Received: 25 December 2017 | Revised: 31 August 2018 | Accepted: 11 June 2019

DOI: 10.1002/asjc.2208

REGULAR PAPER

WILEY

Adaptive predictive path following control based on least squares support vector machines for underactuated autonomous vessels

Chenguang Liu[1,2] | Huarong Zheng[3] | Rudy Negenborn[4] | Xiumin Chu[1] | Shuo Xie[1]

[1]National Engineering Research Center for Water Transport Safety, Wuhan University of Technology, Wuhan, China

[2]State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China

[3]College of Control Science and Engineering, Zhejiang University, Hangzhou, China

[4]Department of Maritime and Transport Technology, Delft University of Technology, Delft, The Netherlands

Correspondence
Xiumin Chu, National Engineering Research Center for Water Transport Safety, Wuhan University of Technology, Wuhan, China.
Email: chuxm@whut.edu.cn

Funding information
China Postdoctoral Science Foundation, Grant/Award Number: 2018M632923; China Scholarship Council, Grant/Award Number: 201506950053; LIESMARS Special Research Funding

Abstract

Since vessel dynamics could vary during maneuvering because of load changes, speed changing, environmental disturbances, aging of mechanism, etc., the performance of model-based path following control may be degraded if the controller uses the same motion model all the time. This article proposes an adaptive path following control method based on least squares support vector machines (LS-SVM) to deal with parameter changes of the motion model. The path following controller consists of two components: the online identification of varying parameters and model predictive control (MPC) using the adaptively identified models. For the online parameter identification, an improved online LS-SVM identification method is proposed based on weighted LS-SVM. Specifically, the objective function of LS-SVM is modified to decrease the errors of parameter estimation, an index is proposed to detect the possible model changes, which speeds up the rate of parameter convergence, and the sliding data window strategy is used to realize the online identification. MPC is combined with the line-of-sight guidance to track straight line reference paths. Finally, case studies are conducted to verify the effectiveness of the proposed path following adaptive controller. Typical parameter varying scenarios, such as rudder aging, current variations and changes of the maneuverability are considered. Simulation results show that the proposed method can handle the above situations effectively.

KEYWORDS

path following, least squares support vector machines (LS-SVM), parameter identification, model predictive control (MPC), autonomous surface vessels (ASV)

1 | INTRODUCTION

Autonomous vessels have received much attention because of their low labor cost, safety, and high efficiency. Path following control of autonomous vessels has been studied significantly in recent years [1–7]. One of challenges in path following control is the fact that vessels are usually underactuated without a sway thruster [4]. It means that the maneuverability of an underactuated vessel degrades compared with a full-actuated vessel during path following control because the sway motion for the underactuated vessel can not be controlled directly.

Another challenge is the guarantee of control robustness under disturbances and changes of vessel dynamics [8]. Robust control and adaptive control are usually used to solve the second challenge [3,9,10]. Robust control is to design a constant gain feedback controller provided that uncertain parameters or disturbances are within some set, and aims to achieve robust performance and stability in the presence of bounded modeling errors [11]. Adaptive control is to design a controller that must adapt to a controlled system with variable parameters or uncertainties [12]. Adaptive control is different from robust control in that it does not need a priori information about the bounds on these uncertain or time-varying parameters [11]. Considering that it is difficult to predict the range and strength of disturbances or uncertainties during path following in advance, an adaptive control method is more suitable to deal with these uncertainties.

The principle of adaptive control is shown in Figure 1. The model-based controller calculates the optimal input according to reference trajectories and an updated motion model. The updated motion model is obtained with a parameter identification method that utilizes system input and output data. Vessels dynamics are varying when there exist disturbances or uncertainties, for instances, changes of vessel properties, environmental disturbances, equipment aging, etc. To avoid the degradation of control performance, it is needed to identify the motion model online and adjust the control law accordingly. The method of support vector machines (SVM) is introduced for classification and function estimation based on structural risk minimization principle in [13,14]. SVM solutions are characterized by convex optimization problems to avoid local minimums with classical neural networks approaches [15,16]. Least squares support vector machines (LS-SVM) based classifiers were proposed by Suykens [17], which works with equality constraints instead of inequality constraints and a sum squared error cost function (SSE). LS-SVM simplifies the problem with the solution characterized by a set of linear equations rather than a convex quadratic program [18,19]. LS-SVM can deal with two classes of problems: classification and function estimation, regression or system identification [17]. As aforementioned, the second class of problem is focused in this article, i.e., system identification for the vessel motion model using LS-SVM. Suykens et al. [18] proposed a weighted LS-SVM method for nonlinear function estimation and solved the robustness and sparse approximation problems with LS-SVM. In [19], an online trained LS-SVM is proposed by means of incremental updating and decremental pruning algorithms. Similarly, an online LS-SVM was derived with adding and deleting a data pair by Li et al. [20].

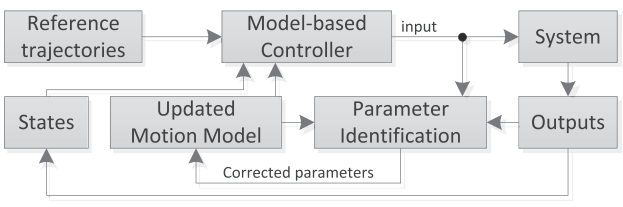


FIGURE 1 Adaptive control principle

Once the motion model is identified, a vessel can follow reference paths with a series of control actions generated by the path following controller with this model. There have been many control methods used for path following [5,21,22]. One of difficulties for path following controller design is to satisfy rudder magnitude constraints [5]. Model predictive control (MPC) offers a good choice to handle this challenge because of its advantage of considering constraints explicitly [23,24]. Since MPC relies on a system model for trajectory predictions, prediction models should be updated when system dynamics change. Therefore, the aforementioned LS-SVM and MPC can be combined. Shi et al. [25] propose a nonlinear model predictive controller based on a nonlinear autoregressive external input (NARX) model with LS-SVM solving the model identification problem. Li et al. combine a generalized predictive control with the online LS-SVM and the proposed algorithm can recursively modify the model by adding a new data pair and deleting the least important data at each sampling period [20].

This article proposes an adaptive path following control method based on an improved online LS-SVM and an MPC algorithm for varying vessel dynamics. The improved parameter identification method is based on the weighted LS-SVM. This proposed method modifies the existing objective function of LS-SVM to increase parameter estimation accuracy, and proposes an index to speed up the rate of parameter convergence. Moreover, some abnormal identified parameters are ignored to avoid the bad performance if parameters are not satisfied within relevant predefined rational limitations. A sliding data window strategy combined with the proposed LS-SVM, namely λ-LS-SVM, is applied to realize online identification. The proposed adaptive method effectively improves the robustness and accuracy of path following control, especially under disturbances or uncertainties.

The remainder of this article is organized as follows. Path following modeling for a motion model and a predictive path following scheme is presented in Section 2. In Section 3, the LS-SVM based online parameter identification method is proposed. In Section 4, two simulation case studies are carried out to verify the performance of the proposed methods under disturbances and maneuverability changes. In the Section 5, the conclusions and future directions are presented.

2 | PATH FOLLOWING MODELING

In this section, a vessel motion model involving nonlinear Nomoto dynamics and rudder servo system is presented, and a path following scheme is proposed with Line-of-sight (LOS) guidance and MPC controller.

2.1 | Vessel motion model

In path following, the sway speed for an underactuated vessel always stays small and the surge speed can be deemed as constant in the body-fixed coordinate system [4,5,26]. To realize the adaptive control based on the online identified model while taking into account nonlinear characteristics of ship dynamics, a Nomoto second order nonlinear model is proposed to use as the vessel motion dynamic model which is as follows:

$$T_1T_2\dddot{\psi} + (T_1+T_2)\ddot{\psi} + \dot{\psi} + \beta\dot{\psi}^3 = K\left(\delta + T_3\dot{\delta}\right) + d_0, \tag{1}$$

where ψ is the heading and $\dot{\psi} = r$ in which r is the angular velocity of yaw; δ is the rudder angle; K is the Nomoto gain; T_1, T_2 and T_3 are maneuverability indices; $|d_0| \leqslant d_{\max}$ is a bias term due to disturbances and unmodeled dynamics [27]; β is a nonlinear coefficient. To use model (1), parameters T_1, T_2, T_3, K, d_0 and β should be identified.

Compared with other models that do not pertain to Nomoto ones, model (1) only has one input, namely the rudder angle, and one output, namely the heading. The involved two parameters, i.e., rudder angle δ and heading ψ can be obtained easily and precisely with angular transducer gyrocompass, respectively. Furthermore, in order to avoid bad identification performance when system input keeps unchanged, a persistent input excitation scheme is introduced in our simulations. Therefore, model (1) is selected for path following motion model of underactuated vessels in this article.

The rudder of a vessel is usually driven by a steering engine. Characteristics of the rudder servo system are modelled by [28]:

$$T_C\dot{\delta} + \delta = K_C\delta_C, \tag{2}$$

where δ_C is the helm order controlled by a course controller, δ is the actual rudder angle, K_C is the rudder gain (K_C=1 in this article), and T_C is the rudder time constant.

Here, a model is proposed that combines (1) and (2) for the path following of vessels. When setting system states and input as $\boldsymbol{x} = [\psi, r, \dot{r}, \delta]^T$ and $\boldsymbol{u}$=δ_C, (1) and (2) are transformed to the following state-space form:

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix} r \\ \dot{r} \\ g(\boldsymbol{x}) \\ \dfrac{1}{T_C}(\boldsymbol{u} - \delta) \end{bmatrix}, \tag{3}$$

where $g(\boldsymbol{x})$ is denoted by:

$$g(\boldsymbol{x}) = \frac{1}{T_1T_2}\Big[K\delta + \frac{KT_3}{T_C}(\boldsymbol{u} - \delta) + d_0 - (T_1+T_2)\dot{r} - r - \beta r^3\Big]. \tag{4}$$

2.2 | Predictive path following scheme with LOS

2.2.1 | Path following controller

The block diagram of the proposed adaptive predictive path following controller is shown in Figure 2. LOS guidance transfers predefined paths to objective headings. Model predictive controller calculates an optimal input, i.e., rudder angle, for the ship with the updated states, the updated Nomoto and rudder model, and the objective heading. λ-LS-SVM updates the parameters of the Nomoto and rudder model in real-time. Ship changes the heading according to the input of rudder angle with the influence of load changes, speed changing, disturbances, aging of mechanism, etc.

