Lu et al. *Advances in Difference Equations* (2019) 2019:195
https://doi.org/10.1186/s13662-019-2131-3

Advances in Difference Equations
a SpringerOpen Journal

RESEARCH **Open Access**

The LS-SVM algorithms for boundary value problems of high-order ordinary differential equations

Yanfei Lu[1], Qingfei Yin[1], Hongyi Li[1], Hongli Sun[1], Yunlei Yang[1] and Muzhou Hou[1*]

*Correspondence:
houmuzhou@sina.com
[1]School of Mathematics and Statistics, Central South University, Changsha, China

Abstract

This paper introduces the improved LS-SVM algorithms for solving two-point and multi-point boundary value problems of high-order linear and nonlinear ordinary differential equations. To demonstrate the reliability and powerfulness of the improved LS-SVM algorithms, some numerical experiments for third-order, fourth-order linear and nonlinear ordinary differential equations with two-point and multi-point boundary conditions are performed. The idea can be extended to other complicated ordinary differential equations.

Keywords: Numerical solutions; High-order ordinary differential equations; Two-point boundary value problems; Multi-point boundary value problems; LS-SVM algorithms

1 Introduction

High-order boundary value problems for ordinary differential equations are used to model different problems in some fields such as biology, economics, and engineering. Due to the importance of high-order ordinary differential equations, a considerable size of research work has been carried out about this problem. Among others, finite difference method [1] was proposed to solve two-point boundary value problems for high-order linear and nonlinear ordinary differential equations. Homotopy perturbation method [2, 3] was used for the solution of fourth-order and sixth-order boundary value problems. Ali [4] proposed the optimal homotopy asymptotic method to solve multi-point boundary value problems. Adomian decomposition method [5–10] was presented for solving two-point and multi-point boundary value problems of high-order ordinary differential equations. Haar wavelets method [11] and Shannon wavelet method [12] were proposed to solve boundary value problems of high-order ordinary differential equations. Doha [13] proposed spectral Galerkin algorithms based on Jacobi polynomials for solving two-point boundary value problems of third-order and fifth-order ordinary differential equations. Doha [14] proposed spectral Galerkin algorithms by using Chebyshev polynomials of the third and fourth kinds for solving high even-order differential equations. Shifted Jacobi collocation method [15] was proposed for solving nonlinear high-order multi-point boundary value problems. Saadatmandi and Dehghan [16] discussed sinc-collocation method for solving multi-point boundary value problems. Variational iteration method [17–19] was applied

Springer

to solving two-point boundary value problems of high-order linear and nonlinear ordinary differential equations. Although these numerical methods provide good approximations to the solution, the approximate solution derivatives are discontinuous and can seriously affect the stability of the solution.

Neural network, which is one of machine intelligence techniques, has universal function approximation capabilities [20–22], and the solution obtained from the neural network is differentiable and in closed analytic form. Neural network has been widely used for solving ordinary differential equations [23, 24], partial differential equations [25–27], fractional differential equations [28–30], and integro-differential equations [31, 32]. Chakraverty and Mall [33] analyzed a regression-based neural network algorithm to solve two-point boundary value problems of fourth-order linear ordinary differential equations. Malek [34] proposed a novel hybrid method based on optimization techniques and feed forward artificial neural networks methods for two-point boundary value problems of fourth-order ordinary differential equations. Mai-Duy [35] discussed radial basis function networks for boundary value problems of high-order ordinary differential equations directly. However, artificial neural network has several drawbacks, such as the need for a large number of controlling parameters and the difficult choice of the number of hidden units. Furthermore, its training procedure is time-consuming and can be trapped in local minima.

SVM algorithms [36] were introduced by Vapnik in the framework of statistical learning theory. SVM algorithms map the input data into a high-dimensional feature space using a feature map. SVM algorithms can achieve a global optimum by solving a convex quadratic programming problem. Meanwhile, SVM algorithms adopt the structural risk minimization principle, which has a better generalization performance. LS-SVM algorithms [37] are a modification of SVM algorithms. LS-SVM algorithms change inequality constraints to equality constraints and regard the sum of squared errors loss function as experience loss of the training set. LS-SVM algorithms will deal with a set of linear equations instead of a quadratic optimization problem, which reduces the computation time of model learning significantly and improves higher solution accuracy. Therefore, LS-SVM algorithms have various applications in the area of pattern recognition [38], fault diagnosis [39], and time-series prediction [40, 41]. In addition, LS-SVM algorithms have been successfully applied for solving differential equations [42, 43], differential algebraic equations [44, 45], and integral equations [46].

LS-SVM algorithms are only used to solve two-point boundary value problems of second-order linear ordinary differential equations [42]. To the best of our knowledge, there are not too many results on LS-SVM algorithms for solving two-point and multi-point boundary value problems of high-order linear and nonlinear ordinary differential equations. The main goal of the present thesis is to develop improved LS-SVM algorithms to solve two-point and multi-point boundary value problems of high-order linear and nonlinear ordinary differential equations.

The remainder of this paper is organized as follows. First, Sect. 2 introduces least squares support vector machines. A brief overview of LS-SVM algorithms for solving ordinary differential equations is provided, and some definitions are given in Sect. 3. Following, in Sect. 4, the proposed LS-SVM algorithms for solving two-point boundary value problems of high-order linear and nonlinear ordinary differential equations and multi-point boundary value problems of high-order linear and nonlinear ordinary differential equations are discussed. In Sect. 5, we present five numerical examples to exhibit the accuracy and the efficiency of our proposed LS-SVM algorithms. Finally, concluding remarks are presented in Sect. 6.

2 Least squares support vector machines

Consider a given training data set $\{(x_i, y_i)|x_i \in R^n, y_i \in R\}_{i=1}^N$ (in this paper $n = 1$), where $\{x_i\}_{i=1}^N$ are input data points and $\{y_i\}_{i=1}^N$ are the corresponding output data points. One assumes that the underlying function describing the relation between input points and output points has the following form:

$$y(x) = \boldsymbol{\omega}^T \boldsymbol{\phi}(x) + b, \tag{1}$$

where $\boldsymbol{\omega}$ and b are parameters of the model that have to be determined and $\boldsymbol{\phi}(x)$ is the nonlinear feature map which maps an input space into a higher dimensional feature space. Then, the optimal solution is sought in that space by minimizing the residual between the model outputs and the measurements [47]. To this end, the LS-SVM model in the primal is formulated as the following optimization problem [37, 48]:

$$\min_{\boldsymbol{\omega}, b, e_i} J(\boldsymbol{\omega}, \boldsymbol{e}) = \frac{1}{2}\boldsymbol{\omega}^T \boldsymbol{\omega} + \frac{1}{2}\gamma \boldsymbol{e}^T \boldsymbol{e} \tag{2}$$

subject to

$$y_i = \boldsymbol{\omega}^T \boldsymbol{\phi}(x_i) + b + e_i, \quad i = 1, 2, \dots, N,$$

where γ is a positive regularization parameter and e_i is the error of the ith input data. The first term is a regularization term, while the second one minimizes the training errors.

The optimization problem with equality constraints (2) can be solved by using the Lagrange multipliers method

$$L(\boldsymbol{\omega}, b, \alpha_i, e_i) = \frac{1}{2}\boldsymbol{\omega}^T \boldsymbol{\omega} + \frac{1}{2}\gamma \boldsymbol{e}^T \boldsymbol{e} - \sum_{i=1}^N \alpha_i \big[\boldsymbol{\omega}^T \boldsymbol{\phi}(x_i) + b + e_i - y_i\big], \tag{3}$$

where α_i ($i = 1, 2, \dots, N$) are Lagrange multipliers that can be positive or negative in the LS-SVM formulation.

According to the KKT conditions, we will obtain

$$\begin{cases} \frac{\partial L}{\partial \boldsymbol{\omega}} = \boldsymbol{\omega} - \sum_{i=1}^N \alpha_i \boldsymbol{\phi}(x_i) = 0; \\ \frac{\partial L}{\partial b} = \sum_{i=1}^N \alpha_i = 0; \\ \frac{\partial L}{\partial e_i} = \alpha_i - \gamma e_i = 0; \\ \frac{\partial L}{\partial \alpha_i} = \boldsymbol{\omega}^T \boldsymbol{\phi}(x_i) + b - y_i + e_i = 0. \end{cases} \tag{4}$$

When $\boldsymbol{\omega}$ and e_i are eliminated from a system of Eq. (4), we obtain the following linear system:

$$\left[\begin{array}{c|c} \Theta_{ij} + \gamma^{-1} E & I_{N-1}^T \\ \hline I_{N-1} & 0 \end{array}\right] \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix}, \tag{5}$$

where $\Theta_{ij} = K(x_i, x_j) = \boldsymbol{\phi}(x_i)^T\boldsymbol{\phi}(x_j)$ $(i, j = 1, 2, \ldots, N)$ is the ijth entry of the positive definite kernel matrix; $\boldsymbol{y} = [y_1, y_1, \ldots, y_N]^T$; $\boldsymbol{\alpha} = [\alpha_1, \alpha_1, \ldots, \alpha_N]^T$ and $I_{N-1} = [1, 1, \ldots, 1]$.

Finally, the LS-SVM model in the dual form can be described as

$$y(x) = \sum_{i=1}^{N} \alpha_i K(x_i, x) + b. \tag{6}$$

3 Brief overview of LS-SVM model for solving ODEs and some definitions

In this section, a brief overview of LS-SVM algorithms for solving ordinary differential equations is provided and some definitions are given.

With regard to the initial value problem of the first-order linear ordinary differential equation in the following form [42]:

$$\begin{cases} \frac{dy}{dx} = a(x)y(x) + r(x), \quad x \in [a, c], \\ y(a) = A, \end{cases} \tag{7}$$

the authors in [42] assume that a general approximate solution is $y = \boldsymbol{\omega}^T\boldsymbol{\phi}(x) + b$, where $\boldsymbol{\omega}$ and b are the parameters to be solved. Then the interval $[a, c]$ is discretized into a series of collocation points by using collocation methods [49], and the optimal values of the parameters $\boldsymbol{\omega}$ and b are obtained by solving the optimization problem with constraints, see [42]. According to the Lagrange multipliers method [50], the optimization problem with constraints is transformed into the Lagrangian function which is composed of the LS-SVM cost function and constraints that the approximate solution $y = \boldsymbol{\omega}^T\boldsymbol{\phi}(x) + b$ satisfies the given first-order linear ordinary differential equation and the initial condition at the collocation points. The described methodology is applicable for solving other types of differential equations including second-order boundary value problems, partial differential equations, and descriptor systems [42–44].

The feature map $\boldsymbol{\phi}$ is not explicitly known in general, so the kernel function will be introduced. By utilizing Mercer's theorem [36], the derivative of the kernel function is defined as [42, 44]

$$\nabla_{n,m}\bigl(K(x_i, x_j)\bigr) = \frac{\partial^{n+m}(K(u, v))}{\partial u^n \partial v^m}\bigg|_{u=x_i, v=x_j} = \boldsymbol{\phi}^{(n)}(x_i)^T\boldsymbol{\phi}^{(m)}(x_j) = [\Theta_{n,m}]_{i,j}. \tag{8}$$

