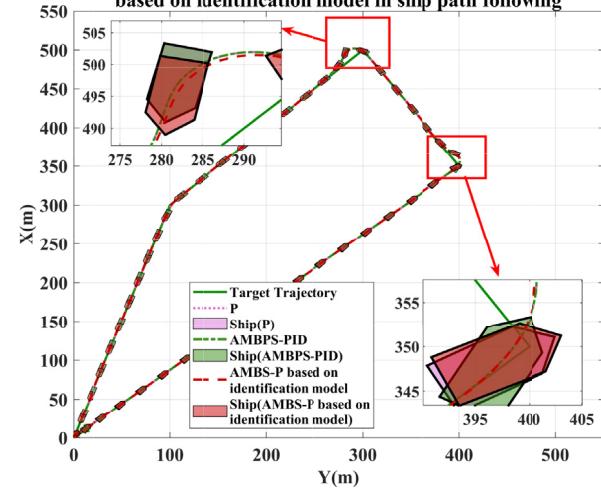
(b)

Fig. 14. Analysis of ship path following under the different  $N_p$ , a) Ship path following under the different  $N_p$ , b) Ship distance error and computing speed of different  $N_p$ .

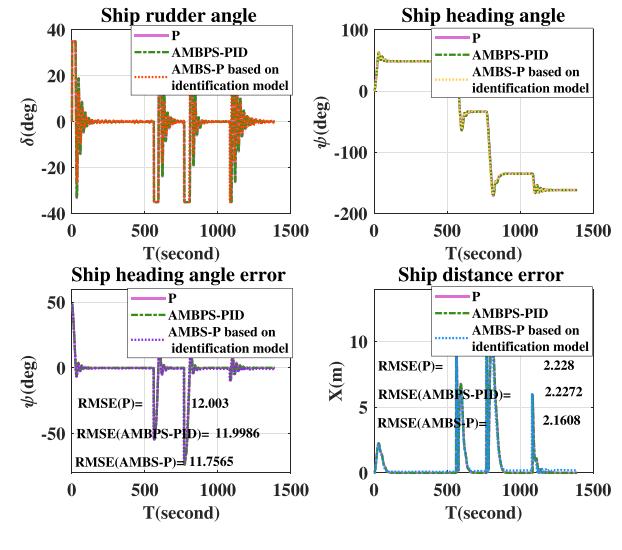
### 5.3.3. Comparison of control results based on identification model and mechanism model

In order to verify the application effect, the identification based model is compared with the mechanism model in paper (Liu et al., 2016). The comparison results are shown in Fig. 16. As can be seen from Table 5, compared with the control method combining AMBS-P with mechanism model, the heading and the distance errors of the control method in this paper are  $1.12^\circ$  and  $0.72m$  less

The comparison effect of P, AMBPS-PID and AMBS-P based on identification model in ship path following



(a)



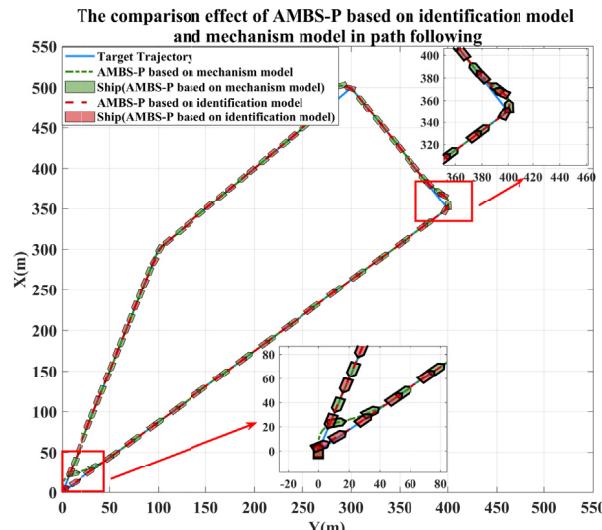
(b)

Fig. 15. The comparison effect of P, AMBPS-PID and AMBS-P based on identification model in ship path following, a) The comparison effect of three methods in ship path following, b) The change of rudder angles, heading angles and errors in ship path following.