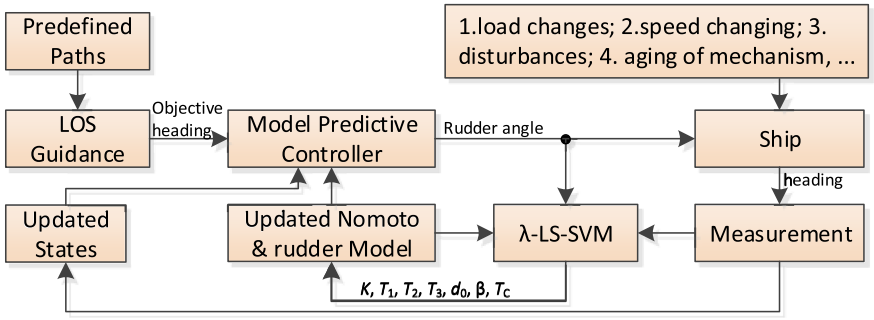


FIGURE 2 Block diagram of the proposed path following controller [Color figure can be viewed at wileyonlinelibrary.com]

2.2.2 | LOS guidance

In path following, given target waypoints, the reference path is generated as a sequence of straight lines that is usually adopted [29]. A typical reference path, as shown in Figure 3, can be considered as several straight line segments generated by connecting waypoints $P_n(x_n,y_n)$, $P_{n+1}(x_{n+1},y_{n+1})$, $P_{n+2}(x_{n+2},y_{n+2})$, etc. LOS guidance is widely used in path following because of its advantage on tracking the reference path precisely in a practical, feasible and efficient way [2,30]. In Figure 3, the inertial motion coordinate is defined as $\{n\}=\{x_n,y_n\}$, and the body-fixed coordinate system is defined as $\{b\}=\{x_b,y_b\}$. Under assumptions that the sway speed $v\approx 0$ and the surge speed u stays constant in $\{n\}$, an underactuated vessel tracks the reference path based on the difference between the heading angle ψ and the LOS angle ψ_{LOS} that can be calculated with a LOS point $P_{LOS}(x_{LOS},y_{LOS})$. The LOS points on the path is generated based on the cross tracking error e and a circle of radius $R_{LOS}=nL$ around O_b where L is the ship length [26].

The LOS point P_{LOS} is calculated by solving the following equations [2]:

$$(x_{LOS}-x_b)^2+(y_{LOS}-y_b)^2=R_{LOS}^2, \tag{5}$$

$$\frac{y_{LOS}-y_n}{x_{LOS}-x_n}=\frac{y_{n+1}-y_n}{x_{n+1}-x_n}. \tag{6}$$

Two solutions can be obtained by solving (6) of which the closer intersection to the current waypoint, i.e., P_{n+1} in Figure 3, is selected as P_{LOS}.

$\tilde{\psi}=\psi-\psi_P$ is defined as vessel relative heading to the path, where ψ_P is the path direction. Then, differential equations of e and $\tilde{\psi}$ can be denoted with $u=u_0$ and $v=0$ as [31]:

$$\dot{e}=u\sin\tilde{\psi}, \tag{7}$$

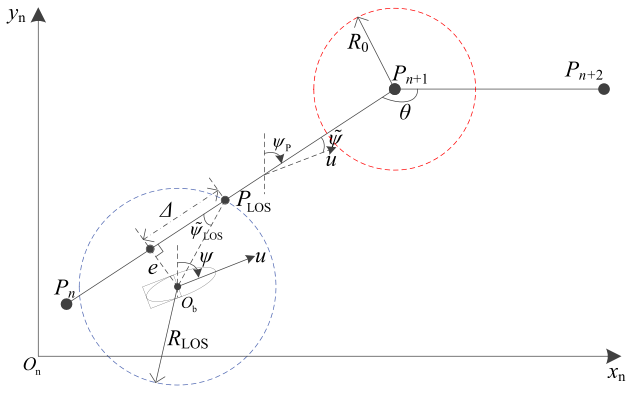


FIGURE 3 Path following scheme [Color figure can be viewed at wileyonlinelibrary.com]

$$\dot{\tilde{\psi}}=r. \tag{8}$$

The LOS angle $\tilde{\psi}_{LOS}$ can be denoted as:

$$\tilde{\psi}_{LOS}=-\arcsin\left(\frac{e}{R_{LOS}}\right), \tag{9}$$

where $\tilde{\psi}_{LOS}\in\left[-\frac{\pi}{2},\frac{\pi}{2}\right]$. To track the path P_nP_{n+1}, the angle $\tilde{\psi}$ is made to satisfy $\tilde{\psi}\to\tilde{\psi}_{LOS}$. Then, $\dot{e}$ can be obtained as follows:

$$\dot{e}=u_0\sin\tilde{\psi}_{LOS}=-\frac{u_0}{R_{LOS}}e. \tag{10}$$

The Lyapunov's second method is utilized to demonstrate $e\to 0$. The Lyapunov function is set as $V(e)=e^2$, then $\dot{V}(e)$ is obtained as follows:

$$\dot{V}(e)=2e\dot{e}=-\frac{u_0}{R_{LOS}}e^2. \tag{11}$$

It can be easily derived that $\dot{V}(e)\leqslant 0$ with $u_0>0$, and $\dot{V}(e)=0$ only when $e=0$. Therefore, (10) has global asymptotic stability, i.e., $e\to 0$ globally as $\tilde{\psi}\to\tilde{\psi}_{LOS}$.

To guarantee that there is always a real solution to (9), the LOS circle radius is set as [32]:

$$R_{LOS}=\begin{cases}3L, & \text{for } e\leqslant 3L,\\ e+L, & \text{otherwise.}\end{cases} \tag{12}$$

Another key point for path following is to switch to the next waypoint P_{n+2} relies on whether the vessel is within an acceptance circle around the current waypoint P_{n+1} or not. If the vessel position O_b satisfies (13), the waypoint will be changed to P_{n+2}.

$$(x_b-x_{n+1})^2+(y_b-y_{n+1})^2\leqslant R_0^2, \tag{13}$$

where R_0 stands for the acceptance circle radius. To guarantee that the solutions to (5) always exist, $R_{LOS}\geqslant R_0$ is needed. R_0 is usually set as a constant based on ship length, and R_0 is taken to equal $2L$ as [26].

2.2.3 | Nonlinear MPC controller for path following

MPC methods utilize a system model for trajectory prediction and optimization. (3) is taken as the system model. Considering that path following aims at making the cross error e, the ship relative heading $\tilde{\psi}$ and the rudder angle δ all converge to 0, the state-space equation (3) is transformed with (8) and (7) to:

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) = \begin{bmatrix} u_0 \sin(\tilde{\psi}) \\ r \\ \dot{r} \\ g(\boldsymbol{x}) \\ \frac{1}{T_C}(\boldsymbol{u} - \delta) \end{bmatrix}, \tag{14}$$

where $\boldsymbol{x} = [e, \tilde{\psi}, r, \dot{r}, \delta]^T$, $\boldsymbol{u} = \delta_C$.

For numerical simulations and implementation in practical applications, proper sampling is needed to obtain discrete-time dynamics for prediction. For the continuous-time model (14), the commonly used Runge-Kutta method is adopted for sampling [33]. The sampled model of (14) is shown as follows:

$$\hat{\boldsymbol{x}}(k+1) = f_d(\hat{\boldsymbol{x}}(k), \hat{\boldsymbol{u}}(k)). \tag{15}$$

To minimize the cross error and energy consumption, the errors between the state vector $\boldsymbol{x} = [e, \tilde{\psi}, r, \dot{r}, \delta]^T$ and the reference state vector $\boldsymbol{x}_r = [0, \tilde{\psi}_{LOS}, 0, 0, 0]^T$ should be minimized. Therefore, at each control step time k, the following quadratic cost function $J(k)$ is minimized:

$$J(k) = \sum_{i=1}^{N_P} \hat{\boldsymbol{x}}_e(k+i)^T \mathbf{Q} \hat{\boldsymbol{x}}_e(k+i) + \sum_{i=1}^{N_C} \hat{\boldsymbol{u}}(k+i-1)^T \mathbf{R} \hat{\boldsymbol{u}}(k+i-1), \tag{16}$$

where $\hat{\boldsymbol{x}}_e(k+i) = \hat{\boldsymbol{x}}(k+i) - \boldsymbol{x}_r(k+i)$ in which $\boldsymbol{x}_r(k+i)$ is the reference state vector at time step $k+i$; N_P stands for the length of the prediction horizon; $\mathbf{Q}$ and $\mathbf{R}$ are weighting matrices. Meanwhile, considering the limitations of the actuator, input constraints should be satisfied during path following as:

$$\begin{cases} (15), \\ \hat{\boldsymbol{u}}(k+i-1) = \hat{\boldsymbol{u}}(k+N_C-1),\ N_C < i \leqslant N_P, \\ \delta_{min} \leqslant \hat{\boldsymbol{u}}(k) \leqslant \delta_{max}, \end{cases} \tag{17}$$

where δ_{min} and δ_{max} are the limit values. where N_C stands for the length of the control horizon and satisfies $N_P \geqslant N_C$.

Therefore, at time step k ($k \geqslant 0$), the optimization problem needs to be solved:

$$\boldsymbol{u}^*(k) = \arg\min_{\boldsymbol{u}} J(k), \tag{18}$$

subject to (17).

Problem (18) is a nonlinear programming problem since cost function (16) and constraint (15) are nonlinear.

The algorithm of predictive path following is summarized in Algorithm 1.

Algorithm 1 Predictive path following

1: Initialize waypoints $\{P_1, P_2, ..., P_n\}$ and system states $\boldsymbol{x}(0)$, and set $k = 0$;
2: **while** ASV has not arrived at the destination **do**
3: Measure current states $\boldsymbol{x}(k)$, and calculate $\hat{\boldsymbol{x}}(k+i)$ with $\boldsymbol{u}(k+i-1)$ for $i = 1, 2, ..., N_P$ as (15);
4: Solve (18) and obtain the optimal control input sequence at k, i.e., $\hat{\boldsymbol{U}}^*(k) = \{\hat{\boldsymbol{u}}^*(k), \hat{\boldsymbol{u}}^*(k+1), ..., \hat{\boldsymbol{u}}^*(k+N_P-1)\}$;
5: Apply the first element $\hat{\boldsymbol{u}}^*(k)$ to vessel dynamics and set $\boldsymbol{u}(k) = \hat{\boldsymbol{u}}^*(k)$;
6: $k = k+1$;
7: **end while**

3 | LS-SVM BASED ADAPTIVE PATH FOLLOWING CONTROL

To make the path following control method proposed in Section 2 become adaptive to disturbances and uncertainties, in this section, an LS-SVM method is proposed for the parameter identification of the vessel motion model. Then, based on this adaptively identified model, an adaptive path following control approach is proposed with a sliding data window strategy and a model switching scheme. Furthermore, in order to avoid bad identification performance when system input keeps unchanged, a persistent input excitation scheme is introduced in our simulations.