In this paper, the RBF kernel $K(u, v) = \exp(-(u-v)^2/\sigma^2)$ is considered as a kernel function, then we can obtain

$$\begin{aligned} \nabla_{1,3}\bigl(K(x_i, x_j)\bigr) &= \frac{\partial^4(K(u, v))}{\partial u \partial v^3}\bigg|_{u=x_i, v=x_j} = \boldsymbol{\phi}^{(1)}(x_i)^T\boldsymbol{\phi}^{(3)}(x_j) = [\Theta_{1,3}]_{i,j} \\ &= -\left[\frac{12}{\sigma^4} - \frac{12}{\sigma^2}\left[\frac{2(x_i - x_j)}{\sigma^2}\right]^2 + \left[\frac{2(x_i - x_j)}{\sigma^2}\right]^4\right]K(x_i, x_j); \\ \nabla_{2,3}\bigl(K(x_i, x_j)\bigr) &= \frac{\partial^5(K(u, v))}{\partial u^2 \partial v^3}\bigg|_{u=x_i, v=x_j} = \boldsymbol{\phi}^{(2)}(x_i)^T\boldsymbol{\phi}^{(3)}(x_j) = [\Theta_{2,3}]_{i,j} \\ &= \left[\frac{60}{\sigma^4} - \frac{20}{\sigma^2}\left[\frac{2(x_i - x_j)}{\sigma^2}\right]^2 + \left[\frac{2(x_i - x_j)}{\sigma^2}\right]^4\right]\frac{2(x_i - x_j)}{\sigma^2}K(x_i, x_j); \end{aligned}$$

$$\begin{aligned}
\nabla_{3,3}\bigl(K(x_i,x_j)\bigr) &= \frac{\partial^6(K(u,v))}{\partial u^3 \partial v^3}\bigg|_{u=x_i,v=x_j} = \boldsymbol{\phi}^{(3)}(x_i)^T\boldsymbol{\phi}^{(3)}(x_j) = [\Theta_{3,3}]_{i,j} \\
&= \left[\frac{120}{\sigma^6} - \frac{180}{\sigma^4}\left[\frac{2(x_i-x_j)}{\sigma^2}\right]^2 + \frac{30}{\sigma^2}\left[\frac{2(x_i-x_j)}{\sigma^2}\right]^4 - \left[\frac{2(x_i-x_j)}{\sigma^2}\right]^6\right] \\
&\quad \times K(x_i,x_j); \\
\nabla_{3,4}\bigl(K(x_i,x_j)\bigr) &= \frac{\partial^7(K(u,v))}{\partial u^3 \partial v^4}\bigg|_{u=x_i,v=x_j} = \boldsymbol{\phi}^{(3)}(x_i)^T\boldsymbol{\phi}^{(4)}(x_j) = [\Theta_{3,4}]_{i,j} \\
&= \left[\frac{840}{\sigma^6} - \frac{420}{\sigma^4}\left[\frac{2(x_i-x_j)}{\sigma^2}\right]^2 + \frac{42}{\sigma^2}\left[\frac{2(x_i-x_j)}{\sigma^2}\right]^4 - \left[\frac{2(x_i-x_j)}{\sigma^2}\right]^6\right] \\
&\quad \times \frac{2(x_i-x_j)}{\sigma^2}K(x_i,x_j); \\
\nabla_{4,4}\bigl(K(x_i,x_j)\bigr) &= \frac{\partial^8(K(u,v))}{\partial u^4 \partial v^4}\bigg|_{u=x_i,v=x_j} = \boldsymbol{\phi}^{(4)}(x_i)^T\boldsymbol{\phi}^{(4)}(x_j) = [\Theta_{4,4}]_{i,j} \\
&= \left[\frac{1680}{\sigma^8} - \frac{3360}{\sigma^6}\left[\frac{2(x_i-x_j)}{\sigma^2}\right]^2 + \frac{840}{\sigma^4}\left[\frac{2(x_i-x_j)}{\sigma^2}\right]^4 - \frac{56}{\sigma^2}\left[\frac{2(x_i-x_j)}{\sigma^2}\right]^6 \right. \\
&\quad \left. + \left[\frac{2(x_i-x_j)}{\sigma^2}\right]^8\right]K(x_i,x_j).
\end{aligned}$$

4 Boundary value problems of high-order ordinary differential equations

In this section, we formulate the improved LS-SVM algorithms to the solution of two-point and multi-point boundary value problems of high-order linear and nonlinear ordinary differential equations.

4.1 Two-point boundary value problems of high-order ordinary differential equations

The improved LS-SVM algorithms to the solution of two-point boundary value problems of high-order linear and nonlinear ordinary differential equations are described.

4.1.1 Nonlinear ordinary differential equations for two-point boundary value problems

Two-point boundary value problems of Mth-order nonlinear ordinary differential equations to be solved can be stated as follows:

$$\frac{d^M y}{dx^M} + a_{M-1}(x)\frac{d^{M-1}y}{dx^{M-1}} + \cdots + a_1(x)\frac{dy}{dx} = f(x,y), \quad x \in [a,c], \tag{9}$$

subject to boundary conditions $y^{(s)}(a) = p_s$, $y^{(r)}(c) = q_r$, $0 \le s \le S$, $0 \le r \le R$, $R = M-2-S$.

The interval $[a,c]$ is discretized into a series of collocation points $\Omega = \{a = x_1 < x_2 < \cdots < x_N = c\}$. Assume that a general approximate solution to (9) is $y = \boldsymbol{\omega}^T\boldsymbol{\phi}(x) + b$. The optimal values of the parameters $\boldsymbol{\omega}$ and b are obtained by the following optimization problem:

$$\min_{\boldsymbol{\omega},b,\boldsymbol{e},\boldsymbol{\xi},y_i} J(\boldsymbol{\omega},\boldsymbol{e},\boldsymbol{\xi}) = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + \frac{1}{2}\gamma\boldsymbol{e}^T\boldsymbol{e} + \frac{1}{2}\gamma\boldsymbol{\xi}^T\boldsymbol{\xi} \tag{10}$$

subject to

$$
\begin{aligned}
&\boldsymbol{\omega}^T\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=1}^{M-1}\boldsymbol{\omega}^T a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i) = f(x_i, y_i) + e_i, \quad i = 2,\dots,N-1;\\
&y_i = \boldsymbol{\omega}^T\boldsymbol{\phi}(x_i) + b + \xi_i, \quad i = 2,\dots,N-1;\\
&\boldsymbol{\omega}^T\boldsymbol{\phi}(x_1) + b = p_0;\\
&\boldsymbol{\omega}^T\boldsymbol{\phi}(x_N) + b = q_0;\\
&\boldsymbol{\omega}^T\boldsymbol{\phi}^{(s)}(x_1) = p_s, \quad s = 1,2,\dots,S;\\
&\boldsymbol{\omega}^T\boldsymbol{\phi}^{(r)}(x_N) = q_t, \quad r = 1,2,\dots,R.
\end{aligned}
$$

Theorem 1 *Given a positive definite kernel function $K : R \times R \to R$ and a regularization parameter $\gamma \in R^+$, the solution to (10) is given by the following dual problem*:

$$
\left[\begin{array}{c|c|c|c|c|c}
[\widehat{\varTheta}_{l,l'}]_{N-2} & [\widehat{\varTheta}_{0,l'}]_{N-2}^T & [\widehat{\varTheta}^1_{s,l'}]_{N-2}^T & [\widehat{\varTheta}^N_{r,l'}]_{N-2}^T & 0_{1,N-2}^T & 0_{N-2}\\ \hline
[\widehat{\varTheta}_{0,l'}]_{N-2} & [\widetilde{\varTheta}_{0,0}]_{N-2} & [\overline{\varTheta}^1_{s,0}]_{N-2}^T & [\overline{\varTheta}^N_{r,0}]_{N-2}^T & I_{1,N-2}^T & -E_{N-2}\\ \hline
[\widehat{\varTheta}^1_{s,l'}]_{N-2} & [\overline{\varTheta}^1_{s,0}]_{N-2} & [\widetilde{\varTheta}_{s,s'}]_{1,1} & [\widetilde{\varTheta}_{r,s'}]_{N,1}^T & \mathbf{1}_s & 0_{s,N-2}\\ \hline
[\widehat{\varTheta}^N_{r,l'}]_{N-2} & [\overline{\varTheta}^N_{r,0}]_{N-2} & [\widetilde{\varTheta}_{r,s'}]_{N,1} & [\widetilde{\varTheta}_{r,r'}]_{N,N} & \mathbf{1}_r & 0_{r,N-2}\\ \hline
0_{1,N-2} & I_{1,N-2} & \mathbf{1}_s^T & \mathbf{1}_r^T & 0 & 0_{1,N-2}\\ \hline
D_{N-2}(\boldsymbol{y}) & E_{N-2} & 0_{s,N-2}^T & 0_{r,N-2}^T & 0_{1,N-2}^T & 0_{N-2}
\end{array}\right]
\times
\left[\begin{array}{c}
\boldsymbol{\alpha}\\ \hline \boldsymbol{\eta}\\ \hline \boldsymbol{\beta}\\ \hline \boldsymbol{\lambda}\\ \hline b\\ \hline \boldsymbol{y}
\end{array}\right]
=
\left[\begin{array}{c}
f_{N-2}(\boldsymbol{x},\boldsymbol{y})\\ \hline 0_{1,N-2}^T\\ \hline \boldsymbol{p}\\ \hline \boldsymbol{q}\\ \hline 0\\ \hline 0_{1,N-2}^T
\end{array}\right]
\tag{11}
$$

where $[\widehat{\varTheta}_{l,l'}]_{N-2} = [\widetilde{\varTheta}_{M,M}]_{N-2} + \overline{D}_{a_l}[\overline{\varTheta}_{l,M}]_{N-2} + [\overline{\varTheta}_{M,l'}]_{N-2}\overline{D}_{a_{l'}}^T + \overline{D}_{a_l}[\overline{\varTheta}_{l,l'}]_{N-2}\overline{D}_{a_{l'}}^T + \gamma^{-1}E$; $[\overline{\varTheta}_{M,l'}]_{N-2} = [[\widetilde{\varTheta}_{M,1}]_{N-2}, [\widetilde{\varTheta}_{M,2}]_{N-2}, \dots, [\widetilde{\varTheta}_{M,M-1}]_{N-2}]$; $[\overline{\varTheta}_{l,M}]_{N-2} = [[\widetilde{\varTheta}_{1,M}]_{N-2}; [\widetilde{\varTheta}_{2,M}]_{N-2}; \dots; [\widetilde{\varTheta}_{M-1,M}]_{N-2}]$; $\overline{D}_{a_l} = [D_{a_1}, D_{a_2}, \dots, D_{a_{M-1}}]$; $[\overline{\varTheta}_{l,l'}]_{N-2} = [\widetilde{\varTheta}_{1:M-1,1:M-1}]_{N-2}$;$\overline{D}_{a_{l'}} = [D_{a_1}, D_{a_2}, \dots, D_{a_{M-1}}]$; $[\widetilde{\varTheta}_{0,0}]_{N-2} = [\varTheta_{0,0}]_{2:N-1,2:N-1} + \gamma^{-1}E$; $l, l' = 1,2,\dots,M-1$; $\boldsymbol{\alpha} = [\alpha_2, \alpha_3, \dots, \alpha_{N-1}]^T$; $D_{a_{l'}} = \operatorname{diag}(a_{l'}(t_2), a_{l'}(t_3), \dots, a_{l'}(t_N))$; $D_{a_l} = \operatorname{diag}(a_l(t_2), a_l(t_3), \dots, a_l(t_N))$; $[\widehat{\varTheta}_{0,l'}]_{N-2} = [\widetilde{\varTheta}_{0,M}]_{N-2} + [\overline{\varTheta}_{0,l'}]_{N-2}\overline{D}_{a_{l'}}^T$; $\boldsymbol{I}_{1,N-2} = [1,1,\dots,1]$; $\boldsymbol{\beta} = [\beta_0, \beta_1, \dots, \beta_S]^T$; $[\overline{\varTheta}_{0,l'}]_{N-2} = [[\widetilde{\varTheta}_{0,1}]_{N-2}, [\widetilde{\varTheta}_{0,2}]_{N-2}, \dots, [\widetilde{\varTheta}_{0,M-1}]_{N-2}]$; $\boldsymbol{\eta} = [\eta_2, \eta_3, \dots, \eta_{N-1}]^T$; $[\widehat{\varTheta}^1_{s,l'}]_{N-2} = [\widetilde{\varTheta}^1_{0:S,M}]_{N-2} + [\overline{\varTheta}^1_{0:S,l'}]_{N-2}\overline{D}_{a_{l'}}^T$; $[\widehat{\varTheta}^N_{r,l'}]_{N-2} = [\widetilde{\varTheta}^N_{0:R,M}]_{N-2} + [\overline{\varTheta}^N_{0:R,l'}]_{N-2}\overline{D}_{a_{l'}}^T$; $[\overline{\varTheta}^1_{0:S,l'}]_{N-2} = [[\widetilde{\varTheta}^1_{0:S,1}]_{N-2}, [\widetilde{\varTheta}^1_{0:S,2}]_{N-2}, \dots, [\widetilde{\varTheta}^1_{0:S,M-1}]_{N-2}]$; $\boldsymbol{q} = [q_0, q_1, q_2, \dots, q_R]^T$; $[\overline{\varTheta}^N_{0:R,l'}]_{N-2} = [[\widetilde{\varTheta}^N_{0:R,1}]_{N-2}, [\widetilde{\varTheta}^N_{0:R,2}]_{N-2}, \dots, [\widetilde{\varTheta}^N_{0:R,M-1}]_{N-2}]$; $\boldsymbol{\lambda} = [\lambda_0, \lambda_1, \lambda_2, \dots, \lambda_R]^T$; $[\overline{\varTheta}^1_{s,0}]_{N-2} = [\widetilde{\varTheta}^1_{0:S,0}]_{N-2}$; $[\overline{\varTheta}^N_{r,0}]_{N-2} = [\widetilde{\varTheta}^N_{0:R,0}]_{N-2}$; $\boldsymbol{y} = [y_2, y_3, \dots, y_{N-1}]$; $0_{s,N-2} = 0_{S+1,N-2}$; $[\widetilde{\varTheta}_{s,s'}]_{1,1} = [\varTheta_{0:S,0:S}]_{1,1}$; $[\widetilde{\varTheta}_{r,s'}]_{N,1} = [\varTheta_{0:R,0:S}]_{N,1}$; $[\widetilde{\varTheta}_{r,r'}]_{N,N} = [\varTheta_{0:R,0:R}]_{N,N}$; $f_{N-2}(\boldsymbol{x},\boldsymbol{y}) = [f(x_2,y_2), f(x_3,y_3), \dots, f(x_{N-1},y_{N-1})]^T$; $\boldsymbol{p} = [p_0, p_1, p_2, \dots, p_S]^T$; $\frac{\partial f(\boldsymbol{x},\boldsymbol{y})}{\partial y} = [\frac{\partial f(x,y)}{\partial y}|_{x=x_2,y=y_2}, \frac{\partial f(x,y)}{\partial y}|_{x=x_3,y=y_3}, \dots, \frac{\partial f(x,y))}{\partial y}|_{x=x_{N-1},y=y_{N-1}}]$; $D_{N-2}(\boldsymbol{y}) = \operatorname{diag}(\frac{\partial f(\boldsymbol{x},\boldsymbol{y})}{\partial y})$; $\mathbf{1}_r = [1;0;\dots;0]_{R+1,1}$; $\mathbf{1}_s = [1;0;\dots;0]_{S+1,1}$; $0_{r,N-2} = 0_{R+1,N-2}$; *where* $[\widetilde{\varTheta}_{m,n}]_{N-2} =$