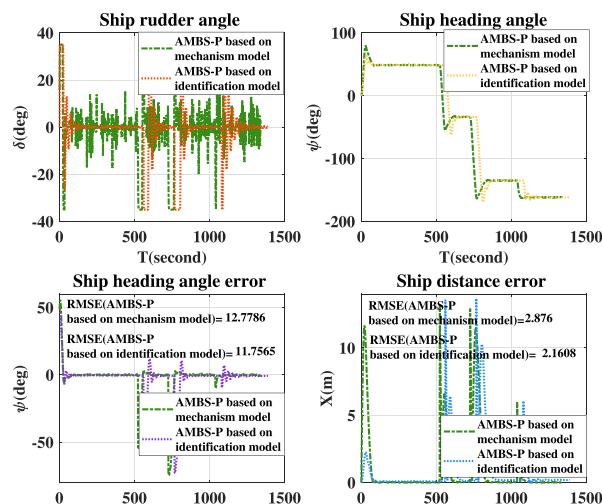
Table 4

The comparison effect of P, AMBPS-PID and AMBS-P based on identification model in ship path following.

Algorithm	RMSE	
	heading error (deg)	distance error(m)
P	12.0030	2.2280
AMBPS-PID	11.9986	2.2272
AMBS-P	11.7565	2.1608



(a)



(b)

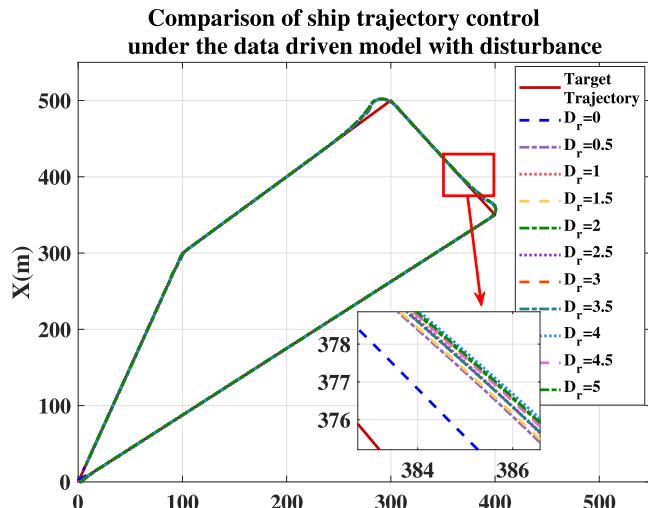
**Fig. 16.** The comparison effect of AMBS-P based on identification model and mechanism model in path following control, a) The comparison effect of AMBS-P based on identification model and mechanism model in path following control, b) The change of rudder angles, heading angles and errors in ship path following.

**Table 5**

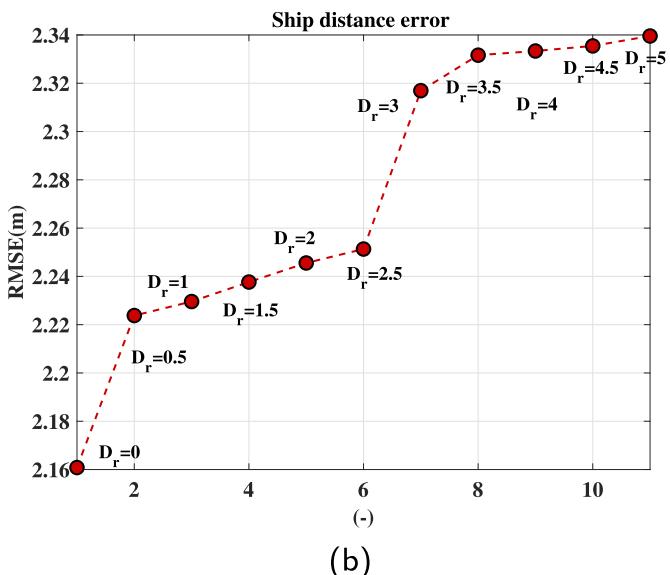
RMSE of identification model and mechanism model in AMBS-P control.