3.1 | LS-SVM based parameter identification

LS-SVM can be used both for machine classifier and system identification [17,19]. This article focuses on the latter. Firstly, LS-SVM for function estimation is introduced, after which the parameter identification method with LS-SVM is elaborated on in detail.

3.1.1 | LS-SVM for function estimation

Given a training data set of N points $\{\boldsymbol{x}_k, y_k\}_{k=1}^N$ where $\boldsymbol{x}_k \in \mathbb{R}^n$ is the k^{th} input data and $y_k \in \mathbb{R}$ is the k^{th} output data. The regression model for the SVM is as follows:

$$y(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\phi}(\boldsymbol{x}) + b \tag{19}$$

where $\boldsymbol{w}$ is the weighting vector that can be infinite dimensional; $\boldsymbol{\phi}(\boldsymbol{x})$ is a nonlinear function that maps the

input space into a higher dimensional space; b is the bias. The LS-SVM optimization problem has an objective function as follows [17,18]:

$$\min_{\boldsymbol{w},\, b,\, e}\left\{J(\boldsymbol{w},\, e)=\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}+\frac{1}{2}\gamma\sum_{k=1}^{N}e_k^2\right\}, \tag{20}$$

subject to:

$$y_k=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_k)+b+e_k,\ k=1,\ \ ...,\ N, \tag{21}$$

where $e_k\in\mathbb{R}$ are the error variables defined by (21); γ is the positive real constant that determines the relative importance of the terms e_k. To solve the problem (20)–(21), a Lagrange function is defined as follows:

$$L(\boldsymbol{w},\, b,\, e,\, \alpha)=J(\boldsymbol{w},\, e)-\sum_{k=1}^{N}\alpha_k\left[\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_k)+b+e_k-y_k\right], \tag{22}$$

where $\alpha_k\in\mathbb{R}$ is a Lagrange multiplier. The relevant conditions for optimality of (22) are given by:

$$\begin{cases}\dfrac{\partial L}{\partial \boldsymbol{w}}=0\rightarrow\boldsymbol{w}=\displaystyle\sum_{k=1}^{N}\alpha_k\boldsymbol{\phi}(\boldsymbol{x}_k),\\ \dfrac{\partial L}{\partial b}=0\rightarrow\displaystyle\sum_{k=1}^{N}\alpha_k=0,\\ \dfrac{\partial L}{\partial e_k}=0\rightarrow\alpha_k=\gamma e_k,\\ \dfrac{\partial L}{\partial \alpha_k}=0\rightarrow\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}_k)+b+e_k-y_k=0,\\ k=1,\ \ ...,\ N.\end{cases} \tag{23}$$

In (23), $\alpha_k=\gamma e_k$ means the support values are nonzero and proportional to the errors, while many support values are zero in the classical SVM [17]. This feature of LS-SVM could cause sparseness and robustness problems [18]. To solve the two problems, weighted LS-SVM methods are widely applied [18,34]. In the weighted LS-SVM, objective function is changed to:

$$\min_{\boldsymbol{w},\, b,\, e}\left\{J(\boldsymbol{w},\, b,\, e)=\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}+\frac{1}{2}\gamma\sum_{k=1}^{N}v_ke_k^2\right\}, \tag{24}$$

where v_k is the weighting factor. After elimination of $\boldsymbol{w}$ and e from (23), the solution is obtained as follows:

$$\begin{bmatrix}0 & \boldsymbol{1}^{\mathrm{T}}\\ \boldsymbol{1} & \Omega+\boldsymbol{V}_\gamma\end{bmatrix}\begin{bmatrix}b\\ \boldsymbol{\alpha}\end{bmatrix}=\begin{bmatrix}0\\ \boldsymbol{Y}\end{bmatrix}, \tag{25}$$

where $\boldsymbol{1}=[1,1,\ ...,1]^{\mathrm{T}}$, $\boldsymbol{\alpha}=[\alpha_1,\alpha_2,\ ...,\alpha_N]^{\mathrm{T}}$, $\boldsymbol{Y}=[y_1,y_2,\ ...,y_N]^{\mathrm{T}}$, and Ω_{kl} stands for the item at the k^{th} row and l^{th} column of $\boldsymbol{\Omega}$, which follows Mercer's condition [18]:

$$\Omega_{kl}=K(x_k,\, x_l)=\boldsymbol{\phi}(x_k)\boldsymbol{\phi}(x_l),\ k,\, l=1,\ \ ...,\ N, \tag{26}$$

where $K(x_k,x_l)$ is a kernel function that can be chosen as linear kernel, polynomial kernel, RBF kernel or MLP kernel [35]. The linear kernel, i.e., $K(x_k,x_l)=x_kx_l$, is chosen in this article because of the need of parameter identification detailed in the next section. The diagonal matrix $\boldsymbol{V}_\gamma$ is given by:

$$\boldsymbol{V}_\gamma=\mathrm{diag}\left\{\frac{1}{\gamma v_1},\ \ ...,\ \frac{1}{\gamma v_N}\right\}. \tag{27}$$

The values of v_k can be defined in different patterns. In [18], v_k defined as follows:

$$v_k=\begin{cases}1, & \text{if } |e_k/\hat{s}|\leqslant c_1,\\ \dfrac{c_2-|e_k/\hat{s}|}{c_2-c_1}, & \text{if } c_1\leqslant|e_k/\hat{s}|\leqslant c_2,\\ 10^{-4}, & \text{otherwise,}\end{cases} \tag{28}$$

where c_1 and c_2 are the constants typically chosen as $c_1=2.5$ and $c_2=3$; $\hat{s}$ is the estimation of the standard deviation of the LS-SVM error variables e_k, which is as follows:

$$\hat{s}=\frac{\mathrm{IQR}}{2\times0.6745}, \tag{29}$$

where, the IQR is the difference between the 75th percentile and 25th percentile.

The resulting LS-SVM model for function estimation is as follows:

$$y(\boldsymbol{x})=\sum_{k=1}^{N}\left(\alpha_k\boldsymbol{x}_k^{\mathrm{T}}\right)\boldsymbol{x}+b, \tag{30}$$

where $\boldsymbol{\alpha}$ and b are the solution to (25).

3.1.2 | Parameter identification of the path following model with LS-SVM

To obtain parameters of a model that needs to be identified, a discrete model pattern is defined as follows:

$$y(\boldsymbol{x})=\theta^{\mathrm{T}}\boldsymbol{x}. \tag{31}$$

A forward difference method is utilized to discretize the Nomoto model (1) and rudder model (2). The n^{th} order forward difference $\Delta_h^n[f](x)$ of function $f(x)$ is defined as:

$$\Delta_h^n[f](x)=\sum_{i=0}^{n}(-1)^i\binom{n}{i}f(x+(n-i)h), \tag{32}$$

where $h>0$ is the spacing of difference; n is the order of forward difference; $\binom{n}{i}=\dfrac{n(n-1)\ ...(n-i+1)}{i!(n-i)!}$ is the

binomial coefficients with $\binom{n}{0}$ and $\binom{n}{n}$ equaling 0. It can be assumed that $f^{(n)}(x) \approx \frac{\Delta_h^n[f](x)}{h^n}$ when h is small enough [36]. Based on these, the Nomoto model (1) is transferred to:

$$\begin{aligned}\Delta_h^3[\psi](t) = \frac{1}{T_1T_2}\{&-(T_1+T_2)h\Delta_h^2[\psi](t) - h^2\Delta_h^1[\psi](t)\\ &-\beta\left[\Delta_h^1[\psi](t)\right]^3 + Kh^3\delta(t)\\ &+h^3d_0 + KT_3h^2\Delta_h^1[\delta](t)\}\end{aligned}. \tag{33}$$

The rudder model (2) is transferred to:

$$[\delta_C(t) - \delta(t)]h = T_C\Delta_h^1[\delta](t). \tag{34}$$

For (33), y_1, θ_1 and $\boldsymbol{x}_1$ with the pattern of (31) are defined as follows:

$$y_1 = \Delta_h^3[\psi](t), \tag{35}$$

$$\theta_1 = \frac{1}{T_1T_2}\begin{bmatrix}T_1+T_2\\ K\\ d_0\\ 1\\ \beta\\ KT_3\end{bmatrix}, \quad \boldsymbol{x}_1 = \begin{bmatrix}-h\Delta_h^2[\psi](t)\\ h^3\delta(t)\\ h^3\\ -h^2\Delta_h^1[\psi](t)\\ -[\Delta_h^1[\psi](t)]^3\\ h^2\Delta_h^1[\delta](t)\end{bmatrix}. \tag{36}$$

For (33), y_2, θ_2 and $\boldsymbol{x}_2$ with the pattern of (31) are also defined as follows:

$$y_2 = [\delta_C(t) - \delta(t)]h, \quad \theta_2 = T_C, \quad \boldsymbol{x}_2 = \Delta_h^1[\delta](t). \tag{37}$$

Compare (30) with (31), the solution of θ when $|b| \approx 0$ is obtained as follows:

$$\hat{\theta} = \sum_{k=1}^{N}(\alpha_k\boldsymbol{x}_k), \tag{38}$$

where $\hat{\theta}$ denotes the approximate value of θ. To guarantee $|b|$ to be small enough, (24) is updated as follows:

$$\min_{\boldsymbol{w},\,b,\,e} J(\boldsymbol{w},\,b,\,e) = \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + \frac{1}{2}\gamma\sum_{k=1}^{N}v_ke_k^2 + \frac{1}{2}\gamma b^2. \tag{39}$$