$[\Theta_{m,n}]_{2:N-1,2:N-1}$, $[\widetilde{\Theta}^1_{0:S,n}]_{N-2} = [[\Theta_{0:S,n}]_{1,2}, [\Theta_{0:S,n}]_{1,3}, \ldots, [\Theta_{0:S,n}]_{1,N-1}]$ *and* $[\widetilde{\Theta}^N_{0:R,n}]_{N-2} = [[\Theta_{0:R,n}]_{N,2}, [\Theta_{0:R,n}]_{N,3}, \ldots, [\Theta_{0:R,n}]_{N,N-1}]$, $m, n = 0, 1, \ldots, M$.

Proof Consider the Lagrangian function of the optimization problem (10):

$$\begin{aligned}
&L(\boldsymbol{\omega}, y_i, \alpha_i, \eta_i, \beta_0, \beta_s, \lambda_0, \lambda_r, b, e_i, \xi_i) \\
&\quad = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + \frac{1}{2}\gamma \boldsymbol{e}^T\boldsymbol{e} + \frac{1}{2}\gamma\boldsymbol{\xi}^T\boldsymbol{\xi} \\
&\qquad - \sum_{i=2}^{N-1}\alpha_i\left[\boldsymbol{\omega}^T\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=1}^{M-1}\boldsymbol{\omega}^T a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i) - f(x_i, y_i) - e_i\right] \\
&\qquad - \sum_{i=2}^{N-1}\eta_i\bigl(\boldsymbol{\omega}^T\boldsymbol{\phi}(x_i) + b + \xi_i - y_i\bigr) - \beta_0\bigl(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}(x_1) + b - p_0\bigr) - \sum_{s=1}^{S}\beta_s\bigl(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}^{(s)}(x_1) - p_s\bigr) \\
&\qquad - \lambda_0\bigl(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}(x_N) + b - q_0\bigr) - \sum_{r=1}^{M-2-S}\lambda_r\bigl(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}^{(r)}(x_N) - q_r\bigr).
\end{aligned} \tag{12}$$

Then the KKT optimality conditions are given by

$$\begin{aligned}
&\frac{\partial L}{\partial \boldsymbol{\omega}} = \boldsymbol{\omega} - \sum_{i=2}^{N-1}\alpha_i\left[\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=1}^{M-1}a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i)\right] - \sum_{i=2}^{N-1}\eta_i\boldsymbol{\phi}(x_i) - \beta_0\boldsymbol{\phi}(x_1) \\
&\qquad - \sum_{s=1}^{S}\beta_s\boldsymbol{\phi}^{(s)}(x_1) - \lambda_0\boldsymbol{\phi}(x_N) - \sum_{r=1}^{M-2-S}\lambda_r\boldsymbol{\phi}^{(r)}(x_N) = 0; \\
&\frac{\partial L}{\partial \alpha_i} = \boldsymbol{\omega}^T\left[\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=1}^{M-1}a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i)\right] - f(x_i, y_i) - e_i = 0, \quad i = 2, 3, \ldots, N-1; \\
&\frac{\partial L}{\partial \eta_i} = \boldsymbol{\omega}^T\boldsymbol{\phi}(x_i) + b + \xi_i - y_i = 0, \quad i = 2, 3, \ldots, N-1; \\
&\frac{\partial L}{\partial \beta_0} = \boldsymbol{\omega}^T\boldsymbol{\phi}(x_1) + b - p_0 = 0; \\
&\frac{\partial L}{\partial \beta_s} = \boldsymbol{\omega}^T\boldsymbol{\phi}^{(s)}(x_1) - p_s = 0, \quad s = 1, 2, \ldots, S; \\
&\frac{\partial L}{\partial \lambda_0} = \boldsymbol{\omega}^T\boldsymbol{\phi}(x_N) + b - q_0 = 0; \\
&\frac{\partial L}{\partial \lambda_r} = \boldsymbol{\omega}^T\boldsymbol{\phi}^{(r)}(x_N) - q_r = 0, \quad r = 1, 2, \ldots, M-2-S; \\
&\frac{\partial L}{\partial b} = -\sum_{i=2}^{N-1}\eta_i - \beta_0 - \lambda_0 = 0; \\
&\frac{\partial L}{\partial y_i} = \eta_i + \alpha_i\frac{\partial f(x_i, y_i)}{\partial y_i} = 0, \quad i = 2, 3, \ldots, N-1; \\
&\frac{\partial L}{\partial e_i} = \alpha_i + \gamma e_i = 0, \quad i = 2, 3, \ldots, N-1; \\
&\frac{\partial L}{\partial \xi_i} = -\eta_i + \gamma\xi_i = 0, \quad i = 2, 3, \ldots, N-1.
\end{aligned} \tag{13}$$

Finally, rewriting the above system in matrix form will result in (11). □

System (11) is solved by Newton's method. Therefore, the LS-SVM model in the dual form becomes

$$\begin{aligned}\hat{y}(x) &= b + \sum_{i=2}^{N-1}\alpha_i\left[\nabla_{M,0}\big(K(x_i,x)\big) + \sum_{l=1}^{M-1}a_l(x_i)\nabla_{l,0}\big(K(x_i,x)\big)\right] \\ &\quad + \sum_{i=2}^{N-1}\eta_i\nabla_{0,0}\big(K(x_i,x)\big) + \beta_0\nabla_{0,0}\big(K(x_1,x)\big) + \sum_{s=1}^{S}\beta_s\nabla_{s,0}\big(K(x_1,x)\big) \\ &\quad + \lambda_0\nabla_{0,0}\big(K(x_N,x)\big) + \sum_{r=1}^{M-2-S}\lambda_r\nabla_{r,0}\big(K(x_N,x)\big). \end{aligned} \tag{14}$$

4.1.2 Linear ordinary differential equations for two-point boundary value problems

Two-point boundary value problems of Mth-order linear ordinary differential equations to be solved can be stated as follows:

$$\frac{d^My}{dx^M} + a_{M-1}(x)\frac{d^{M-1}y}{dx^{M-1}} + \cdots + a_1(x)\frac{dy}{dx} + a_0(x)y = r(x), \quad x \in [a,c], \tag{15}$$

subject to boundary conditions $y^{(s)}(a) = p_s$, $y^{(r)}(c) = q_r$, $0 \le s \le S$, $0 \le r \le R$, $R = M-2-S$.

Assume that a general approximate solution to (15) is $y = \boldsymbol{\omega}^T\boldsymbol{\phi}(x) + b$. To obtain the optimal values of the parameters $\boldsymbol{\omega}$ and b, collocation methods which discretize the interval $[a,c]$ into a series of collocation points $\Omega = \{a = x_1 < x_2 < \cdots < x_N = c\}$ can be used. Therefore, these parameters are obtained by solving the following optimization problem:

$$\min_{\omega,b,e_i} J(\boldsymbol{\omega},\boldsymbol{e}) = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + \frac{1}{2}\gamma\boldsymbol{e}^T\boldsymbol{e} \tag{16}$$

subject to

$$\begin{aligned} &\boldsymbol{\omega}^T\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=0}^{M-1}\boldsymbol{\omega}^T a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i) + a_0(x_i)b = r(x_i) + e_i, \quad i = 2,\dots,N-1; \\ &\boldsymbol{\omega}^T\boldsymbol{\phi}(x_1) + b = p_0; \\ &\boldsymbol{\omega}^T\boldsymbol{\phi}(x_N) + b = q_0; \\ &\boldsymbol{\omega}^T\boldsymbol{\phi}^{(s)}(x_1) = p_s, \quad s = 1,2,\dots,S; \\ &\boldsymbol{\omega}^T\boldsymbol{\phi}^{(r)}(x_N) = q_t, \quad r = 1,2,\dots,R. \end{aligned}$$

Theorem 2 *Given a positive definite kernel function $K: R \times R \to R$ and a regularization parameter $\gamma \in R^+$, the solution to (16) is obtained by the following dual problem*:

$$\left[\begin{array}{c|c|c|c} [\widehat{\Theta}_{l,l'}]_{N-2} & [\widehat{\Theta}^1_{s,l'}]^T_{N-2} & [\widehat{\Theta}^N_{r,l'}]^T_{N-2} & A^T \\ \hline [\widehat{\Theta}^1_{s,l'}]_{N-2} & [\widetilde{\Theta}_{s,s'}]_{1,1} & [\widetilde{\Theta}_{r,s'}]^T_{N,1} & \mathbf{1}_s \\ \hline [\widehat{\Theta}^N_{r,l'}]_{N-2} & [\widetilde{\Theta}_{r,s'}]_{N,1} & [\widetilde{\Theta}_{r,r'}]_{N,N} & \mathbf{1}_r \\ \hline A & \mathbf{1}_s^T & \mathbf{1}_r^T & 0 \end{array}\right] \left[\begin{array}{c} \boldsymbol{\alpha} \\ \hline \boldsymbol{\beta} \\ \hline \boldsymbol{\lambda} \\ \hline b \end{array}\right] = \left[\begin{array}{c} r(\boldsymbol{x}) \\ \hline \boldsymbol{p} \\ \hline \boldsymbol{q} \\ \hline 0 \end{array}\right] \tag{17}$$

where $[\widehat{\Theta}_{l,l'}]_{N-2} = [\widetilde{\Theta}_{M,M}]_{N-2} + \overline{D}_{a_l}[\overline{\Theta}_{l,M}]_{N-2} + [\overline{\Theta}_{M,l'}]_{N-2}\overline{D}^T_{a_{l'}} + \overline{D}_{a_l}[\overline{\Theta}_{l,l'}]_{N-2}\overline{D}^T_{a_{l'}} + \gamma^{-1}E$; $[\overline{\Theta}_{M,l'}]_{N-2} = [[\widetilde{\Theta}_{M,0}]_{N-2}, [\widetilde{\Theta}_{M,1}]_{N-2}, \dots, [\widetilde{\Theta}_{M,M-1}]_{N-2}]$; $\overline{D}_{a_{l'}} = [D_{a_0}, D_{a_1}, \dots, D_{a_{M-1}}]$;