Model	RMSE	
	heading error (deg)	distance error(m)
Identification model	11.7565	2.1608
Mechanism model	12.8754	2.8808

respectively. When converted into percentage form, the heading tracking error is reduced by 8.69% and the path following error is



(a)

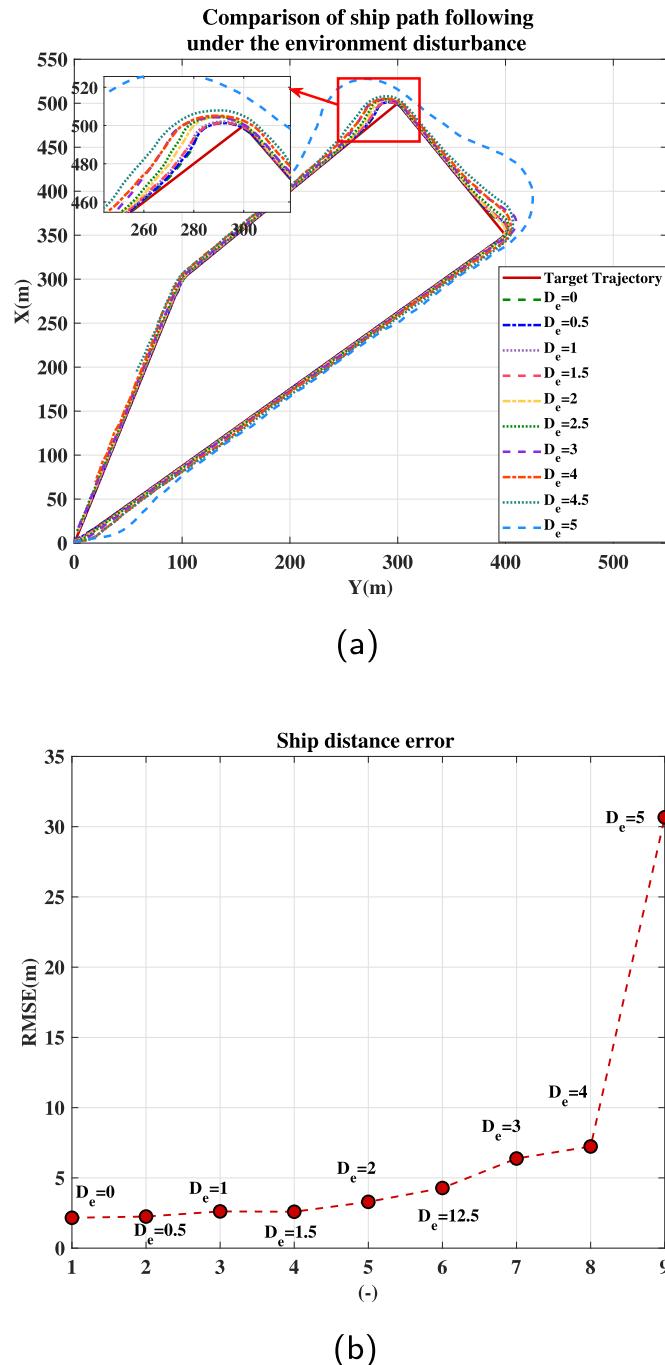


**Fig. 17.** Ship path following under disturbance of training data, a) Ship path following under disturbance of training data, b) Distance error of ship path following.

reduced by 33.32%. In particular, it can be seen from Fig. 16 (b) that the change frequency of rudder angle is low, which can effectively protect the rudder in practical application.

#### 5.3.4. Disturbance of training data

If the acquisition of training data is disturbed by the external environment, the acquired training data will have a certain degree of error, which needs to be corrected by the designed controller. Set  $\tau_r$  according to Eq. (24),  $D_r = [0, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5]$ .  $\tau_r$  is the disturbance error sequence generated for the training data. As can be seen from Fig. 17, the path-following error increases with the model error. The change rate changes smoothly in the beginning and then suddenly increased, but the overall effect is good, which



**Fig. 18.** Comparison of ship path following under environmental disturbance, a) Ship path following under environmental disturbance, b) Distance error of ship path following.

indicates that the controller has the ability to correct the inaccurate model.