Therefore, the relevant conditions of (39) for optimality is accordingly as:

$$\begin{cases}\dfrac{\partial L}{\partial \boldsymbol{w}} = 0 \rightarrow \boldsymbol{w} = \sum\limits_{k=1}^{N}\alpha_k\phi(\boldsymbol{x}_k),\\ \dfrac{\partial L}{\partial b} = 0 \rightarrow \sum\limits_{k=1}^{N}\alpha_k = \gamma b,\\ \dfrac{\partial L}{\partial e_k} = 0 \rightarrow \alpha_k = \gamma v_k e_k,\\ \dfrac{\partial L}{\partial \alpha_k} = 0 \rightarrow \boldsymbol{w}^{\mathrm{T}}\phi(\boldsymbol{x}_k) + b + e_k - y_k = 0,\\ k = 1,\ \ldots,\ N\end{cases} \tag{40}$$

the solution is changed accordingly to:

$$\begin{bmatrix}-\gamma & \boldsymbol{1}^{\mathrm{T}}\\ \boldsymbol{1} & \Omega + \boldsymbol{V}_\gamma\end{bmatrix}\begin{bmatrix}b\\ \boldsymbol{\alpha}\end{bmatrix} = \begin{bmatrix}0\\ \boldsymbol{Y}\end{bmatrix}, \tag{41}$$

where $\boldsymbol{V}_\gamma = \mathrm{diag}\left\{\frac{1}{\gamma v_1},\ \ldots,\ \frac{1}{\gamma v_N}\right\}$. If $\left[-\gamma,\ \boldsymbol{1}^{\mathrm{T}},\ \boldsymbol{1},\ \Omega + \boldsymbol{V}_\gamma\right]$ is singular or very close to singular, a small changes can be taken to avoid no solution for $[b,\ \boldsymbol{\alpha}]$ with adding a term $10^{-8}\boldsymbol{I}_{N+1}$ to $\left[-\gamma,\ \boldsymbol{1}^{\mathrm{T}},\ \boldsymbol{1},\ \Omega + \boldsymbol{V}_\gamma\right]$ to be $\left[-\gamma + 10^{-8},\ \boldsymbol{1}^{\mathrm{T}},\ \boldsymbol{1},\ \Omega + \boldsymbol{V}_\gamma + 10^{-8}\boldsymbol{I}_N\right]$ as in [37], where $\boldsymbol{I}_N$ is an identity matrix with dimension N.

According to the solution of $\hat{\theta}$ in (38), $\hat{\theta}_1$ and $\hat{\theta}_2$ can be denote as follows:

$$\hat{\theta_1} = \sum_{k=1}^{N}(\alpha_{1k}\boldsymbol{x}_{1k}^{\mathrm{T}}),\ \hat{\theta_2} = \sum_{k=1}^{N}(\alpha_{2k}\boldsymbol{x}_{2k}^{\mathrm{T}}), \tag{42}$$

where $\boldsymbol{\alpha}_1 = [\alpha_{11}, \alpha_{12},\ \ldots, \alpha_{1N}]^{\mathrm{T}}$, $\boldsymbol{\alpha}_2 = \left[\alpha_{21},\ \alpha_{22},\ \ldots,\ \alpha_{2N}^{\mathrm{T}}\right]$ can be solved with (25).

To identify each parameter of model (36), algebraic transformations are just needed to θ_1, i.e., $K = \theta_1(2)/\theta_1(4)$, $T_3 = \theta_1(6)/\theta_1(2)$, $d_0 = \theta_1(3)/\theta_1(4)$, $\beta = \theta_1(5)/\theta_1(4)$. Considering that T_1 and T_2 can be exchanged in model (36), one of them can be specified if some conditions are given, e.g., if $T_1 > T_2$,

$$T_1 = (\theta_1(1) + \sqrt{[\theta_1(1)]^2 - 4\theta_1(4)})/(2\theta_1(4)),$$

$$T_2 = (\theta_1(1) - \sqrt{[\theta_1(1)]^2 - 4\theta_1(4)})/(2\theta_1(4)).$$

3.2 | Online LS-SVM based adaptive path following control

Adaptive path following control strategy in this article means that optimal inputs are calculated using recursively identified motion models (1) and (2) as the prediction models in MPC. The motion model recursive training data are generated online during the path following of the vessel. Considering that (33) would be

inaccurate if state data sampling time h is not small enough, and the control sampling time T_s can not be so small because of the limitations of actuator physical properties and the time for solving optimization problems, it is reasonable to set different values for h and T_s.

3.2.1 | Sliding data window strategy

Since the training sample set will become larger and larger if the old training data are not pruned, it is necessary to prune part of the old data when adding new training samples as [38]. Considering that the control period T_s is much longer than training data period h, it is not necessary to identify the model at every state data sampling time. Therefore, the sliding data window strategy is utilized to update the training data. The procedure of sliding data window switching, as shown in Figure 4, is as follows:

1. The sliding data window size is set as N_w, based on which parameters $\hat{\theta}_1(t_k)$ and $\hat{\theta}_2(t_k)$ can be identified with the LS-SVM at time t_k;
2. With new training data added into the sliding data window ceaselessly, the sliding data window size keeps increasing;
3. When the sliding data window size is equal to N_w+N_u at time t_{k+1}, the first N_u training data are deleted from the training data set, and new parameters $\hat{\theta}_1(t_{k+1})$ and $\hat{\theta}_2(t_{k+1})$ are identified with the new training data set;
4. Recursively conduct steps 1–3 until there is no new training data.

3.2.2 | Model switching scheme

The accuracy of parameter identification depends on the training data quality. It is possible to obtain outliers because of measurement errors. Therefore, it is necessary to improve the robustness of the LS-SVM. In the weighted LS-SVM (28), the weighting v_k is set small when the error $|e_k|$ is big. However, when the system model parameters changes, it could happen that a small amount of new training data generated with the changed model are mixed with a large amount of old training data generated by the unchanged system model in the sliding data window strategy. In this scenario, the unexpected identifying results could emerge because of the inconsistency of training data generated by different models. Moreover, it is beneficial for better control performance if the changed model parameters can be identified earlier. In this way, the key is the criterion for recognizing the changes of model parameters. Adopted from the fact that the magnitude of the identification errors increases denoted in [39], a parameter changing index λ_k is proposed as follows:

$$\lambda_k = \frac{1}{N_u}\left|\sum_{i=N_w-N_u+1}^{N_w} e_i^{k-1}\right|, \tag{43}$$

where e_i^k stands for the i^{th} identification error generated with the identified parameters at time k, which is defined as (21). The index $\lambda_k>0$ describes the average error of new updated data subset in the training set with the old identified model at time $k-1$. If λ_k is large, it means that the new training data subset is generated by different models, i.e., the model parameters have changed; otherwise, it means that the new training data subset is generated by the same model or the slightly varied model. If model parameter changes are detected, then weighting is added for the new data to improve the parameter convergence rate. The new weighting $\bar{v}_k$ are defined as follows:

$$\bar{v}_k = \begin{cases} c_3 v_k, & \text{if } k < N_w/2, \\ c_4 v_k, & \text{otherwise}, \end{cases} \tag{44}$$

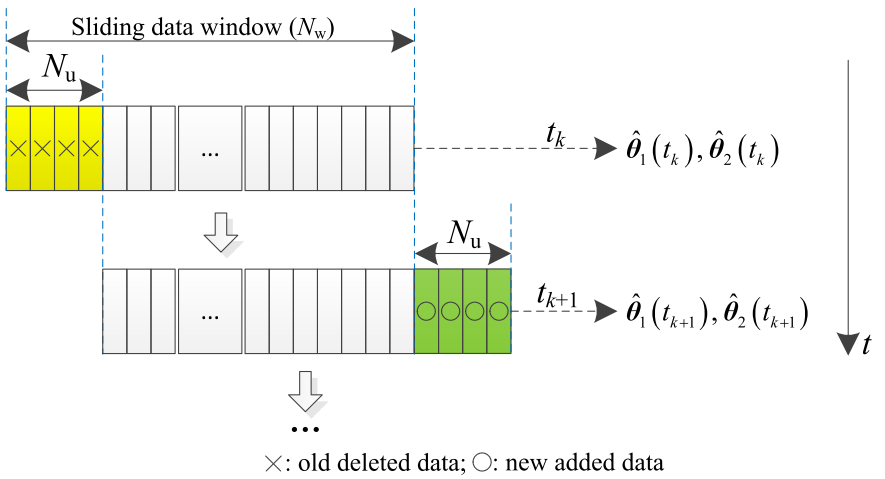


FIGURE 4 Sliding data window strategy [Color figure can be viewed at wileyonlinelibrary.com]

where v_k is defined as (28); c_3 and c_4 are constants with $0<c_3<1$ and $c_4>0$.

An identified model can not be used if this identified parameters are not satisfied vessel's maneuverability demand, for instance, when $K<0$ or $T<0$. Therefore, identified parameters can be utilized to judge the reliability of an identified model.

Based on the foregoing in this section, an improved weighted LS-SVM algorithm, termed λ-LS-SVM, for path following is proposed as Algorithm 2.

Algorithm 2 λ-LS-SVM for path following

1: Initialize model parameters $\Theta_0 = \left\{\hat{\boldsymbol{\theta}}_1(0), \hat{\boldsymbol{\theta}}_2(0)\right\}$ and set $k = 1$;
2: **while** ASV has not arrived at the destination **do**
3: Obtain training data set $\mathcal{T}_k$ with sliding data window strategy, and calculate λ_k^1 of $\hat{\boldsymbol{\theta}}_1(k)$ and λ_k^2 of $\hat{\boldsymbol{\theta}}_2(k)$ by (43);
4: Calculate $\hat{\boldsymbol{\theta}}_1(k)$ and $\hat{\boldsymbol{\theta}}_2(k)$ by the weighted LS-SVM method by (42) with weighting $v_n^1(k)$ and $v_n^2(k)$ $(n = 1, 2, ..., N_w)$ calculated as (28);
5: Calculate λ_k^1 and λ_k^2 as (43);
6: **if** $\lambda_k^1 < \lambda_c^1$ **and** $\lambda_k^2 < \lambda_c^2$ **then**
7: $\hat{\boldsymbol{\theta}}_1(k)$ and $\hat{\boldsymbol{\theta}}_2(k)$ keep unchanged;
8: **else if** $\lambda_k^{1(2)} > \lambda_c^{1(2)}$ **then**
9: The new weighting $\bar{v}_n^{1(2)}(k)$ of identification errors are obtained with $v_n^{1(2)}(k)$ as (44);
10: **end if**
11: **if** $K < 0$ **or** $T_1 < 0$ in $\hat{\boldsymbol{\theta}}_1(k)$ **then**
12: $\hat{\boldsymbol{\theta}}_1(k) = \hat{\boldsymbol{\theta}}_1(k-1)$;
13: **end if**
14: Set $\Theta_k = \left\{\hat{\boldsymbol{\theta}}_1(k), \hat{\boldsymbol{\theta}}_2(k)\right\}$ as predictive models for MPC at k;
15: $k = k + 1$;
16: **end while**

3.2.3 | Persistent input excitation

Input excitation is important on the performance of parameter identification. When a vessel is tracking a straight line, the rudder control input δ_C is mostly constant if environmental disturbances d_0 are also constant. White noise signals are usually used for input excitation because they have a flat spectrum over the range of frequencies [40]. Persistent input excitation is used in this article adding white Gaussian noises to the control input. However, considering that the added white Gaussian noises may affect the system performance, their magnitudes should be kept small. The control inputs added with persistent input excitation u_{per} is as follows:

$$u_{per}(i) = u_{ori}(i) + u_{add}(i), \quad i = 1, 2, \ ..., \tag{45}$$

where, $u_{ori}(i)$ is the original control input, and $u_{add}(i)$ is the added excitation input. The added excitation input is independently and identically distributed, i.e., $u_{add}(i) \sim \mathcal{N}(0, \sigma^2)$.