$[\overline{\Theta}_{l,M}]_{N-2} = [[\widetilde{\Theta}_{0,M}]_{N-2}; [\widetilde{\Theta}_{1,M}]_{N-2}; \ldots; [\widetilde{\Theta}_{M-1,M}]_{N-2}]$; $\overline{D}_{a_l} = [D_{a_0}, D_{a_1}, \ldots, D_{a_{M-1}}]$; $[\overline{\Theta}_{l,l'}]_{N-2} = [\widetilde{\Theta}_{0:M-1,0:M-1}]_{N-2}$; $D_{a_{l'}} = \operatorname{diag}(a_{l'}(t_2), a_{l'}(t_3), \ldots, a_{l'}(t_{N-1}))$; $l, l' = 0, 1, \ldots, M-1$; $D_{a_l} = \operatorname{diag}(a_l(t_2), a_l(t_3), \ldots, a_l(t_{N-1}))$; $\boldsymbol{p} = [p_0, p_1, p_2, \ldots, p_S]^T$; $\boldsymbol{q} = [q_0, q_1, q_2, \ldots, q_R]^T$; $[\widehat{\Theta}^1_{s,l'}]_{N-2} = [\widetilde{\Theta}^1_{0:S,M}]_{N-2} + [\overline{\Theta}^1_{0:S,l'}]_{N-2}\overline{D}^T_{a_{l'}}$; $[\widehat{\Theta}^N_{r,l'}]_{N-2} = [\widetilde{\Theta}^N_{0:R,M}]_{N-2} + [\overline{\Theta}^N_{0:R,l'}]_{N-2}\overline{D}^T_{a_{l'}}$; $[\overline{\Theta}^1_{0:S,l'}]_{N-2} = [[\widetilde{\Theta}^1_{0:S,0}]_{N-2}, [\widetilde{\Theta}^1_{0:S,1}]_{N-2}, \ldots, [\widetilde{\Theta}^1_{0:S,M-1}]_{N-2}]$; $\mathbf{1}_s = [1; 0; \ldots; 0]_{S+1,1}$; $[\overline{\Theta}^N_{0:R,l'}]_{N-2} = [[\widetilde{\Theta}^N_{0:R,0}]_{N-2}, [\widetilde{\Theta}^N_{0:R,1}]_{N-2}, \ldots, [\widetilde{\Theta}^N_{0:R,M-1}]_{N-2}]$; $\boldsymbol{\alpha} = [\alpha_2, \alpha_3, \ldots, \alpha_{N-1}]^T$; $[\widetilde{\Theta}_{s,s'}]_{1,1} = [\Theta_{0:S,0:S}]_{1,1}$; $[\widetilde{\Theta}_{r,s'}]_{N,1} = [\Theta_{0:R,0:S}]_{N,1}$; $[\widetilde{\Theta}_{r,r'}]_{N,N} = [\Theta_{0:R,0:R}]_{N,N}$; $\boldsymbol{\beta} = [\beta_0, \beta_1, \beta_2, \ldots, \beta_S]^T$; $\boldsymbol{\lambda} = [\lambda_0, \lambda_1, \lambda_2, \ldots, \lambda_R]^T$; $\mathbf{1}_r = [1; 0; \ldots; 0]_{R+1,1}$; $A = [a_0(x_2), a_0(x_3), \ldots, a_0(x_{N-1})]$; $r(\boldsymbol{x}) = [r(x_2), r(x_3), \ldots, r(x_{N-1})]^T$.

Proof We construct the Lagrangian function of the optimization problem (16):

$$\begin{aligned} &L(\boldsymbol{\omega}, \alpha_i, \beta_0, \beta_s, \lambda_0, \lambda_r, b, e_i) \\ &\quad = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + \frac{1}{2}\gamma \boldsymbol{e}^T\boldsymbol{e} \\ &\qquad - \sum_{i=2}^{N-1} \alpha_i \left[\boldsymbol{\omega}^T\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=0}^{M-1} \boldsymbol{\omega}^T a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i) + a_0(x_i)b - r(x_i) - e_i \right] \\ &\qquad - \beta_0\big(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}(x_1) + b - p_0\big) - \sum_{s=1}^{S} \beta_s\big(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}^{(s)}(x_1) - p_s\big) - \lambda_0\big(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}(x_N) + b - q_0\big) \\ &\qquad - \sum_{r=1}^{M-2-S} \lambda_r\big(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}^{(r)}(x_N) - q_r\big). \end{aligned} \tag{18}$$

The conditions for optimality are as follows:

$$\begin{aligned} &\frac{\partial L}{\partial \boldsymbol{\omega}} = \boldsymbol{\omega} - \sum_{i=2}^{N-1} \alpha_i \left[\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=0}^{M-1} a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i) \right] - \beta_0\boldsymbol{\phi}(x_1) - \sum_{s=1}^{S} \beta_s\boldsymbol{\phi}^{(s)}(x_1) \\ &\qquad - \lambda_0\boldsymbol{\phi}(x_N) - \sum_{r=1}^{M-2-S} \lambda_r\boldsymbol{\phi}^{(r)}(x_N) = 0; \\ &\frac{\partial L}{\partial \alpha_i} = \boldsymbol{\omega}^T \left[\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=0}^{M-1} a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i) \right] + a_0(x_i)b - r(x_i) - e_i = 0, \quad i = 2, 3, \ldots, N-1; \\ &\frac{\partial L}{\partial \beta_0} = \boldsymbol{\omega}^T\boldsymbol{\phi}(x_1) + b - p_0 = 0; \\ &\frac{\partial L}{\partial \beta_s} = \boldsymbol{\omega}^T\boldsymbol{\phi}^{(s)}(x_1) - p_s = 0, \quad s = 1, 2, \ldots, S; \\ &\frac{\partial L}{\partial \lambda_0} = \boldsymbol{\omega}^T\boldsymbol{\phi}(x_N) + b - q_0 = 0; \\ &\frac{\partial L}{\partial \lambda_r} = \boldsymbol{\omega}^T\boldsymbol{\phi}^{(r)}(x_N) - q_r = 0, \quad r = 1, 2, \ldots, M-2-S; \\ &\frac{\partial L}{\partial b} = -\sum_{i=2}^{N-1} a_0(x_i)\alpha_i - \beta_0 - \lambda_0 = 0; \\ &\frac{\partial L}{\partial e_i} = \alpha_i + \gamma e_i = 0, \quad i = 2, 3, \ldots, N-1. \end{aligned} \tag{19}$$

Finally, rewriting the above system in matrix form will result in (17). □

The linear system (17), which consists of unknowns $(\alpha, \beta, \lambda, b)$, is solved. The LS-SVM model in the dual form becomes

$$\begin{aligned}\hat{y}(x) = {} & \sum_{i=2}^{N-1} \alpha_i \left[\nabla_{M,0}\big(K(x_i, x)\big) + \sum_{l=0}^{M-1} a_l(x_i) \nabla_{l,0}\big(K(x_i, x)\big) \right] \\ & + \beta_0 \nabla_{0,0}\big(K(x_1, x)\big) + \sum_{s=1}^{S} \beta_s \nabla_{s,0}\big(K(x_1, x)\big) \\ & + \lambda_0 \nabla_{0,0}\big(K(x_N, x)\big) + \sum_{r=1}^{R} \lambda_r \nabla_{r,0}\big(K(x_N, x)\big) + b. \end{aligned} \tag{20}$$

4.2 Multi-point boundary value problems of high-order ordinary differential equations

The improved LS-SVM algorithms to the solution of multi-point boundary value problems of high-order linear and nonlinear ordinary differential equations are described.

4.2.1 Nonlinear ordinary differential equations for multi-point boundary value problems

Consider the following Mth-order nonlinear ordinary differential equations for multi-point boundary value problems [15]:

$$\frac{d^M y}{dx^M} + a_{M-1}(x) \frac{d^{M-1} y}{dx^{M-1}} + \cdots + a_1(x) \frac{dy}{dx} = f(x, y), \quad x \in [a, c], \tag{21}$$

subject to $y^{(q_0)}(a) = s_0$, $y^{(q_j)}(x_{p_j}) = s_j$, $y^{(q_{M-1})}(c) = s_{M-1}$, $x_{p_j} \in [a, c]$, $p_j \in Z$, $j = 1, 2, \ldots, M-2$, $0 \le q_0, q_1, \ldots, q_{M-1} \le M-1$.

The interval $[a, c]$ is discretized into a series of collocation points $\Omega = \{a = x_{p_0} = x_1 < x_2 < \cdots < x_{p_1} < \cdots < x_{p_2} < \cdots < x_{p_{M-2}} < \cdots < x_{p_{M-1}} = x_N = c\}$. Assume that the approximate solution to (21) is $y = \boldsymbol{\omega}^T \boldsymbol{\phi}(x) + b$, the primal optimization problem is described as follows:

$$\min_{\boldsymbol{\omega}, b, e_i, \boldsymbol{\xi}, y_i} J(\boldsymbol{\omega}, \boldsymbol{e}, \boldsymbol{\xi}) = \frac{1}{2} \boldsymbol{\omega}^T \boldsymbol{\omega} + \frac{1}{2} \gamma \boldsymbol{e}^T \boldsymbol{e} + \frac{1}{2} \gamma \boldsymbol{\xi}^T \boldsymbol{\xi} \tag{22}$$

subject to

$$\begin{aligned} & \boldsymbol{\omega}^T \left[\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=1}^{M-1} a_l(x_i) \boldsymbol{\phi}^{(l)}(x_i) \right] = f(x_i, y_i) + e_i, \quad i = 1, 2, \ldots, N-M; \\ & y_i = \boldsymbol{\omega}^T \boldsymbol{\phi}(x_i) + b + \xi_i, \quad i = 1, 2, \ldots, N-M; \\ & \boldsymbol{\omega}^T \boldsymbol{\phi}^{(q_0)}(x_1) + b^{(q_0)} = s_0; \\ & \boldsymbol{\omega}^T \boldsymbol{\phi}^{(q_j)}(x_{p_j}) + b^{(q_j)} = s_j, \quad j = 1, 2, \ldots, M-2; \\ & \boldsymbol{\omega}^T \boldsymbol{\phi}^{(q_{M-1})}(x_N) + b^{(q_{M-1})} = s_{M-1}. \end{aligned}$$

Theorem 3 *Given a positive definite kernel function $K : R \times R \to R$ and a regularization parameter $\gamma \in R^+$, the solution to (22) is obtained by the following dual prob-*

lem:

$$\left[\begin{array}{c|c|c|c|c} [\widehat{\Theta}_{l,l'}]_{N-M} & [\widehat{\Theta}_{0,l'}]_{N-M}^T & [\widehat{\Theta}_{q_j,l'}]_{M,N-M}^T & 0_{1,N-M}^T & 0_{N-M} \\ \hline [\widehat{\Theta}_{0,l'}]_{N-M} & [\widetilde{\Theta}_{0,0}]_{N-M} & [\widetilde{\Theta}_{q_j,0}]_{M,N-M}^T & I_{1,N-M}^T & -E_{N-M} \\ \hline [\widehat{\Theta}_{q_j,l'}]_{M,N-M} & [\widetilde{\Theta}_{q_j,0}]_{M,N-M} & [\widetilde{\Theta}_{q_j,q_{j'}}]_{p_j,p_{j'}} & B^T & 0_{M,N-M} \\ \hline 0_{1,N-M} & I_{1,N-M} & B & 0 & 0_{1,N-M} \\ \hline D_{N-M}(\boldsymbol{y}) & E_{N-M} & 0_{M,N-M}^T & 0_{1,N-M}^T & 0_{N-M} \end{array}\right]$$

$$\times \left[\begin{array}{c} \boldsymbol{\alpha} \\ \hline \boldsymbol{\eta} \\ \hline \boldsymbol{\beta} \\ \hline b \\ \hline \boldsymbol{y} \end{array}\right] = \left[\begin{array}{c} f_{N-M}(\boldsymbol{x},\boldsymbol{y}) \\ \hline 0_{1,N-M}^T \\ \hline \boldsymbol{s} \\ \hline 0 \\ \hline 0_{1,N-M}^T \end{array}\right], \tag{23}$$