### 5.3.5. Environmental disturbance

It is also evaluated whether AMBS-P based on identification model controller has the ability to resist external disturbances caused by the environment. The first-order Markov process is used to describe the external low-frequency disturbance suffered by the ship:

$$\begin{cases} \tau_e = \mathbf{T}^T(\psi)d_e, \\ \dot{d}_e = -T_e^{-1}d_e + \gamma_e G_s, \end{cases} \quad (33)$$

where,  $\tau_e$  is the disturbance error sequence generated for the environment.  $\dot{d}_e = -T_e^{-1}d_e + \gamma_e G_s$  is the first-order Markov process.  $T_e$  is a diagonal constant matrix.  $\gamma_e$  is a diagonal matrix scaling the amplitude of  $G_s$ .  $G_s$  is a vector of zero-mean Gaussian white noises. Since the model used in this paper is a one-dimensional “rudder-heading” model, Eq. (33) is simplified to the disturbance of heading control, taken as:  $\tau_e = \int_{\lambda_t}^{\lambda_{t+1}} (-T_e^{-1}d_e + \gamma_e G_s)$ . Here,  $\lambda_t$  is the sampling time.  $d_e(0) = 0.1$ ,  $T_e = 2D_e$ ,  $\gamma_e = D_e$ ,  $D_e$  is disturbance amplitude coefficient, and  $D_e = [0, 0.5, 1, 1.5, 2, 2.5, 3, 4, 4.5, 5]$ . It can be seen from Fig. 18 that the path-following error increases with the increase of the environmental interference coefficient. In the initial stage, the error change rate is low, which shows that the algorithm has certain robustness.

## 6. Conclusions and future work

This paper combines data-driven model with intelligent algorithm to deal with the path-following control problem of under-actuated ships without knowing its model parameters. Through the analysis of various regression algorithms, the elastic net regression algorithm is used to fit the model. In order to select the appropriate elastic net regression parameters, AMBS-P algorithm is used to find the optimal weights. The LOS angle of the prediction step is obtained by setting the dynamic LOS circle. Combined with the data-driven model, the ship state can be predicted, upon which a AMBS-P-based controller is designed. Simulation results show that compared with PID control, the heading tracking error is reduced by 2.02% and the path following error is reduced by 2.98%. The proposed method has accurate control performance and could deal with environmental disturbances. This implies that in situations in which an accurate ship model is difficult to obtain, the proposed method could provide a model-free way to design effective controllers for the target ship.

For future work, this research will be extended in several directions. Firstly, experiments in actual environments will be carried out to further validate its effectiveness. Secondly, other control algorithms can also be integrated with the identified model compare their performances. Moreover, it is noted that this paper uses PID to generate initial data due to its simplicity and practicality, the PID controller could also be replaced with other advance control techniques to generate other types of initial data. It would be interesting to investigate how the quality of the off-line data would affect the model identification process.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

Supported by Program of Marine Economy Development Special Fund (Six Marine Industries) under Department of Natural Resources of Guangdong Province (Project No. GDNRC[2021]59), National Natural Science Foundation of China (62003250), Southern Marine Science and Engineering Guangdong Laboratory (Zhuhai) (SML2021SP101).

## A. Symbols

**Table 6**  
Parameter setting of control algorithm.

Symbol	Meaning	Value	Unit
$bl_1$	Step scale coefficient	0.99	—
$bl_2$	Antenna scale coefficient	0.95	—
$c_1, c_2$	Location coefficients	2, 2	—
$r_1, r_2$	0 – 1 random variables	0–1	—
$D_0$	Initial antennae length	0.99	—
$q$	Weight factor	1	—
$K$	Maximum number of iterations	10	—
$L_{pp}$	Ship length between perpendiculares	7	m
$L_{step}$	Beetle step length	1	—
$N$	Spatial dimension	1	—
$N_p$	The number of prediction steps	5	—
$\Theta_0$	Aggregation degree of beetle swarm threshold	1	—
$\alpha$	Identification model coefficient	0.8934	—
$R_w$	Identification model weight	0.3454	—
$\omega_n$	Second-order filter coefficient	0.07	—
$\xi$	Second-order filter coefficient	0.95	—
$[\delta_{E_{lmax}}, \delta_{E_{rmax}}]$	Rudder angle limit	[−35, 35]	deg