4 | CASE STUDY

The proposed approach is applied to path following control in different scenarios including aging of rudder equipment, variable current and changes of the vessel maneuverability. The simulation experiments in these scenarios are implemented based on a motion model from a scale model ship in our laboratory. The main geometric parameters of the model ship are: ship length L=0.95 m, ship breadth B=0.24 m, ship mass M=5.40 kg, nominal speed U=0.80 m/s. The initial parameters in (3) with the surge speed u_0=0.80 m/s are: K=0.5060 s^{-1}, T_1=1.2481 s, T_2=0.1245 s, T_3=−0.0757 s, d_0=−1.2370°/s, $\beta = 0.0081$ s^2 and T_C=0.1000 s. The simulation parameter is set to: the state data sampling time h=0.01 s and the control sampling time T_s=0.5 s; N_w=1200, N_u=50 for sliding data window; $\gamma=10^{15}$, c_3=0.01, c_4=0.99, $\lambda_c^1 = 10^{-5}$ and $\lambda_c^2 = 10^{-2}$ for λ-LS-SVM; N_P=10, N_C=8, $-30° \leqslant \delta_C \leqslant 30°$, **Q**=diag [1,1,0.01,0.01,0.001] and **R**=0.01 for MPC controller. The simulation experiments for the following three cases are conducted on the platform of MathWorks Matlab R2016b.

4.1 | Case 1: aging of rudder equipment

The ship steering gear system consists of several electro-hydraulic steering subsystems: telemotor position servo and rudder servo actuator [28]. These components lead to time-delay and non-synchronization feature between rudder command and real rudder angle as denoted in (2). In this case, it is assumed that the time-delay constant T_C changes from T_{C0}=0.1 s to T_{C1}=1.0 s at time T = 12.00 s because of aging or maintenance. The waypoints of path are set to: (1,1), (15,1), (15,10), (29,10). The path following performance and cross error e of adaptive control with the λ-LS-SVM, the adaptive control with weighted LS-SVM and the non-adaptive control method (traditional MPC method with LOS guidance) is shown in Figure 5 and Figure 6, respectively. To verify the performance of path following with different control methods, an evaluation index e_a, i.e., average cross error (ACE), for path following performance is defined as:

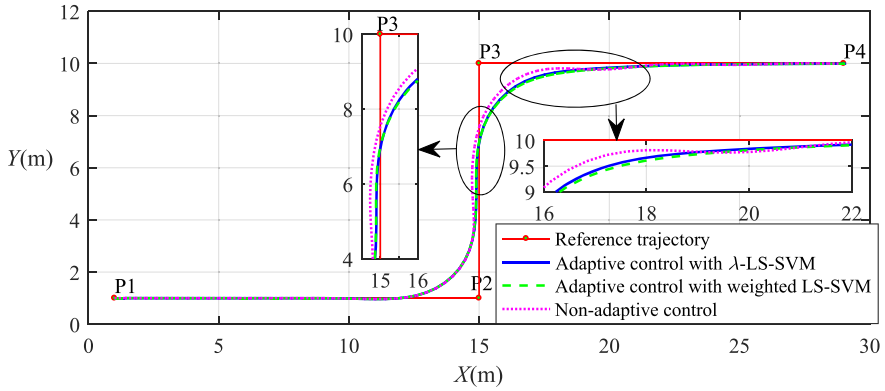


FIGURE 5 Path following performance of rudder aging [Color figure can be viewed at wileyonlinelibrary.com]

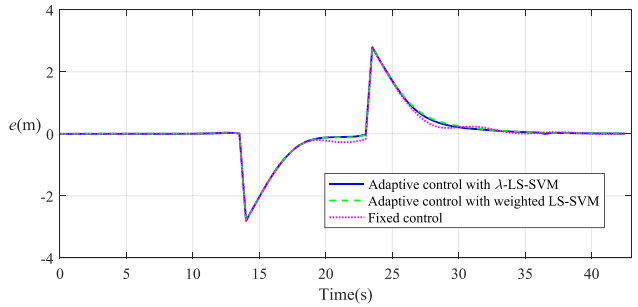


FIGURE 6 Error comparison during path following of rudder aging [Color figure can be viewed at wileyonlinelibrary.com]

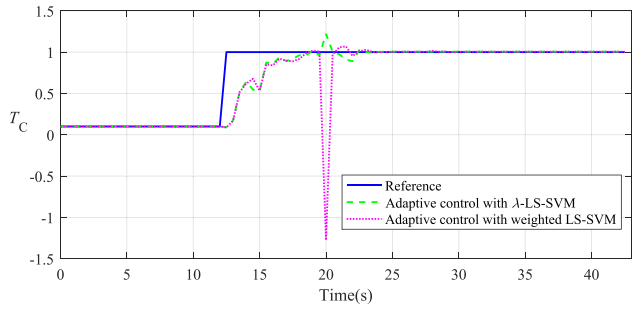


FIGURE 7 T_C variation during path following of rudder aging [Color figure can be viewed at wileyonlinelibrary.com]

parameters changes. Note that the model accuracy hardly has an impact on the tracking performance when the objective heading keeps unchanged. The differences of e_a between the λ-LS-SVM based adaptive control method and the other two methods are not big because most of the time during path following the objective heading does not need to be changed.

Apart from ACE, the cross error e convergence rate also plays an important role in evaluating the performance of path following. Specifically, in this article, the moment, t_c, when $|e|$ keeps below 0.05 m after the model parameters change is an index of the cross error convergence rate. In this case, the values of t_c are 35.5 s, 36.0 s and 38.5 s for the λ-LS-SVM based adaptive control, the weighted LS-SVM based adaptive control and the non-adaptive control method, respectively. It can be seen that the λ-LS-SVM has the fastest cross error convergence rate.

From Figure 7, parameter T_C converges to reference value in the finite time and T_C does not appear some outliers with λ-LS-SVM, while the weighted LS-SVM has generated some abnormal values. In Figure 8, all λ_k^2 for $\lambda_k^2 > \lambda_c^2$ are in a transition training data area, where the training data set for identification consists of two different models. It illustrates that λ_k^2 can become the indication of model changes.

$$e_a = \frac{1}{N_T}\sum_{i=1}^{N_T}|e(i)|, \tag{46}$$

where N_T is the total number of steps, and $e(i)$ is the cross error e at time i after the model parameters change, which is shown in Figure 3. The smaller e_a is, the better the performance will be. In this case, ACE values e_a for the λ-LS-SVM based adaptive control, the weighted LS-SVM based adaptive control and the non-adaptive control method are 0.495 m, 0.508 m and 0.502 m, respectively. It can be seen that λ-LS-SVM based adaptive control has the best performance with the smallest e_a when model

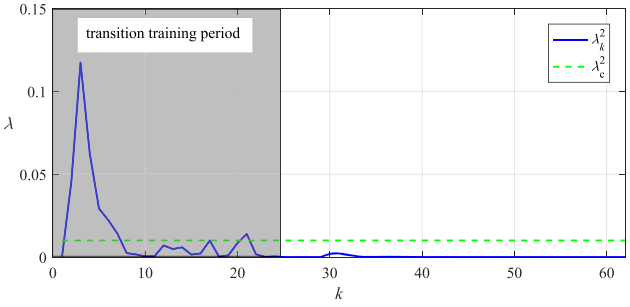


FIGURE 8 λ_c^2 variation during path following of rudder aging [Color figure can be viewed at wileyonlinelibrary.com]

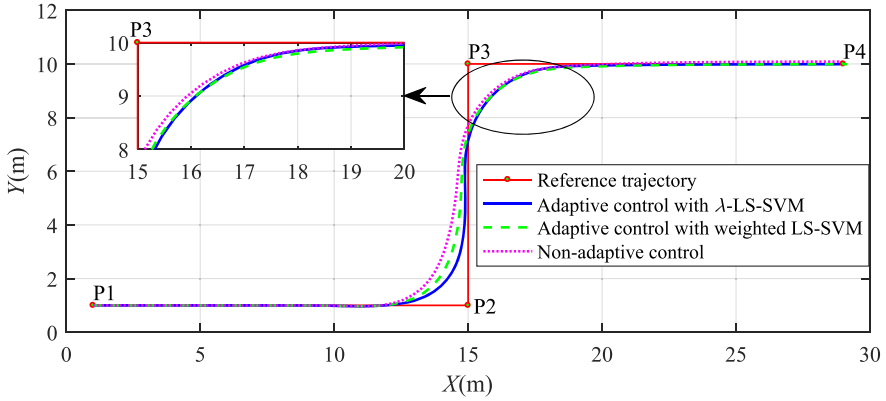


FIGURE 9 Path following performance of variable current [Color figure can be viewed at wileyonlinelibrary.com]

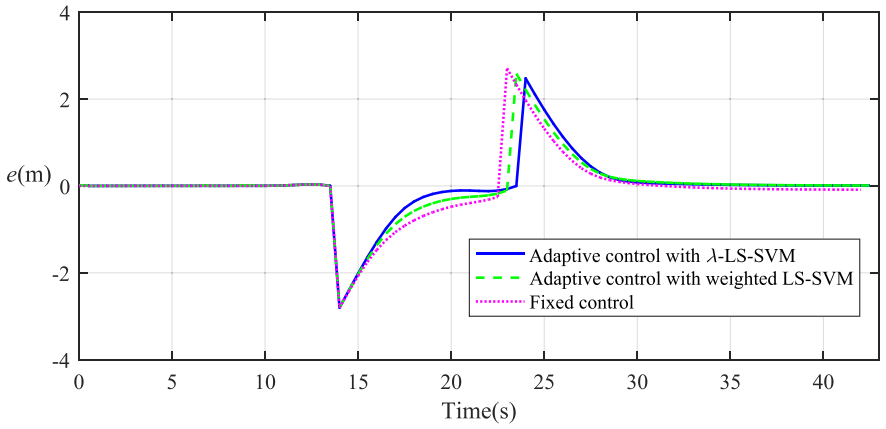


FIGURE 10 Error comparison during path following of variable current [Color figure can be viewed at wileyonlinelibrary.com]