where $[\widehat{\Theta}_{l,l'}]_{N-M} = [\widetilde{\Theta}_{M,M}]_{N-M} + \overline{D}_{a_l}[\overline{\Theta}_{l,M}]_{N-M} + [\overline{\Theta}_{M,l'}]_{N-M}\overline{D}_{a_{l'}}^T + \overline{D}_{a_l}[\overline{\Theta}_{l,l'}]_{N-M}\overline{D}_{a_{l'}}^T + \gamma^{-1}E$; $[\overline{\Theta}_{M,l'}]_{N-M} = [[\widetilde{\Theta}_{M,1}]_{N-M}, [\widetilde{\Theta}_{M,2}]_{N-M}, \ldots, [\widetilde{\Theta}_{M,M-1}]_{N-M}]$; $[\overline{\Theta}_{l,M}]_{N-M} = [[\widetilde{\Theta}_{1,M}]_{N-M}; [\widetilde{\Theta}_{2,M}]_{N-M}; \ldots; [\widetilde{\Theta}_{M-1,M}]_{N-M}]$; $\overline{D}_{a_{l'}} = [D_{a_1}, D_{a_2}, \ldots, D_{a_{M-1}}]$; $\overline{D}_{a_l} = [D_{a_1}, D_{a_2}, \ldots, D_{a_{M-1}}]$; $[\overline{\Theta}_{l,l'}]_{N-M} = [\widetilde{\Theta}_{1:M-1,1:M-1}]_{N-M}$; $\boldsymbol{\alpha} = [\alpha_2, \ldots, \alpha_{p_1-1}, \alpha_{p_1+1}, \ldots, \alpha_{N-1}]^T$; $[\widetilde{\Theta}_{0,0}]_{N-M} = [\Theta_{0,0}]_{1:N-M,1:N-M} + \gamma^{-1}E$; $l, l' = 1, 2, \ldots, M-1$; $D_{a_l} = \operatorname{diag}(a_l(x_2), \ldots, a_l(x_{p_1-1}), a_l(x_{p_1+1}), \ldots, a_l(x_{N-1}))$; $B = [\chi_{b_0}, \chi_{b_1}, \ldots, \chi_{b_{M-1}}]$; $D_{a_{l'}} = \operatorname{diag}(a_{l'}(x_2), \ldots, a_{l'}(x_{p_1-1}), a_{l'}(x_{p_1+1}), \ldots, a_{l'}(x_{N-1}))$; $[\widehat{\Theta}_{0,l'}]_{N-M} = [\widetilde{\Theta}_{0,M}]_{N-M} + [\overline{\Theta}_{0,l'}]_{N-M}\overline{D}_{a_{l'}}^T$; $\boldsymbol{s} = [s_0, s_1, s_2, \ldots, s_{M-1}]^T$; $[\overline{\Theta}_{0,l'}]_{N-M} = [[\widetilde{\Theta}_{0,1}]_{N-M}, [\widetilde{\Theta}_{0,2}]_{N-M}, \ldots, [\widetilde{\Theta}_{0,M-1}]_{N-M}]$; $\boldsymbol{y} = [y_2, \ldots, y_{P_1-1}, y_{P_1+1}, \ldots, y_{N-1}]^T$; $[\widehat{\Theta}_{q_j,l'}]_{M,N-M} = [\widetilde{\Theta}_{q_0:q_{M-1},M}]_{M,N-M} + [\overline{\Theta}_{q_0:q_{M-1},l'}]_{M,N-M}\overline{D}_{a_{l'}}^T$; $\boldsymbol{\beta} = [\beta_0, \beta_1, \beta_2, \ldots, \beta_{M-1}]^T$; $[\overline{\Theta}_{q_0:q_{M-1},l'}]_{M,N-M} = [[\widetilde{\Theta}_{q_0:q_{M-1},1}]_{M,N-M}, [\widetilde{\Theta}_{q_0:q_{M-1},2}]_{M,N-M}, \ldots, [\widetilde{\Theta}_{q_0:q_{M-1},M-1}]_{M,N-M}]$; $[\widetilde{\Theta}_{q_j,0}]_{M,N-M} = [\Theta_{q_0:q_{M-1},0}]_{M,N-M}$; $\boldsymbol{\eta} = [\eta_2, \ldots, \eta_{p_1-1}, \eta_{p_1+1}, \ldots, \eta_{N-1}]^T$; $[\widetilde{\Theta}_{q_j,q_{j'}}]_{p_j,p_{j'}} = [\Theta_{q_0:q_{M-1},q_0:q_{M-1}}]_{p_0:p_{M-1},p_0:p_{M-1}}$; $D_{N-M}(\boldsymbol{y}) = \operatorname{diag}(\frac{\partial f(\boldsymbol{x},\boldsymbol{y})}{\partial y})$; $f_{N-M}(\boldsymbol{x},\boldsymbol{y}) = [f(x_2,y_2), \ldots, f(x_{p_1-1},y_{p_1-1}), f(x_{p_1+1},y_{p_1+1}), \ldots, f(x_{N-1},y_{N-1})]^T$; $\frac{\partial f(\boldsymbol{x},\boldsymbol{y})}{\partial y} = [\frac{\partial f(x,y)}{\partial y}|_{\substack{x=x_2\\y=y_2}}, \ldots, \frac{\partial f(x,y)}{\partial y}|_{\substack{x=x_{p_1-1}\\y=y_{p_1-1}}}, \frac{\partial f(x,y)}{\partial y}|_{\substack{x=x_{p_1+1}\\y=y_{p_1+1}}}, \ldots, \frac{\partial f(x,y))}{\partial y}|_{\substack{x=x_{N-1}\\y=y_{N-1}}}]$, where $[\widetilde{\Theta}_{m,n}]_{N-M} = [\Theta_{m,n}]_{1:N-M,1:N-M}$; $[\widetilde{\Theta}_{q_0:q_{M-1},m}]_{M,N-M} = [\Theta_{q_0:q_{M-1},m}]_{p_0:p_{M-1},1:N-M}$; $m, n = 0, 1, \ldots, M-1$.

Proof The Lagrangian function of the constrained optimization problem (22) is introduced as follows:

$$\begin{aligned} &L(\boldsymbol{\omega}, y_i, \alpha_i, \eta_i, \beta_j, b, e_i, \xi_i) \\ &\quad= \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + \frac{1}{2}\gamma\boldsymbol{e}^T\boldsymbol{e} + \frac{1}{2}\gamma\boldsymbol{\xi}^T\boldsymbol{\xi} \\ &\qquad- \sum_{i=1}^{N-M}\alpha_i\left[\boldsymbol{\omega}^T\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=1}^{M-1}\boldsymbol{\omega}^T a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i) - f(x_i,y_i) - e_i\right] \\ &\qquad- \sum_{i=1}^{N-M}\eta_i\bigl(\boldsymbol{\omega}^T\boldsymbol{\phi}(x_i) + b + \xi_i - y_i\bigr) - \sum_{j=0}^{M-1}\beta_j\bigl(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}^{(q_j)}(x_{p_j}) + b^{(q_j)} - s_j\bigr). \end{aligned} \tag{24}$$

The conditions for optimality

$$
\begin{aligned}
&\frac{\partial L}{\partial \boldsymbol{\omega}} = \boldsymbol{\omega} - \sum_{i=1}^{N-M} \alpha_i \left[\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=1}^{M-1} a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i)) \right] - \sum_{i=1}^{N-M} \eta_i \boldsymbol{\phi}(x_i) - \sum_{j=0}^{M-1} \beta_j \boldsymbol{\phi}^{(q_j)}(x_{p_j}) = 0; \\
&\frac{\partial L}{\partial \alpha_i} = \boldsymbol{\omega}^T \left(\boldsymbol{\phi}^{(M)}(x_i) + \sum_{l=1}^{M-1} a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i) \right) - f(x_i, y_i) - e_i = 0, \quad i = 1, 2, \ldots, N-M; \\
&\frac{\partial L}{\partial \eta_i} = \boldsymbol{\omega}^T \boldsymbol{\phi}(x_i) + b + \xi_i - y_i = 0, \quad i = 1, 2, \ldots, N-M; \\
&\frac{\partial L}{\partial \xi_i} = -\eta_i + \gamma \xi_i = 0, \quad i = 1, 2, \ldots, N-M; \\
&\frac{\partial L}{\partial e_i} = \alpha_i + \gamma e_i = 0, \quad i = 1, 2, \ldots, N-M; \\
&\frac{\partial L}{\partial \beta_j} = \boldsymbol{\omega}^T \boldsymbol{\phi}^{(q_j)}(x_{p_j}) + b^{(q_j)} - s_j = 0, \quad j = 0, 1, \ldots, M-1; \\
&\frac{\partial L}{\partial b} = -\sum_{i=1}^{N-M} \eta_i - \sum_{j=0}^{M-1} \beta_j \chi_{b_j} = 0, \quad \chi_{b_j} = \begin{cases} 1, & q_j = 0; \\ 0, & q_j = 1, 2, \ldots, M-1; \end{cases} \\
&\frac{\partial L}{\partial y_i} = \eta_i + \alpha_i \frac{\partial f(x_i, y_i)}{\partial y_i} = 0
\end{aligned}
\tag{25}
$$

can be written as a system in matrix form (23), after eliminating parameters ω and e_i. □

System (23), which consists of $3N - 2M + 1$ equations with unknowns $(\alpha, \eta, \beta, b, y)$, is solved by Newton's method. The LS-SVM model in the dual form becomes

$$
\begin{aligned}
\hat{y}(x) = {} & \sum_{i=1}^{N-M} \alpha_i \left[\nabla_{M,0}\big(K(x_i, x)\big) + \sum_{l=1}^{M-1} a_l(x_i) \nabla_{l,0}\big(K(x_i, x)\big) \right] \\
& + \sum_{i=1}^{N-M} \eta_i \nabla_{0,0}\big(K(x_i, x)\big) + \sum_{j=0}^{M-1} \beta_j \nabla_{q_j,0}\big(K(x_{p_j}, x)\big) + b.
\end{aligned}
\tag{26}
$$

4.2.2 Linear ordinary differential equations for multi-point boundary value problems

Consider the following Mth-order linear ordinary differential equations for multi-point boundary value problems:

$$
\frac{d^M y}{dx^M} + a_{M-1}(x)\frac{d^{M-1} y}{dx^{M-1}} + \cdots + a_1(x)\frac{dy}{dx} + a_0(x)y = r(x), \quad x \in [a, c],
\tag{27}
$$

subject to $y^{(q_0)}(a) = s_0$, $y^{(q_j)}(x_{p_j}) = s_j$, $y^{(q_{M-1})}(c) = s_{M-1}$, $x_{p_j} \in [a, c]$, $p_j \in Z$, $j = 1, 2, \ldots, M-2$, $0 \le q_0, q_1, \ldots, q_{M-1} \le M-1$.

The interval $[a, c]$ is discretized into a series of collocation points $\Omega = \{a = x_{p_0} = x_1 < x_2 < \cdots < x_{p_1} < \cdots < x_{p_2} < \cdots < x_{p_{M-2}} < \cdots < x_{p_{M-1}} = x_N = c\}$. Suppose that the approximate solution to (27) is $y = \boldsymbol{\omega}^T \boldsymbol{\phi}(x) + b$, the original optimal problem is described as follows:

$$
\min_{\boldsymbol{\omega}, b, e_i} J(\boldsymbol{\omega}, \boldsymbol{e}) = \frac{1}{2}\boldsymbol{\omega}^T \boldsymbol{\omega} + \frac{1}{2}\gamma \boldsymbol{e}^T \boldsymbol{e}
\tag{28}
$$

subject to

$$\begin{aligned}
&\boldsymbol{\omega}^T\left[\boldsymbol{\phi}^{(M)}(x_i)+\sum_{l=0}^{M-1}a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i)+a_0(x_i)b\right]=r(x_i)+e_i,\\
&\quad i=2,3,\dots,p_1-1,p_1+1,\dots,N-1;\\
&\boldsymbol{\omega}^T\boldsymbol{\phi}^{(q_0)}(x_1)+b^{(q_0)}=s_0;\\
&\boldsymbol{\omega}^T\boldsymbol{\phi}^{(q_j)}(x_{p_j})+b^{(q_j)}=s_j,\quad j=1,2,\dots,M-2;\\
&\boldsymbol{\omega}^T\boldsymbol{\phi}^{(q_{M-1})}(x_N)+b^{(q_{M-1})}=s_{M-1}.
\end{aligned}$$