## References

- Carron, A., Arcari, E., Wermelinger, M., Hewing, L., Zeilinger, M.N., 2019. Data-driven model predictive control for trajectory tracking with a robotic arm. *IEEE Robot. Automat. Lett.* 4, 3758–3765.
- Deng, Y., Zhang, X., Zhang, G., 2019. Fuzzy logic based speed optimization and path following control for sail-assisted ships. *Ocean Eng.* 171, 300–310.
- Doris, Schadler, Ernst, Stadlober, 2019. Fault detection using online selected data and updated regression models. *Measurement* 437–449.
- Fossen, T.I., Pettersen, K.Y., Galeazzi, R., 2015. Line-of-sight path following for dubins paths with adaptive sideslip compensation of drift forces. *IEEE Trans. Control Syst. Technol.* 23, 820–827.
- Haseltalab, A., Negenborn, R.R., 2019. Adaptive control for autonomous ships with uncertain model and unknown propeller dynamics. *Control Eng. Pract.* 91, 104116.1–104116.12.
- Hoerl, A., Hoerl, C., 1962. Application of ridge analysis to regression problems. *Chem. Eng. Prog.* 58, 54–59.
- Hui, Z., Hastie, T., 2004. Regression shrinkage and selection via the elastic net, with applications to microarrays. *J. Roy. Stat. Soc. Ser. B Stat. Methodol.* 67, 1–10.
- Jiang, X., Li, S., 2017. Bas: beetle antennae search algorithm for optimization problems. *Int. J. Robot. Contr.* 1, 1–3.
- Karagiannidis, P., Themelis, N., 2021. Data-driven modelling of ship propulsion and the effect of data pre-processing on the prediction of ship fuel consumption and speed loss. *Ocean Eng.* 222, 108616.
- Koyama, T., Watanabe, T., Watanabe, I., 1973. An application of the least square method to the ship maneuverability identification. *J. Soc. Nav. Archit. Jpn.* 134, 173–181.
- Lee, S., Toxopeus, S., Quadflieg, F., 2007. Free Sailing Manoeuvring Tests on KVLC1 and KVLC2. Technical Report. Maritime Research Institute Netherlands (MARIN), Wageningen, The Netherlands.
- Li, S., Liu, J., Negenborn, R.R., Wu, Q., 2020. Automatic Docking for Underactuated Ships Based on Multi-Objective Nonlinear Model Predictive Control. *IEEE Access*, pp. 70044–70057.
- Liu, J., Quadflieg, F., Hekkenberg, R., 2016. Impacts of the rudder profile on manoeuvring performance of ships. *Ocean Eng.* 124, 226–240.
- Liu, Y., Bu, R., Gao, X., 2018. Ship trajectory tracking control system design based on sliding mode control algorithm. *Pol. Marit. Res.* 25, 26–34.
- Liu, Y., Bu, R., Li, Q., 2019. Design for underactuated ships' trajectory tracking control. *Comput. Integr. Manuf. Syst.* 36, 6–10.
- Liu, Z., 2019a. Practical backstepping control for underactuated ship path following associated with disturbances. *IET Intell. Transp. Syst.* 13, 834–840.
- Liu, Z., 2019b. Pre-filtered backstepping control for underactuated ship path following. *Pol. Marit. Res.* 26, 68–75.
- Ma, Y., Zhu, G., Li, Z., 2019. Error-driven-based nonlinear feedback recursive design for adaptive nn trajectory tracking control of surface ships with input saturation. *IEEE Trans. Syst. Mag.* 17–28.
- Ostafew, C.J., Schoellig, A.P., Barfoot, T.D., Collier, J., 2016. Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking. *J. Field Robot.* 33, 133–152.
- Sawada, R., Hirata, K., Kitagawa, Y., Saito, E., Fukuto, J., 2021. Path following algorithm application to automatic berthing control. *J. Mar. Sci. Technol.* 26, 541–554.