4.2 | Case 2: variable current

Current has an effect on the ship maneuverability because of d_0 changing in model (1). Usually, current can be thought as constant in the inertial motion coordinate system during a period of time [7]. However, the constant current can also change the ship maneuverability in varying degrees because the direction of current changes in the body-fixed coordinate system if ship heading changes. Hence, it is necessary to detect the variation of d_0 deduced by current during path following. In this case, it is assumed that the parameter d_0 changes from d_{00}=−1.2370°/s to d_{01}=−4.0000°/s at time T=12.00 s. The ship can track the reference trajectory with the λ-LS-SVM based adaptive control method better than that with the weighted LS-SVM based adaptive control and the non-adaptive control method, which is shown in Figure 9 and Figure 10. ACE values e_a for the λ-LS-SVM based adaptive control, the weighted LS-SVM based adaptive control and the non-adaptive control method are 0.427 m, 0.488 m and 0.544 m, respectively. It can be seen that λ-LS-SVM based adaptive control has the best performance in term of the cross error. In this case, the values of t_c are 32.0 s, 33.5 s for the λ-LS-SVM based adaptive control and the weighted LS-SVM based adaptive control method, while the non-adaptive control cannot keep $|e|$ <0.05 m during path following. It can be seen that the λ-LS-SVM has a faster cross error convergence rate than the weighted LS-SVM based adaptive control and the non-adaptive control method.

Similarly with Case 1, system model parameter d_0 can be identified with λ-LS-SVM during d_0 varying, and can keep at a stable value when system model does not vary, shown in Figure 11. In Figure 12, all λ_k^1 for $\lambda_k^1 > \lambda_c^1$ are in a transition training data area. It also illustrates that the λ_k^2 can become the indication of model changes.

4.3 | Case 3: change of maneuverability

In this case, it is considered that there exist errors and outliers in training data set because of sensor measurement errors and malfunction. The maneuverability model parameters $\{K,T_1,T_2,T_3,d_0,\beta,T_C\}$ change from {0.5900, 0.9526, 0.0247, 0.2215, −1.2370, 0.0001, 0.1000} to {6.0000, 3.0000, 1.0000, −0.6000, −4.0000, 0.0020, 0.5000} at time T=18.00 s. Waypoints of a path named *Path* 1 are set to: (1,1), (12,1), (17,13), (28,13), (33,25). Measurement errors are denoted by Gaussian distribution

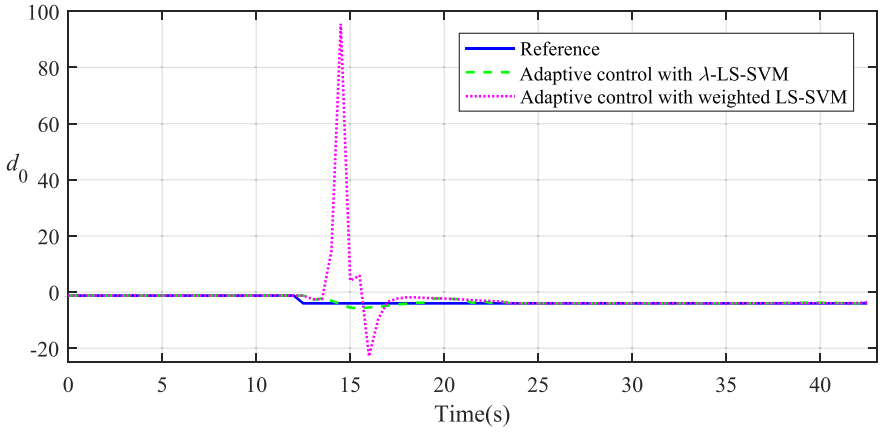


FIGURE 11 d_0 variation during path following of variable current [Color figure can be viewed at wileyonlinelibrary.com]

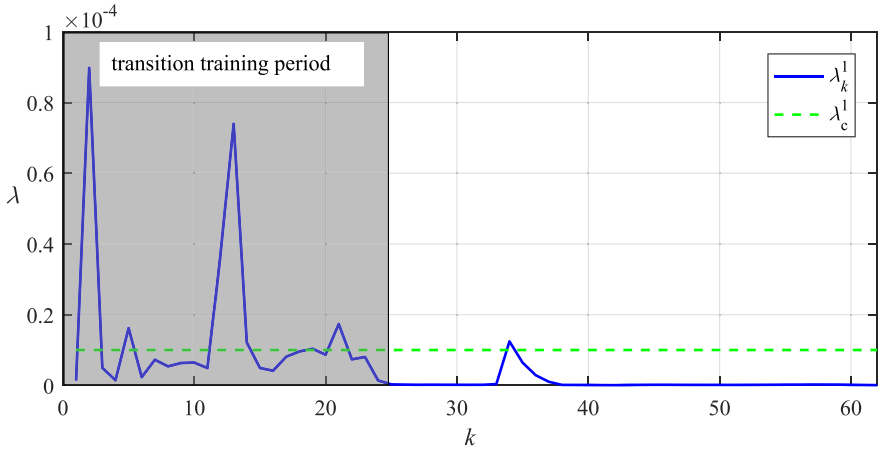


FIGURE 12 λ_c^1 variation during path following of variable current [Color figure can be viewed at wileyonlinelibrary.com]

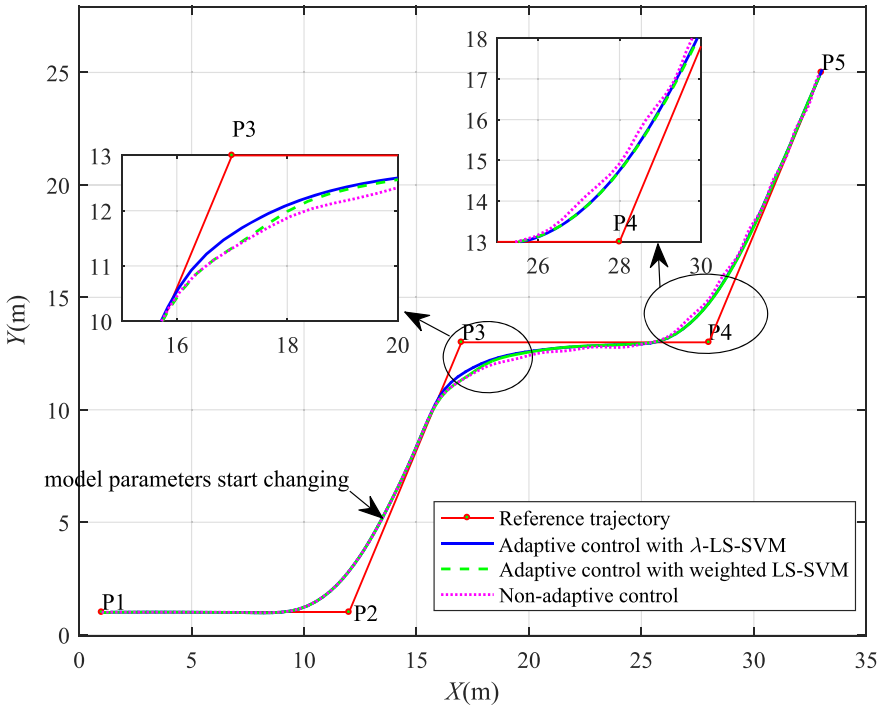


FIGURE 13 Path following performance of maneuverability change [Color figure can be viewed at wileyonlinelibrary.com]

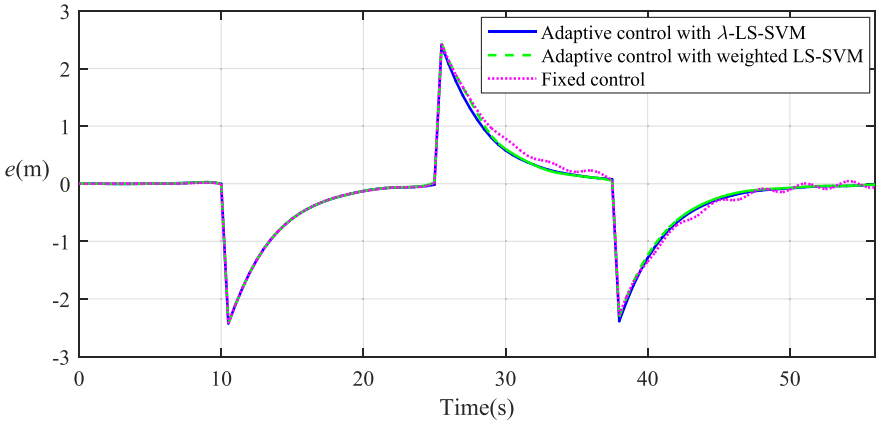


FIGURE 14 Error comparison during path following of maneuverability change [Color figure can be viewed at wileyonlinelibrary.com]

whose mean μ=0 and standard deviation σ=0.2. The outliers are set as Figure 16.

The path following performance and error comparison of adaptive control with the λ-LS-SVM, adaptive control with the weighted LS-SVM and the non-adaptive control is shown Figure 13 and Figure 14. It can be found that adaptive control with λ-LS-SVM has less tracking error (e_a=0.449 m) than the others (e_a=0.451 m for adaptive control with the weighted LS-SVM and e_a=0.502 m for the non-adaptive control) from Figure 14, and it is proven with model parameters identification results shown in Figure 15. Similar to Case 2, the values of t_c are 52.0 s, 51.5 s for the λ-LS-SVM based adaptive control and the weighted LS-SVM based adaptive control method, while the non-adaptive control method cannot keep $|e|$<0.05 m during path following. It can be seen that the λ-LS-SVM control and the weighted LS-SVM based adaptive control method as a faster cross error convergence rate than the non-adaptive control method.