Theorem 4 *Given a positive definite kernel function $K:R\times R\to R$ and a regularization parameter $\gamma\in R^+$, the solution to* (28) *is obtained by the following dual problem*:

$$\left[\begin{array}{c|c|c}
[\widehat{\Theta}_{l,l'}]_{N-M} & [\widehat{\Theta}_{q_j,l'}]^T_{M,N-M} & A^T\\
\hline
[\widehat{\Theta}_{q_j,l'}]_{M,N-M} & [\widetilde{\Theta}_{q_j,q_{j'}}]_{p_j,p_{j'}} & B^T\\
\hline
A & B & 0
\end{array}\right]
\left[\begin{array}{c}\boldsymbol{\alpha}\\ \hline \boldsymbol{\beta}\\ \hline b\end{array}\right]
=\left[\begin{array}{c}r(\boldsymbol{x})\\ \hline \boldsymbol{s}\\ \hline 0\end{array}\right], \tag{29}$$

where $[\widehat{\Theta}_{l,l'}]_{N-M}=[\widetilde{\Theta}_{M,M}]_{N-M}+\overline{D}_{a_l}[\overline{\Theta}_{l,M}]_{N-M}+[\overline{\Theta}_{M,l'}]_{N-M}\overline{D}^T_{a_{l'}}+\overline{D}_{a_l}[\overline{\Theta}_{l,l'}]_{N-M}\overline{D}^T_{a_{l'}}+\gamma^{-1}E$; $[\overline{\Theta}_{M,l'}]_{N-M}=[[\widetilde{\Theta}_{M,0}]_{N-M},[\widetilde{\Theta}_{M,1}]_{N-M},\dots,[\widetilde{\Theta}_{M,M-1}]_{N-M}]$; $\overline{D}_{a_{l'}}=[D_{a_0},D_{a_1},\dots,D_{a_{M-1}}]$; $\overline{D}_{a_l}=[D_{a_0},D_{a_1},\dots,D_{a_{M-1}}]$; $[\overline{\Theta}_{l,l'}]_{N-M}=[\widetilde{\Theta}_{0:M-1,0:M-1}]_{N-M}$; $[\overline{\Theta}_{l,M}]_{N-M}=[[\widetilde{\Theta}_{0,M}]_{N-M};[\widetilde{\Theta}_{1,M}]_{N-M};\dots;[\widetilde{\Theta}_{M-1,M}]_{N-M}]$; $D_{a_{l'}}=\operatorname{diag}(a_{l'}(x_2),\dots,a_{l'}(x_{p_1-1}),a_{l'}(x_{p_1+1}),\dots,a_{l'}(x_{N-1}))$; $\boldsymbol{s}=[s_0,s_1,s_2,\dots,s_{M-1}]^T$; $D_{a_l}=\operatorname{diag}(a_l(x_2),\dots,a_l(x_{p_1-1}),a_l(x_{p_1+1}),\dots,a_l(x_{N-1}))$; $l,l'=0,1,\dots,M-1$; $[\widehat{\Theta}_{q_j,l'}]_{M,N-M}=[\widetilde{\Theta}_{q_j,M}]_{M,N-M}+[\overline{\Theta}_{q_j,l'}]_{M,N-M}\overline{D}^T_{a_{l'}}$; $\boldsymbol{\beta}=[\beta_0,\beta_1,\beta_2,\dots,\beta_{M-1}]^T$; $[\overline{\Theta}_{q_j,l'}]_{M,N-M}=[[\widetilde{\Theta}_{q_j,0}]_{M,N-M},[\widetilde{\Theta}_{q_j,1}]_{M,N-M},\dots,[\widetilde{\Theta}_{q_j,M-1}]_{M,N-M}]$; $[\widetilde{\Theta}_{q_j,l'}]_{M,N-M}=[\Theta_{q_j,l'}]_{p_0:p_{M-1},1:N-M}$; $[\widetilde{\Theta}_{q_j,q_{j'}}]_{p_j,p_{j'}}=[\Theta_{q_0:q_{M-1},q_0:q_{M-1}}]_{p_0:p_{M-1},p_0:p_{M-1}}$; $A=[a_l(x_2),\dots,a_l(x_{p_1-1}),a_l(x_{p_1+1}),\dots,a_l(x_{N-1})]$; $\boldsymbol{\alpha}=[\alpha_2,\dots,\alpha_{x_{p_1-1}},\alpha_{x_{p_1+1}},\dots,\alpha_{N-1}]^T$; $B=[\chi_{b_0},\chi_{b_1},\dots,\chi_{b_{M-1}}]$; $r(\boldsymbol{x})=[r(x_2),\dots,r(x_{p_1-1}),r(x_{p_1+1}),\dots,r(x_{N-1})]^T$.

Proof The Lagrangian function of the optimization problem (28) becomes

$$\begin{aligned}
&L(\boldsymbol{\omega},\alpha_i,\beta_j,b,e_i)\\
&\quad=\frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega}+\frac{1}{2}\gamma\boldsymbol{e}^T\boldsymbol{e}\\
&\qquad-\sum_{i=1}^{N-M}\alpha_i\left[\boldsymbol{\omega}^T\boldsymbol{\phi}^{(M)}(x_i)+\sum_{l=0}^{M-1}\boldsymbol{\omega}^Ta_l(x_i)\boldsymbol{\phi}^{(l)}(x_i)+a_0(x_i)b-r(x_i)-e_i\right]\\
&\qquad-\sum_{j=0}^{M-1}\beta_j\big(\boldsymbol{\omega}^{\boldsymbol{T}}\boldsymbol{\phi}^{(q_j)}(x_{p_j})+b^{(q_j)}-s_j\big).
\end{aligned} \tag{30}$$

Setting the partial derivatives of the Lagrangian function to zero, we will obtain

$$\begin{aligned}
&\frac{\partial L}{\partial\boldsymbol{\omega}}=\boldsymbol{\omega}-\sum_{i=1}^{N-M}\alpha_i\left(\boldsymbol{\phi}^{(M)}(x_i)+\sum_{l=0}^{M-1}a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i)\right)-\sum_{j=0}^{M-1}\beta_j\boldsymbol{\phi}^{(q_j)}(x_{p_j})=0;\\
&\frac{\partial L}{\partial\alpha_i}=\boldsymbol{\omega}^T\left(\boldsymbol{\phi}^{(M)}(x_i)+\sum_{l=0}^{M-1}a_l(x_i)\boldsymbol{\phi}^{(l)}(x_i)\right)+a_0(x_i)b-r(x_i)-e_i=0,\quad i=1,2,\dots,N-M;
\end{aligned}$$

$$\begin{aligned}
&\frac{\partial L}{\partial e_i} = \alpha_i + \gamma e_i = 0, \quad i = 1,2,\dots,N-M; \\
&\frac{\partial L}{\partial \beta_j} = \boldsymbol{\omega}^T \boldsymbol{\phi}^{(q_j)}(x_{p_j}) + b^{(q_j)} - s_j = 0, \quad j = 0,1,\dots,M-1; \\
&\frac{\partial L}{\partial b} = -\sum_{i=1}^{N-M} a_0(x_i)\alpha_i - \sum_{j=0}^{M-1} \beta_j \chi_{b_j} = 0, \quad \chi_{b_j} = \begin{cases} 1, & q_j = 0; \\ 0, & q_j = 1,2,\dots,M-1. \end{cases}
\end{aligned} \tag{31}$$

Finally, rewriting system (31) in matrix form will result in (29). □

System (29) with unknowns (α, β, b) is solved. The LS-SVM model in the dual form becomes

$$\begin{aligned}
\hat{y}(x) = &\sum_{i=1}^{N-M} \alpha_i \left[\nabla_{M,0}\big(K(x_i,x)\big) + \sum_{l=0}^{M-1} a_l(x_i)\nabla_{l,0}\big(K(x_i,x)\big) \right] \\
&+ \sum_{j=0}^{M-1} \beta_j \nabla_{q_j,0}\big(K(x_{p_j},x)\big) + b.
\end{aligned} \tag{32}$$

5 Numerical experiments

In this section, some numerical experiments are performed in order to demonstrate the reliability and powerfulness of the improved LS-SVM algorithms. The algorithms are applied to third-order, fourth-order linear and nonlinear ordinary differential equations with two-point boundary conditions and to third-order, fourth-order linear and nonlinear ordinary differential equations with multi-point boundary conditions.

In our experiments, the performance of the proposed LS-SVM algorithms is directly related to the choice of the regularization parameter γ and the kernel parameter σ. The larger the regularization parameter γ is, the smaller the error e_i is, but when γ is a quite large value, the system of equations will be ill-conditioning. Therefore, the chosen value for γ was 10^{10}. The validation set is obtained to be the set of midpoints $Z = \{z_i | z_i = (x_i + x_{i+1})/2, i = 1,\dots,N-1\}$, where $\{x_i\}_{i=1}^N$ are training points [42]. The optimal parameter σ that results in minimum root mean squared error (RMSE) on the validation set is selected and used for evaluating the LS-SVM model on the test set. The RMSE is defined as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\big[y(z_i) - \hat{y}(z_i)\big]^2}. \tag{33}$$

5.1 Example 1

Consider the fourth-order nonlinear ordinary differential equation [51]:

$$\frac{d^4 y}{dx^4} = -\frac{x^2}{1+y^2} - 72\big(1 - 5x + 5x^2\big) + \frac{x^2}{1+(x-x^2)^6}, \quad x \in [0,1], \tag{34}$$

subject to two-point boundary conditions $y(0) = 0, y'(0) = 0, y(1) = 0, y'(1) = 0$. The analytic solution is $y = x^3(1-x)^3$.

We train the proposed LS-SVM algorithm for 11 equidistant points in the given interval $[0,1]$. The exact solution and the approximate solution via our proposed LS-SVM algorithm are shown in Fig. 1(a). Furthermore, the error between the analytic solution and

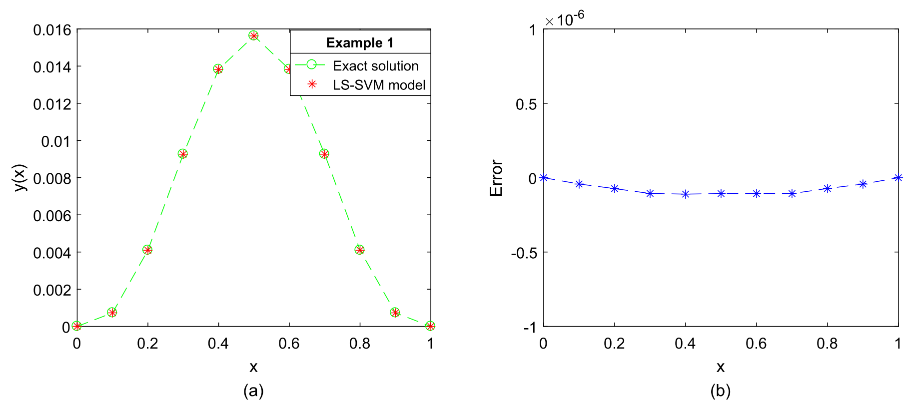


Figure 1 Two-point BVP of fourth-order nonlinear ODE (Example 1)

Table 1 Comparison between the exact solution and the LS-SVM solution (Example 1)

x	Exact solution	LS-SVM solution	Absolute error
0.0000	0.000000000000	0.000000000217	2.1737e–10
0.0915	0.000574431275	0.000574468625	3.7350e–08
0.1518	0.002134568612	0.002134627307	5.8695e–08
0.2410	0.006120352774	0.006120441263	8.8489e–08
0.3604	0.012248409908	0.012248522843	1.1293e–07
0.5000	0.015625000000	0.015625106969	1.0697e–07
0.6395	0.012252859500	0.012252970047	1.1055e–07
0.7590	0.006120352774	0.006120442154	8.9380e–08
0.8482	0.002134568612	0.002134628711	6.0099e–08
0.9084	0.000576126428	0.000576167078	4.0650e–08
1.0000	0.000000000000	0.000000000129	1.2915e–10

the approximate solution is plotted in Fig. 1(b). In spite of using fewer points, we can see that the proposed LS-SVM algorithm could have a much better performance in terms of accuracy. The mean squared error is 6.5732×10^{-15} and the maximum absolute error is approximately 1.1063×10^{-7}.

Table 1 lists the results of the exact solution and the approximate solution via our proposed LS-SVM algorithm for 11 testing points at unequal intervals in the domain $[0, 1]$. The absolute errors are shown in Table 1, in which we can see that the maximum absolute error is approximately 1.1293×10^{-7}.

Figure 2 shows the logarithmic relation between the kernel bandwidth and the RMSE in Example 1. The red circle indicates the location of selected kernel bandwidth.

5.2 Example 2

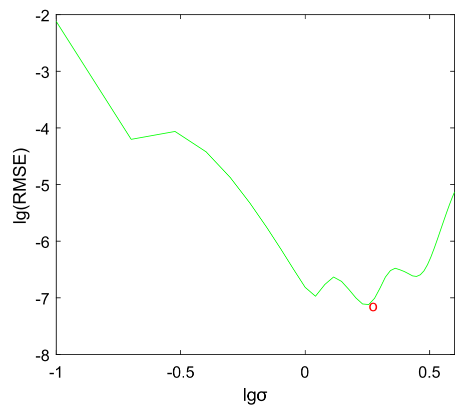
Let us consider the fourth-order linear ordinary differential equation [34]:

$$\frac{d^4y}{dx^4} = 120x, \quad x \in [-1, 1], \tag{35}$$

subject to two-point boundary conditions $y(-1) = 1$, $y'(-1) = 5$, $y(1) = 3$, $y'(1) = 5$. The analytic solution is $y = x^5 + 2$.