- Shen, Z., Bi, Y., Guo, T., Wang, R., 2019a. Adaptive dynamic surface output feedback trajectory tracking control for underactuated ships with nonlinear observer. *Syst. Eng. Electron.* 41, 409–416.
- Shen, Z., Jing, F., 2019. Neuron adaptive iterative sliding-mode control for path tracking of underactuated ship. *J. Harbin Eng. Univ.* 40, 60–65+71.
- Shen, Z., Zou, T., W, R., 2019b. Extended state observer based adaptive dynamic surface output feedback control for underactuated surface vessel trajectory tracking with low-frequency learning. *Control Theory & Appl.* 36, 867–876.
- Shin, J., Dong, J.K., Lee, Y., 2017. Adaptive path-following control for an unmanned surface vessel using an identified dynamic model. *IEEE ASME Trans. Mechatron.* 22, 1143–1153.
- Ugo, R., Zhang, X., Borrelli, F., 2018. Data-driven predictive control for autonomous systems. *Annu. Rev. Contr.* 49, 259–286.
- Wang, L., Wu, Q., Liu, J., Li, S., Negenborn, R.R., 2019a. Ship motion control based on ambps-pid algorithm. *IEEE Access* 7, 183656–183671.
- Wang, X., Z, Z., 2012. Identification of ship manoeuvring response model based on fruit fly optimization algorithm. *J. Dalian Marit. Univ.* 38, 1–4.
- Wang, X.G., Zou, Z., Hou, X.R., Xu, F., 2015. System identification modelling of ship manoeuvring motion based on support vector regression. *J. Hydrodyn.* 27, 502–512.
- Wang, Z., Xu, H., Xia, L., Zou, Z., Soares, C.G., 2020. Kernel-based support vector regression for nonparametric modeling of ship maneuvering motion. *Ocean Eng.* 216, 1–12.
- Wang, Z., Zou, Z., Soares, C.G., 2019b. Identification of ship manoeuvring motion based on mi-support vector machine. *Ocean Eng.* 183, 270–281.
- Xue, Y., Liu, Y., Ji, C., Xue, G., Huang, S., 2020. System identification of ship dynamic model based on Gaussian process regression with input noise. *Ocean Eng.* 216, 107862.
- Yasukawa, H., Yoshimura, Y., 2015. Introduction of mmg standard method for ship maneuvering predictions. *J. Mar. Sci. Technol.* 20, 37–52.
- Yu, Y., 2018. Ship trajectory tracking using improved simulated annealing and reinforcement learning. *IEEE Trans. Syst. Man Cybern. Part B* 1, 1384–1389.
- Zhang, G., Zhang, X., Pang, H., 2015. Multi-innovation auto-constructed least squares identification for 4 dof ship manoeuvring modelling with full-scale trial data. *ISA (Instrum. Soc. Am.) Trans.* 58, 186–195.
- Zhang, X.G., Zou, Z.J., 2011. Identification of abkowitz model for ship manoeuvring motion using e-support vector regression. *J. Hydrodyn.* 23, 353–360.
- Zhao, Y., Qi, X., Incecik, A., Ma, Y., Li, Z., 2020. Broken lines path following algorithm for a water-jet propulsion usv with disturbance uncertainties. *Ocean Eng.* 201, 1–9.
- Zhu, G., Du, J., 2020. Global robust adaptive trajectory tracking control for surface ships under input saturation. *IEEE J. Ocean. Eng.* 45, 442–450.
- Zhu, M., Axel, H., Wen, Y.Q., 2018. Identification-based controller design using cloud model for course-keeping of ships in waves. *Eng. Appl. Artif. Intell.* 75, 22–35.
- Zhu, M., Hahn, A., Wen, Y.Q., Bolles, A., 2017. Identification-based simplified model of large container ships using support vector machines and artificial bee colony algorithm. *Appl. Ocean Res.* 68, 249–261.