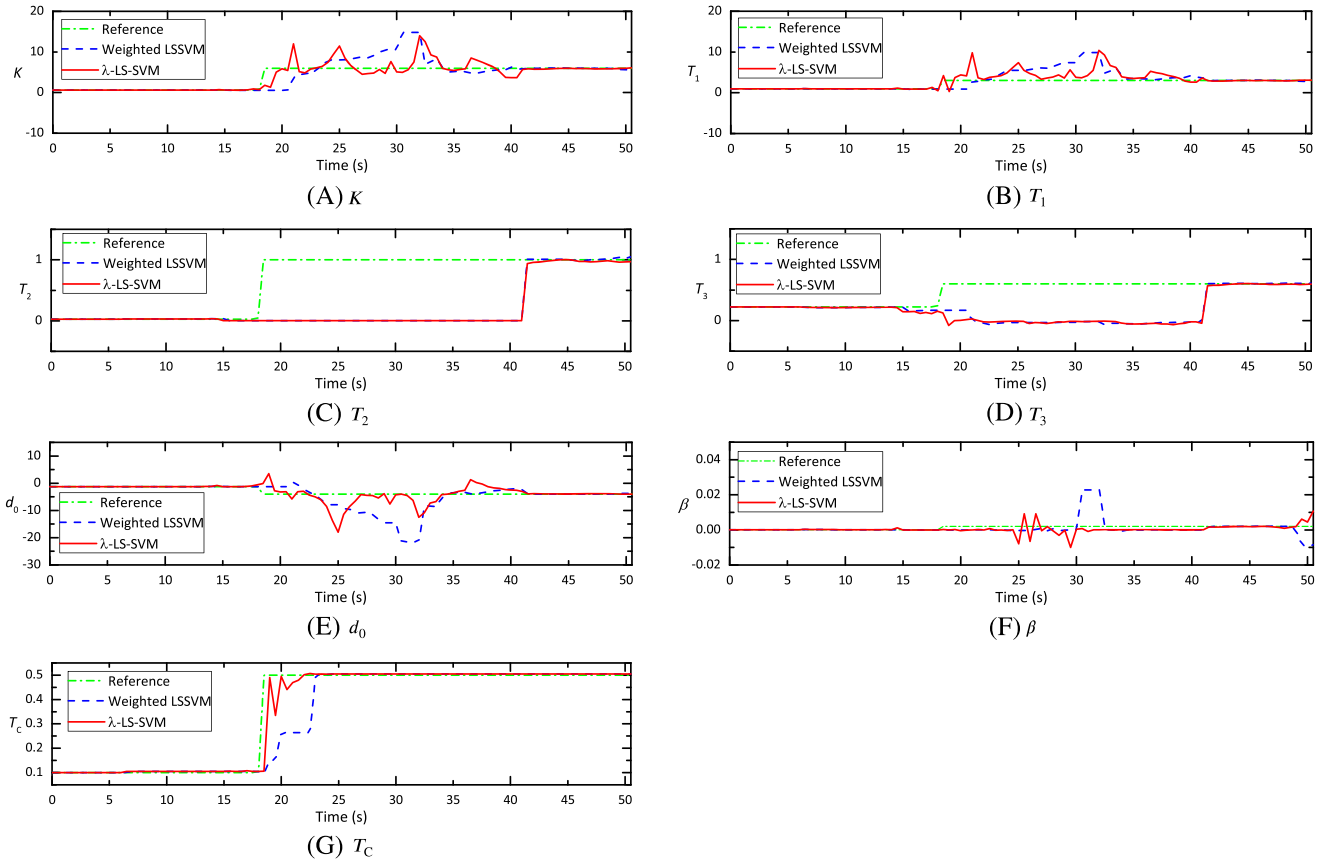


FIGURE 15 Identified parameters comparison between two adaptive control methods during path following of maneuverability change [Color figure can be viewed at wileyonlinelibrary.com]

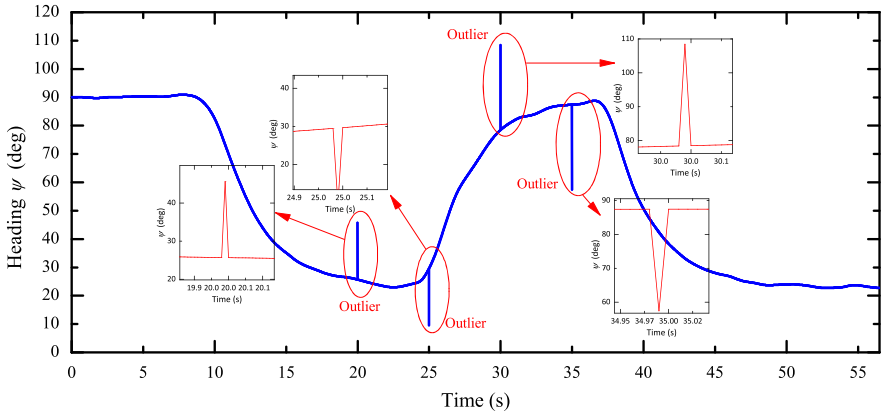


FIGURE 16 Ship heading during path following of maneuverability change [Color figure can be viewed at wileyonlinelibrary.com]

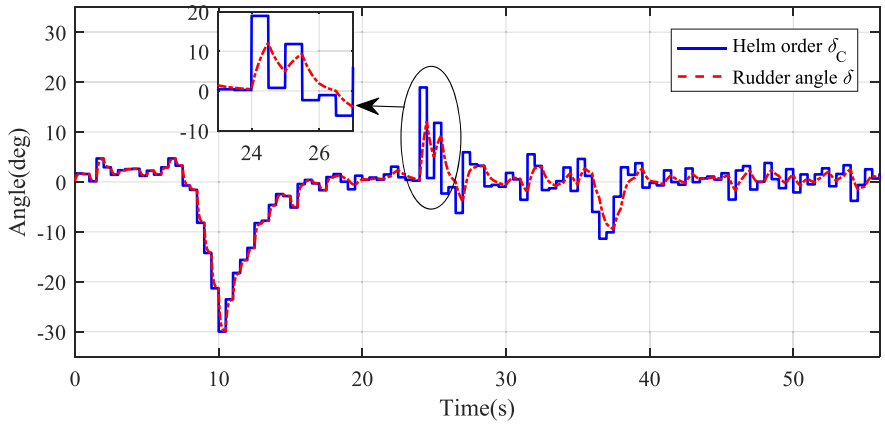


FIGURE 17 Ship helm order and rudder angle during path following of maneuverability change [Color figure can be viewed at wileyonlinelibrary.com]

Generally, the λ-LS-SVM can make the model parameters converge faster and have less fluctuations especially for T_C. Moreover, ship heading and rudder performance are shown in Figure 16 and 17, respectively. From Figure 13 and Figure 16, it can be found that the outliers have trivial effects on the performance of path following using adaptive control methods. In Figure 17, the rudder angle δ values are satisfied with the relevant limitation.

4.4 | Selection of λ_c value

From Figure 8 and Figure 12 in Case 1 and Case 2, it can be found that λ_k^1 or λ_k^2 becomes larger when the model parameters change than that when the model parameters keep unchanged. λ_c^1 or λ_c^2 is used to measure whether the parameters change or not. The magnitudes of λ_k are different according to different models, for instance, the magnitudes

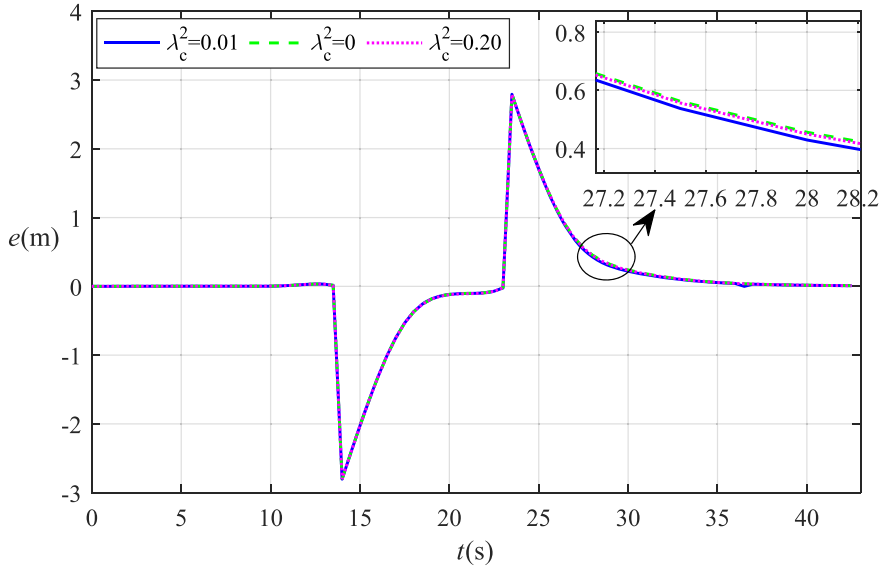


FIGURE 18 Cross error e with different λ_c in Case 1 [Color figure can be viewed at wileyonlinelibrary.com]

of λ_k^1 of model (1) and λ_k^2 of model (2) are different. Therefore, λ_c should be assigned according to its relevant model. If λ_c is set too large, the model parameters change cannot be detected; if λ_c is set too small, the model parameters change could be detected wrongly when the model parameters does not change actually. In Case 1, the cross error e with different λ_c^2 is shown in Figure 18. The values of e_a are 0.495 m, 0.501 m and 0.500 m with λ_c^2 equal to 0.01, 0 and 0.20, respectively. The values of t_c are 35.5 s, 36.0 s and 35.5 s with λ_c^2 equal to 0.01, 0 and 0.20, respectively. Hence, $\lambda_c^2 = 0.01$ is selected for Case 1. Considering that λ_c is a threshold of λ_k to detect the change of model parameters, λ_c should be a value to distinguish the model parameters changing between 0 and maximum of λ_k.

5 | CONCLUSIONS AND FUTURE RESEARCH

The performance of model-based path following control may be degraded when vessel dynamics vary if the controller for path following uses the same motion model all the time. An adaptive controller for path following is studied in this article based on online LS-SVM and MPC algorithms. An improved online LS-SVM identification method is proposed based on weighted LS-SVM in order to speed up the rate of parameter convergence. The objective function of LS-SVM is modified to decrease the error of parameter estimation. An index λ for LS-SVM is designed to detect the changes of model and speed up the rate of model parameter convergence. A sliding data window strategy combined with the online LS-SVM is used to realize the online parameter identification. The path following controller is designed based on the LOS and MPC that utilizes the updated nonlinear 2nd order Nomoto model with the online parameter identification method. The simulation results show that the proposed λ-LS-SVM method can speed up the rate of parameter convergence, improve the tracking accuracy of path following effectively, and make the cross error converge faster when the vessel dynamics change.