The proposed LS-SVM model has been trained with 11 equidistant points in the given interval $[-1, 1]$. Figure 3(a) shows comparison between the exact solution and the approx-

Figure 2 The logarithmic relation between σ and RMSE (Example 1)



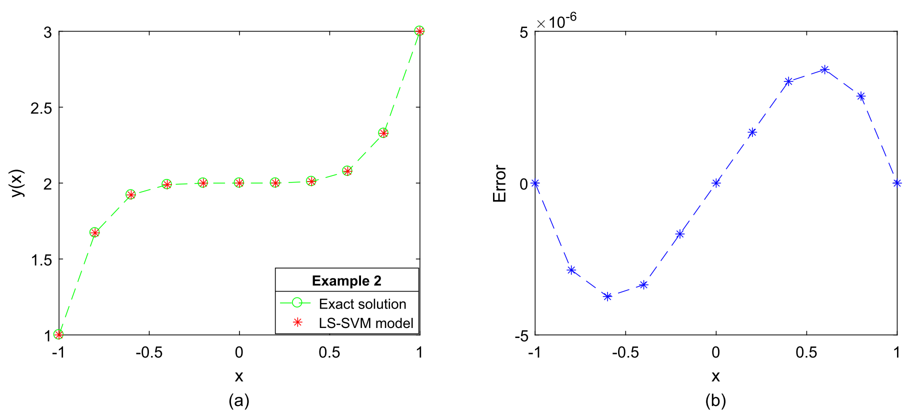


Figure 3 Two-point BVP of fourth-order linear ODE (Example 2)

imate solution via our proposed LS-SVM algorithm, and Fig. 3(b) depicts the error plot between the analytic solution and the approximate solution. From the obtained results, we can see that the mean squared error is 6.5835×10^{-12} and the maximum absolute error is approximately 3.7390×10^{-6}. The error obtained by the proposed LS-SVM algorithm remains low for the training points.

Finally, the test results of the exact solution and the approximate solution via our proposed LS-SVM algorithm for 11 equidistant points in the domain $[-1, 1]$ are listed in Table 2. The absolute errors are shown in Table 2, in which we can see that the maximum absolute error is approximately 3.7071×10^{-6}. It is clear that the proposed LS-SVM algorithm has a better performance in terms of accuracy.

5.3 Example 3

Consider the fourth-order linear ordinary differential equation [52]:

$$\frac{d^4y}{dx^4} + y(x) = \left(\left(\frac{\pi}{2}\right)^4 + 1\right)\cos\left(\frac{\pi}{2}x\right), \quad x \in [-1, 1], \tag{36}$$

subject to two-point boundary conditions $y(-1) = 0$, $y'(-1) = \pi/2$, $y(1) = 0$, $y'(1) = -\pi/2$. The analytic solution is $y = \cos(\pi x/2)$.

Table 2 Comparison between the exact solution and the LS-SVM solution (Example 2)

x	Exact solution	LS-SVM solution	Absolute error
−1.000	1.000000000000	0.999999999445	5.5479e−10
−0.815	1.640426196740	1.640428927360	2.7306e−06
−0.630	1.900756345700	1.900760018292	3.6726e−06
−0.445	1.982549814222	1.982553400148	3.5859e−06
−0.260	1.998811862400	1.998814089420	2.2270e−06
−0.075	1.999997626953	1.999998222454	5.9550e−07
0.1100	2.000016105100	2.000015225700	8.7940e−07
0.2950	2.002234138434	2.002231597236	2.5412e−06
0.4800	2.025480396800	2.025476689675	3.7071e−06
0.6650	2.130049362166	2.130045793200	3.5690e−06
0.8500	2.443705312500	2.443702978729	2.3338e−06

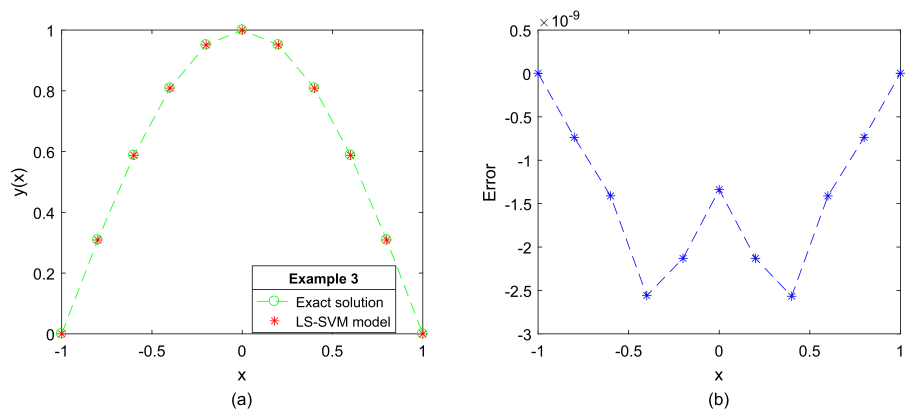


Figure 4 Two-point BVP of fourth-order linear ODE (Example 3)

The proposed LS-SVM algorithm for two-point boundary value problems of high-order linear ordinary differential equation has been trained with 11 equidistant points in the given interval $[-1, 1]$. Comparison between the exact solution and the approximate solution via our proposed LS-SVM algorithm is depicted in Fig. 4(a). Plot of the error function is cited in Fig. 4(b), from which we can see that the mean squared error is 2.6426×10^{-18} and the maximum absolute error is approximately 2.5670×10^{-9}. The accuracy of the error obtained by the proposed LS-SVM algorithm is $O(10^{-9})$. The results reveal that the proposed LS-SVM algorithm has higher accuracy, although we only choose 11 equidistant points for training process.

Finally, Table 3 incorporates results of the exact solution and the approximate solution via our proposed LS-SVM algorithm for 11 testing points at unequal intervals in the domain $[-1, 1]$. The absolute errors are shown in Table 3, in which we can see that the maximum absolute error is approximately 2.5543×10^{-9}.

5.4 Example 4

Consider the third-order nonlinear ordinary differential equation [15]:

$$\frac{d^3y}{dx^3} = -y^2 - \cos(x) + \sin^2(x), \quad x \in [0, 1], \tag{37}$$

Table 3 Comparison between the exact solution and the LS-SVM solution (Example 3)

x	Exact solution	LS-SVM solution	Absolute error
−1.000	0.000000000000000	0.000000000007191	7.19075e−12
−0.815	0.286524552727799	0.286524553452789	7.24991e−10
−0.630	0.549022817998132	0.549022819226559	1.22843e−09
−0.445	0.765483213493088	0.765483215882123	2.38904e−09
−0.260	0.917754625683981	0.917754628118062	2.43408e−09
−0.075	0.993068456954926	0.993068458430696	1.47577e−09
0.1100	0.985109326154774	0.985109327779211	1.62444e−09
0.2950	0.894544639838025	0.894544642393205	2.55518e−09
0.4800	0.728968627421412	0.728968629629382	2.20797e−09
0.6650	0.502265533143373	0.502265534200269	1.05690e−09
0.8500	0.233445363855905	0.233445364490761	6.34856e−10

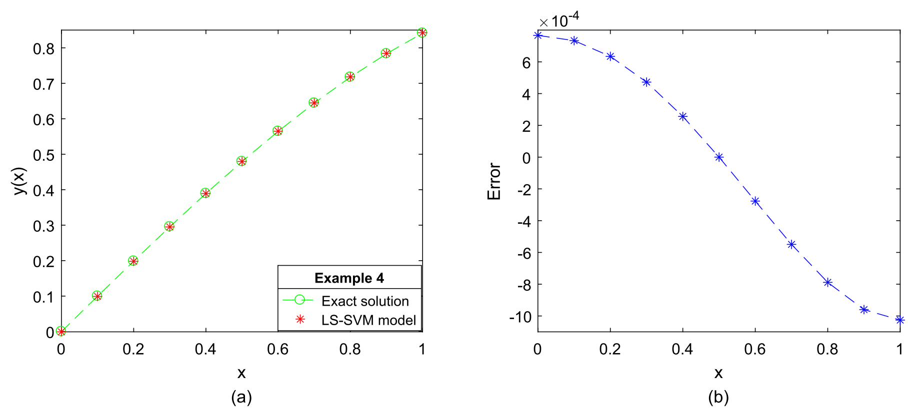


Figure 5 Multi-point BVP of third-order nonlinear ODE (Example 4)

subject to multi-point boundary conditions $y'(0) = 1$, $y(\frac{1}{2}) = \sin(\frac{1}{2})$, $y'(1) = \cos(1)$. The analytic solution is $y = \sin(x)$.

When 11 equidistant points in the interval $[0, 1]$ are used for training, the results are depicted in Fig. 5(a). Figure 5(b) shows the errors between the exact solution and the approximate solution obtained by the proposed LS-SVM algorithm. From the obtained results, although training was performed just for 11 equidistant points in the domain $[0, 1]$, the mean squared error is approximately 4.3564×10^{-7}. The proposed LS-SVM algorithm obtains a satisfactory result for multi-point boundary value problems of third-order nonlinear ordinary differential equation.

Finally, Table 4 tabulates results of the exact solution and the approximate solution via our proposed LS-SVM algorithm for 11 testing points at unequal intervals in the domain $[0, 1]$. The absolute errors are shown in Table 4, in which we can see that the mean squared error is approximately 4.9717×10^{-7}.

5.5 Example 5

Consider the fourth-order linear ordinary differential equation:

$$\frac{d^4y}{dx^4} + \frac{dy}{dx} = 4x^3 + 24, \quad x \in [0, 1], \tag{38}$$

Table 4 Comparison between the exact solution and the LS-SVM solution (Example 4)

x	Exact solution	LS-SVM solution	Absolute error
0.0000	0.000000000	0.000766592	7.6659e–04
0.0915	0.091372377	0.090633366	7.3901e–04
0.1518	0.151217677	0.150527278	6.9040e–04
0.2410	0.238673845	0.238098672	5.7517e–04
0.3604	0.352648564	0.352301106	3.4746e–04
0.5000	0.479425539	0.479425539	6.2766e–13
0.6395	0.596794319	0.597180775	3.8646e–04
0.7590	0.688196265	0.688892884	6.9662e–04
0.8482	0.750091219	0.750973103	8.8188e–04
0.9084	0.788520736	0.789490968	9.7023e–04
1.0000	0.841470985	0.842497209	0.1026e–04

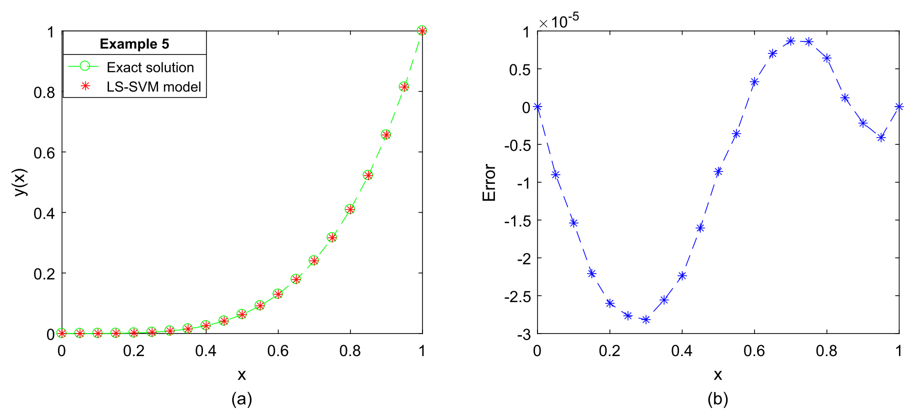


Figure 6 Multi-point BVP of third-order linear ODE (Example 5)

subject to multi-point boundary conditions $y(0) = 0, y'''(0.25) = 6, y''(0.5) = 3, y(1) = 1$. The analytic solution is $y = x^4$.

When 21 equidistant points in the interval $[0, 1]$ are used for training, the approximate solution obtained by the proposed LS-SVM algorithm is compared with the exact solution in Fig. 6(a), and the error is plotted in Fig. 6(b). From which, the mean squared error is approximately 2.2915×10^{-10}. The proposed LS-SVM algorithm can obtain the desired accuracy, although the training was performed using just a small part points in the domain $[0, 1]$.