Considering that all the experiments in this article have been conducted in the simulation environment, some actual experiments should be done to make the verification of the proposed method more convincing in the future study. Moreover, the proposed adaptive control method is designed with the assumption that all of system states can be measured. However, it could be difficult to obtain all the system states in real-time. Therefore, for future work, it is of interest to design an adaptive controller combined with a state observer that can provide updated information on system states and environmental disturbances. In [41], the vessel position error is constrained with the proposed error-constrained line-of-sight path following method. Hence, the output constraints should be considered to improve the safety of vessel path following in the future work.

ACKNOWLEDGEMENTS

This research is supported by the China Scholarship Council (201506950053), the LIESMARS Special Research Funding and the China Postdoctoral Science Foundation (2018M632923).

ORCID

Chenguang Liu https://orcid.org/0000-0002-4130-1243

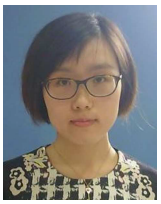
REFERENCES

1. M. Breivik and T. I. Fossen, Path following for marine surface vessels, Inproceedings of the Techno-Ocean Conference, Kobe, Japan, 2004, pp. 2282–2289.
2. T. I. Fossen, M. Breivik, and R. Skjetne, *Line-of-sight path following of underactuated marine craft*, Girona, Spain, 2003, pp. 244–249.
3. K. D. Do, Z. P. Jiang, and J. Pan, *Robust adaptive path following of underactuated ships*, Automatica **40** (2004), no. 6, 929–944.
4. Z. Li, J. Sun, and S. R. Oh, *Design, analysis and experimental validation of a robust nonlinear path following controller for marine surface vessels*, Automatica **45** (2009), no. 7, 1649–1658.
5. S. R. Oh and J. Sun, *Path following of underactuated marine surface vessels using line-of-sight based model predictive control*, Ocean Eng. **37** (2010), no. 2, 289–295.
6. W. Caharija and other, *Relative velocity control and integral line of sight for path following of autonomous surface vessels: Merging intuition with theory*, J. Eng. Mar. Environ. **228** (2014), no. 2, 180–191.
7. H. Zheng, R. R. Negenborn, and G. Lodewijks, *Predictive path following with arrival time awareness for waterborne AGVs*, Transp. Res. Part C: Emerg. Technol. **70** (2016), 214–237.
8. M. Fu, *A cross-coupling control approach for coordinated formation of surface vessels with uncertain disturbances: a cross-coupling control for coordinated formation*, Asian Journal of Control **20** (2018), no. 6, 1–10.
9. R. Skjetne, Ø. N. Smogeli, and T. I. Fossen, *A nonlinear ship manoeuvering model: Identification and adaptive control with experiments for a model ship*. Modeling, Identification and Control **25** (2004), no. 1, 3–27.
10. Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, *Adaptive configuration control of multiple UAVs*, Control. Eng. Pract. **21** (2013), no. 8, 1043–1052.
11. P. A. Ioannou and J. Sun, *Robust Adaptive Control*, Dover Publications, New York, 2012.
12. S. H. Mousavi and A. Khayatian, *Dead-zone model based adaptive backstepping control for a class of uncertain saturated systems*, Asian J. Control **18** (2016), no. 4, 1395–1405.

13. C. Cortes and V. Vapnik, *Support-Vector Networks*, Mach. Learn. **20** (1995), no. 3, 273–297.
14. V. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.
15. C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, 1995.
16. J. Sun, C. Liu, and N. Liu, *Data-driven adaptive critic approach for nonlinear optimal control via least squares support vector machine*, Asian J. Control **20** (2018), no. 1, 104–114.
17. J. A. K. Suykens and J. Vandewalle, *Least squares support vector machine classifiers*, Neural. Process. Lett. **9** (1999), no. 3, 293–300.
18. J. A. K. Suykens and other, *Weighted least squares support vector machines: robustness and sparse approximation*, Neurocomputing **48** (2002), no. 1, 85–105.
19. H. Tang and other, *Online weighted LS-SVM for hysteretic structural system identification*, Eng. Struct. **28** (2006), no. 12, 1728–1735.
20. L. Li, H. Su, and J. Chu, *Generalized predictive control with online least squares support vector machines*, Acta Automatica Sinica **33** (2007), no. 11, 1182.
21. K. D. Do and J. Pan, *Underactuated ships follow smooth paths with integral actions and without velocity measurements for feedback: theory and experiments*, IEEE Trans. Control Syst. Technol. **14** (2006), no. 2, 308–322.
22. E. W. Mcgookin and other, *Experimental results from supply ship autopilot optimization using genetic algorithms*, Journal of the Institute of Meas. Control **22** (2000), no. 2, 141–178.
23. D. Q. Mayne and other, *Constrained model predictive Stability and optimality*, Automatica **36** (2000), no. 6, 789–814.
24. D. Shi and Z. Mao, *Multi-step control set-based nonlinear model predictive control with persistent disturbances*, Asian J. Control **21** (2019), no. 2, 868–878.
25. Y. Shi and other, A nonlinear model predictive control based on least squares support vector machines NARX model, proceedings of the International Conference on Machine Learning and Cybernetics, Hong Kong, China, 2007, 721–725.
26. L. Moreira, T. I. Fossen, and C. G. Soares, *Path following control system for a tanker ship model*, Ocean Eng. **34** (2007), no. 14, 2074–2085.
27. T. I. Fossen and A. M. Lekkas, *Direct and indirect adaptive integral line-of-sight path-following controllers for marine craft exposed to ocean currents*, Int. J. Adapt Control Signal Process. **31** (2017), no. 4, 445–463.
28. J. Velagic, Z. Vukic, and E. Omerdic, *Adaptive fuzzy ship autopilot for track-keeping*, Control. Eng. Pract. **11** (2003), no. 4, 433–443.
29. T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, John Wiley & Sons, West Sussex, 2011.
30. E. W. McGookin and other, *Ship steering control system optimisation using genetic algorithms*, Control. Eng. Pract. **8** (2000), no. 4, 429–443.
31. R. Skjetne and T. I. Fossen, Nonlinear maneuvering and control of ships, proceedings of the MTS/IEEE Oceans Conference and Exhibition, Honolulu, USA, 2001, pp. 1808–1815.
32. C. Liu and other, *Predictive path following based on adaptive line-of-sight for underactuated autonomous surface vessels*, J. Mar. Sci. Technol. **23** (2018), no. 3, 483–494.
33. H. Zheng, R. R. Negenborn, and G. Lodewijks, Trajectory tracking of autonomous vessels using model predictive control. *IFAC Proceedings Volumes*, vol. **47**, Cape Town, South Africa, 2014, no. 3, pp. 8812–8818.
34. C. Lin and S. Wang, *Fuzzy support vector machines*, IEEE Trans. Neural Netw. **13** (2002), no. 2, 464–471.
35. T. V. Gestel and other, *Benchmarking least squares support vector machine classifiers*, Mach. Learn. **54** (2004), no. 1, 5–32.
36. P. E. Gill and other, *Computing forward-difference intervals for numerical optimization*, SIAM J. Sci. Stat. Comput. **4** (1983), no. 2, 310–321.
37. Y. Zhao and J. Sun, *Recursive reduced least squares support vector regression*, Pattern Recogn. **42** (2009), no. 5, 837–842.
38. J. C. Yin, L. D. Wang, and N. N. Wang, *A variable-structure gradient RBF network with its application to predictive ship motion control*, Asian J. Control **14** (2012), no. 3, 716–725.
39. K. S. Narendra and J. Balakrishnan, *Improving transient response of adaptive control systems using multiple models and switching*, IEEE Trans. Autom. Control **39** (1994), no. 9, 1861–1866.
40. D. Wang and A. Haldar, *Element-level system identification with unknown input*, J. Eng. Mech. **120** (1994), no. 1, 159–176.
41. Z. Zheng, L. Sun, and L. Xie, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **48** (2017), no. 10, 1794–1805.

AUTHOR BIOGRAPHIES

Chenguang Liu is an assistant professor in the National Engineering Research Center for Water Transport Safety, Wuhan University of Technology, Wuhan, China. He received his PhD degree and M.S. degree in the School of Energy and Power Engineering, Wuhan University of Technology, China in 2013 and 2017, respectively. He finished his post-doctoral research in Wuhan University in 2019. His current research interests include ship intelligence, ship motion control, and model predictive control.



Huarong Zheng is a post-doctoral research fellow working on coordination and control of unmanned systems at the College of Control Science and Engineering, Zhejiang University, Hangzhou, China. She received the Ph.D. degree in 2016 from the Department of Marinetime & Transport Technology, Delft University of Technology, Delft, the Netherlands, and the B.Sc. and M.Sc. degrees from Wuhan University of Technology, Wuhan, China in 2011 and 2013, respectively. Her research interests include intelligent unmanned vehicles, model predictive control and distributed control.

Rudy Negenborn is full professor "Multi-Machine Operations & Logistics" at TU Delft within the Department of Maritime and Transport Technology, 3mE. His more fundamental research interests are in the areas of distributed control, multi-agent systems, model predictive control, and optimization. He applies the developed theories to address control problems in large-scale transportation and logistics systems, as well as for realizing real-time coordination among autonomous vessels. In addition, he edited the book Intelligent Infrastructures (2010), Distributed Model Predictive Control Made Easy (2014), and Transport Of Water versus Transport Over Water (2015), and obtained an NWO/STW VENI grant and various other national and international research grants. He currently has over 200 peer reviewed publications in international journals and conference proceedings.

Xiumin Chu is a professor in the National Engineering Research Center for Water Transport Safety, Wuhan University of Technology, Wuhan, China. He received the PhD degree (2002) and M.S. degree (1998) majoring in Automobile Application Engineering in Jilin University. His research topics include waterway transportation intelligence, smart ship, and ship motion simulation.

Shuo Xie is a PhD candidate of National Engineering Research Center for Water Transport Safety, Wuhan University of Technology. He received his master degree in Wuhan University of Technology. His research interests include ship model identification, ship control and collision avoidance.

How to cite this article: Liu C, Zheng H, Negenborn R, Chu X, Xie S. Adaptive predictive path following control based on least squares support vector machines for underactuated autonomous vessels. *Asian J Control*. 2021;23:432–448. https://doi.org/10.1002/asjc.2208