The test results of the exact solution and the approximate solution via our proposed LS-SVM algorithm for 20 equidistant points in the domain $[0, 1]$ are listed in Table 5, and the absolute error is also calculated in Table 5. We can see that the mean squared error is approximately 2.3557×10^{-10} and the maximum absolute error is approximately 2.7702×10^{-5}. The improved LS-SVM algorithm has a good performance for solving multi-point boundary value problems of fourth-order linear ordinary differential equations.

6 Conclusion

In this paper, the improved LS-SVM algorithms have been developed for solving two-point and multi-point boundary value problems of high-order linear and nonlinear ordinary differential equations. Accuracy of the improved LS-SVM algorithms has been checked by solving a fourth-order nonlinear ordinary differential equation with two-point boundary

Table 5 Comparison between exact solution and LS-SVM solution (Example 5)

x	Exact solution	LS-SVM solution	x	Exact solution	LS-SVM solution
0.000	0.000000000000	0.000000000000	0.525	0.075969140625	0.075975418091
0.075	0.000031640625	0.000044822693	0.575	0.109312890625	0.109313011169
0.125	0.000244140625	0.000263214111	0.625	0.152587890625	0.152583122253
0.175	0.000937890625	0.000963211060	0.675	0.207594140625	0.207587242126
0.225	0.002562890625	0.002589225769	0.725	0.276281640625	0.276272773743
0.275	0.005719140625	0.005746841431	0.775	0.360750390625	0.360743522644
0.325	0.011156640625	0.011182785034	0.825	0.463250390625	0.463246345520
0.375	0.019775390625	0.019799232483	0.875	0.586181640625	0.586181640625
0.425	0.032625390625	0.032644271851	0.925	0.732094140625	0.732098579407
0.475	0.050906640625	0.050920486450	1.000	1.000000000000	1.000000000000

conditions, two fourth-order linear ordinary differential equations with two-point boundary conditions, a three-order nonlinear ordinary differential equation with multi-point boundary conditions, and a fourth-order linear ordinary differential equation with multi-point boundary conditions. The results obtained by the improved LS-SVM algorithms are compared with the exact solution. It has been noted that our proposed LS-SVM algorithms can solve two-point and multi-point boundary value problems of high-order linear and nonlinear ordinary differential equations with higher accuracy in the tables and graphs. So the improved LS-SVM algorithms in the use of the two-point and multi-point boundary value problems are found to be efficient and straightforward.

Acknowledgements
The authors sincerely thank all the reviewers and the editors for their careful reading and valuable comments, which improved the quality of this paper.

Funding
This study was funded by the National Natural Science Foundation of China under Grants 61375063.

Availability of data and materials
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
All authors contributed to the draft of the manuscript and all authors read and approved the final manuscript.

Publisher's Note

Received: 3 January 2019 Accepted: 7 May 2019 Published online: 21 May 2019

References

1. Chawla, M.M., Katti, C.P.: Finite difference methods for two-point boundary value problems involving high order differential equations. BIT Numer. Math. **19**, 27–33 (1979)
2. Mohyud-Din, S.T., Noor, M.A.: Homotopy perturbation method for solving fourth-order boundary value problems. Math. Probl. Eng. **2007**, Article ID 98602 (2007)
3. Noor, M.A., Mohyud-Din, S.T.: Homotopy perturbation method for solving sixth-order boundary value problems. Comput. Math. Appl. **55**, 2953–2972 (2008)
4. Ali, J., Islam, S., Islam, S., Zaman, G.: The solution of multipoint boundary value problems by the optimal homotopy asymptotic method. Comput. Math. Appl. **59**, 2000–2006 (2010)
5. Tatari, M., Dehghan, M.: The use of the Adomian decomposition method for solving multipoint boundary value problems. Phys. Scr. **73**, 672–676 (2006)
6. Wazwaz, A.M.: A new algorithm for calculating Adomian polynomials for nonlinear operators. Appl. Math. Comput. **111**, 53–69 (2000)
7. Wazwaz, A.M.: The numerical solution of fifth-order boundary value problems by the decomposition method. J. Comput. Appl. Math. **136**, 259–270 (2001)
8. Wazwaz, A.M.: The numerical solution of sixth-order boundary value problems by the modified decomposition method. Appl. Math. Comput. **118**, 311–325 (2001)

9. Wazwaz, A.M.: Approximate solutions to boundary value problems of higher order by the modified decomposition method. Comput. Math. Appl. **40**, 679–691 (2000)
10. Wazwaz, A.M.: The modified decomposition method for analytic treatment of differential equations. Appl. Math. Comput. **173**, 165–176 (2006)
11. Aziz, I., Siraj-ul-Islam, Nisar, M.: An efficient numerical algorithm based on Haar wavelet for solving a class of linear and nonlinear nonlocal boundary-value problems. Calcolo **53**, 621–633 (2016)
12. Shi, Z., Li, F.: Numerical solution of high-order differential equations by using periodized Shannon wavelets. Appl. Math. Model. **38**, 2235–2248 (2014)
13. Doha, E.H., Bhrawy, A.H., Hafez, R.M.: A Jacobi–Jacobi dual-Petrov–Galerkin method for third- and fifth-order differential equations. Math. Comput. Model. **53**, 1820–1832 (2011)
14. Doha, E.H., Abd-Elhameed, W.M., Bassuony, M.A.: New algorithms for solving high even-order differential equations using third and fourth Chebyshev–Galerkin methods. J. Comput. Phys. **236**, 563–579 (2013)
15. Doha, E.H., Bhrawy, A.H., Hafez, R.M.: On shifted Jacobi spectral method for high-order multi-point boundary value problems. Commun. Nonlinear Sci. Numer. Simul. **17**, 3802–3810 (2012)
16. Saadatmandi, A., Dehghan, M.: The use of sinc-collocation method for solving multi–point boundary value problems. Commun. Nonlinear Sci. Numer. Simul. **17**, 593–601 (2012)
17. Noor, M.A., Mohyud-Din, S.T.: Variational iteration technique for solving higher order boundary value problems. Appl. Math. Comput. **189**, 1929–1942 (2007)
18. Xu, L.: The variational iteration method for fourth order boundary value problems. Chaos Solitons Fractals **39**, 1386–1394 (2009)
19. Noor, M.A., Mohyud-Din, S.T.: Modified variational iteration method for solving fourth-order boundary value problems. J. Appl. Math. Comput. **29**, 81–94 (2009)
20. Hou, M., Han, X.: The multidimensional function approximation based on constructive wavelet RBF neural network. Appl. Soft Comput. **11**(2), 2173–2177 (2011)
21. Hou, M., Han, X.: Multivariate numerical approximation using constructive $L^2(R)$ RBF neural network. Neural Comput. Appl. **21**(1), 25–34 (2012)
22. Hou, M., Han, X.: Constructive approximation to multivariate function by decay RBF neural network. IEEE Trans. Neural Netw. **21**(9), 1517–1523 (2010)
23. Yang, Y., Hou, M., Luo, J.: A novel improved extreme learning machine algorithm in solving ordinary differential equations by Legendre neural network methods. Adv. Differ. Equ. **2018**, 469 (2018)
24. Mall, S., Chakraverty, S.: Application of Legendre neural network for solving ordinary differential equations. Appl. Soft Comput. **43**, 347–356 (2016)
25. Rudd, K., Ferrari, S.: A constrained integration (CINT) approach to solving partial differential equations using artificial neural networks. Neurocomputing **155**, 277–285 (2015)
26. Sun, H., Hou, M., Yang, Y., Zhang, T., Weng, F., Han, F.: Solving partial differential equation based on Bernstein neural network and extreme learning machine algorithm. Neural Process. Lett. (2018). https://doi.org/10.1007/s11063-018-9911-8
27. Yang, Y., Hou, M., Sun, H., Zhang, T., Weng, F., Luo, J.: Neural network algorithm based on Legendre improved extreme learning machine for solving elliptic partial differential equations. Soft Comput. (2019). https://doi.org/10.1007/s00500-019-03944-1
28. Zuniga-Aguilar, C.J., Romero-Ugalde, H.M., Gomez-Aguilar, J.F., et al.: Solving fractional differential equations of variable-order involving operators with Mittag–Leffler kernel using artificial neural networks. Chaos Solitons Fractals **103**, 382–403 (2017)
29. Rostami, F., Jafarian, A.: A new artificial neural network structure for solving high-order linear fractional differential equations. Int. J. Comput. Math. **95**(3), 528–539 (2018)
30. Pakdaman, M., Ahmadian, A., Effati, S., et al.: Solving differential equations of fractional order using an optimization technique based on training artificial neural network. Appl. Math. Comput. **293**, 81–95 (2017)
31. Chaharborj, S.S., Chaharborj, S.S., Mahmoudi, Y.: Study of fractional order integro–differential equations by using Chebyshev neural network. J. Math. Stat. **13**(1), 1–13 (2017)
32. Zhou, T., Liu, X., Hou, M., Liu, C.: Numerical solution for ruin probability of continuous time model based on neural network algorithm. Neurocomputing **331**, 67–76 (2019)
33. Chakraverty, S., Mall, S.: Regression based weight generation algorithm in neural network for solution of initial and boundary value problems. Neural Comput. Appl. **25**, 585–594 (2014)
34. Malek, A., Beidokhti, R.S.: Numerical solution for high order differential equations using a hybrid neural network–optimization method. Appl. Math. Comput. **183**, 260–271 (2006)
35. Mai-Duy, N.: Solving high order ordinary differential equations with radial basis function networks. Int. J. Numer. Methods Eng. **62**(6), 824–852 (2005)
36. Vapnik, V.N.: The Nature of Statistical Learning Theory, 1st edn. Springer, New York (1995)
37. Suykens, J.A.K., Vandewalle, J.: Least squares support vector machine classifiers. Neural Process. Lett. **9**(3), 293–300 (1999)
38. Yang, Y., Tan, M., Dai, Y.: An improved CS-LSSVM algorithm-based fault pattern recognition of ship power equipments. PLoS ONE **12**, 1–10 (2017)
39. Liu, X., Bo, L., Luo, H.: Bearing faults diagnostics based on hybrid LS-SVM and EMD method. Measurement **59**, 145–166 (2015)
40. Yu, L., Chen, H., Wang, S., Lai, K.K.: Evolving least squares support vector machines for stock market trend mining. IEEE Trans. Evol. Comput. **13**, 87–102 (2009)
41. Junga, H.C., Kimb, J.S., Heo, H.: Prediction of building energy consumption using an improved real coded genetic algorithm based least squares support vector machine approach. Energy Build. **90**, 76–84 (2015)
42. Mehrkanoon, S., Falck, T., Suykens, J.A.K.: Approximate solutions to ordinary differential equations using least squares support vector machines. IEEE Trans. Neural Netw. Learn. Syst. **23**(9), 1356–1367 (2012)
43. Mehrkanoon, S., Suykens, J.A.K.: Learning solutions to partial differential equations using LS-SVM. Neurocomputing **159**, 105–116 (2015)

44. Mehrkanoon, S., Suykens, J.A.K.: LS-SVM approximate solution to linear time varying descriptor systems. Automatica **48**, 2502–2511 (2012)
45. Zhang, G., Wang, S., Wang, Y.: LS-SVM approximate solution for affine nonlinear systems with partially unknown functions. J. Ind. Manag. Optim. **10**, 621–636 (2014)
46. Wang, Q., Wang, K., Chen, S.: Least squares approximation method for the solution of Volterra–Fredholm integral equations. J. Comput. Appl. Math. **272**, 141–147 (2014)
47. Mehrkanoon, S., Suykens, J.A.K.: Deep hybrid neural–kernel networks using random Fourier features. Neurocomputing **298**, 46–54 (2018)
48. Suykens, J.A.K., Gestel, T.V., Brabanter, J.D., Moor, B.D., Vandewalle, J.: Least Squares Support Vector Machines. World Scientific, Singapore (2002)
49. Kincaid, D.R., Cheney, E.W.: Numerical Analysis: Mathematics of Scientific Computing, 3rd edn. Brooks/Cole, Pacific Grove (2002)
50. Arfken, G.B., Weber, H.J.: Mathematical Methods for Physicists, 4nd edn. Academic Press, New York (1995) 973.
51. Gamel, M.E., Behiry, S.H., Hashish, H.: Numerical method for solution of special nonlinear fourth-order boundary value problems. Appl. Math. Comput. **145**, 717–734 (2003)
52. Nurmuhammada, A., Muhammada, M., Moria, M., Sugiharab, M.: Double exponential transformation in the sinc-collocation method for a boundary value problem with fourth-order ordinary differential equation. J. Comput. Appl. Math. **182**, 32–50 (2005)