Soft Computing (2020) 24:4623–4643
https://doi.org/10.1007/s00500-019-04223-9

METHODOLOGIES AND APPLICATION

An adaptive sliding mode controller based on online support vector regression for nonlinear systems

Kemal Uçak[1] · Gülay Öke Günel[2]

Published online: 18 July 2019

Abstract
In this paper, a novel adaptive sliding mode controller (SMC) based on support vector regression (SVR) is introduced for nonlinear systems. The closed-loop margin notion introduced for self-tuning regulators is rearranged in order to optimize the parameters of SMC. The proposed adjustment mechanism consists of an online SVR to identify the forward dynamics of the controlled system and SMC parameter estimators realized by separate online SVRs to approximate each tunable controller parameter. The performance of the proposed control architecture has been evaluated by simulations performed on a nonlinear continuously stirred tank reactor system, and the obtained results indicate that the SMC based on SVR provides robust and stable closed-loop performance.

Keywords Sliding mode control · Stability analysis · Support vector regression · SVR-based parameter estimator · SVR-based SMC

1 Introduction

Nonlinearity and uncertainty are the main inevitable complexities in identification of system dynamics. Due to the inadequacy of linear methods in control of nonlinear systems and discrepancies between dynamics of the actual system and mathematical model of the controlled systems, it is necessary to deploy adaptive nonlinear robust controller structures to successfully identify and control nonlinear systems.

Sliding mode control (SMC), the main idea of which is based on variable structure control (VSC), is one of the most notable nonlinear deterministic control techniques owing to its implementation simplicity, high robustness to strong nonlinearities, tolerance to modelling and system parameter inaccuracies and order reduction features (Feng et al. 2014; Sabanovic 2011; Utkin 1992; Tokat et al. 2003). Basically, in SMC, the aim is to compel the system dynamics to a predefined sliding surface that subsumes the desired stable dynamics (Tokat et al. 2003). Then, using the merits of Lyapunov's stability theory, system dynamics are retained on this sliding surface and shifted to the origin of the phase plane, resulting in a simultaneous shift of the error dynamics towards the origin (Efe MÖ et al. 2001b). Thus, the design of a SMC consists of two main phases, namely a reaching phase and a sliding phase.

In reaching phase, a control law carrying the states of the system from initial conditions to the desired sliding surface is derived using the approximated dynamics of the controlled system. Therefore, the reaching time of the system states to sliding surface depends on the accuracy of the system model and slope of the sliding surface. In sliding phase, the control law (equivalent control law) holding the system states on the sliding surface (Liu and Wang 2012) and ensuring stability and convergence is attained. Since the sliding surface is constituted via Lyapunov's stability theory, robust tracking is ensured once system states arrive at the sliding surface, whereas robustness is not guaranteed during the reaching phase (Bartoszewicz 1996). SMC consists of continuous and discontinuous parts when examined in terms of reaching and sliding phases. Because of the discontinuous nature of the

Communicated by V. Loia.

✉ Kemal Uçak
ucak@mu.edu.tr

Gülay Öke Günel
gulay.oke@itu.edu.tr

[1] Department of Electrical and Electronics Engineering, Faculty of Engineering, Muğla Sıtkı Koçman University, 48000 Kötekli, Muğla, Turkey

[2] Department of Control and Automation Engineering, Faculty of Electrical-Electronics Engineering, Istanbul Technical University, 34469 Maslak, Istanbul, Turkey

switching mechanism, chattering phenomenon emerges and the unmodelled high-frequency dynamics of the controlled system may be stimulated. Hence, the control signal becomes more sensitive to measurement noise and fragility of the controller against measurement noise increases. Therefore, various solutions to restrain the chattering phenomenon in SMC have been proposed and enhanced since the precessor form of SMC called as variable structure control (VSC) was first proposed in Emelyanov's first monograph on variable structure systems (VSS) in Soviet Union/Moscow in 1967 (de la Parte et al. 2002; Emelyanov 1967; Emel'yanov 2007; Utkin 1977). Various SMC structures have been introduced such as integral SMC(ISMC) in order to eliminate the drawbacks resulting from reaching phase (Pan et al. 2018).

There are several factors affecting the performance of SMC. These factors can be mainly examined under two headings:

- Optimal selection of several parameters utilized in the design of SMC (such as slope of sliding surface, gain of switching function and parameters of saturation etc.).
- Accurate estimation of system dynamics required to design the controller (estimation of system dynamics).

Since pure SMC suffers from chattering and is vulnerable to measurement noise (Kaynak et al. 2001) and also a good mathematical model of the system is required to compute the equivalent control law, artificial intelligence (AI)-based solutions have been proposed to improve the performance of SMC and to overcome its drawbacks (Al-Duwaish and Al-Hamouz 2011; Guo et al. 2006; Baric et al. 2005; Sun et al. 2011; Ertugrul and Kaynak 2000; Fei and Ding 2012; Kim and Lee 1995; Ngo et al. 2017; Al-Holou et al. 2002; Hušek 2016; Roopaei and Jahromi 2009; Yau and Chen 2006; Hung and Chung 2007; Lin and Shen 2006; Lin et al. 2001; Li et al. 2008a, b; Li and Li 2008a, b; Tokat et al. 2009a, b; Tokat 2006).

The parameter selection of SMC includes the optimization of sliding surface parameters such as slope, gain and tunable parameter of saturation or switching function. Also, these parameters must be adaptive to deal with time-varying effects of the system, the changing reference signal, disturbances, noise, etc. Duwaish and Hamouz proposed to deploy an NN structure to identify the dynamics between operating points and SMC parameters in order to enhance the control performance under different operation conditions (Al-Duwaish and Al-Hamouz 2011). The training data pairs are gathered using genetic algorithms, and a radial basis function (RBF)-NN structure is trained in offline manner. Even if fuzzy structures have no learning ability, they can be deployed in SMC to approximate the switching control law. Kim and Lee have proposed a fuzzy controller with fuzzy sliding surface for nonlinear systems (Kim and Lee 1995). This fuzzy controller has single input and output different from conventional fuzzy controllers. The input of the fuzzy controller is the sliding function, and output is switching control signal. The fuzzy rule base of the controller is constructed via fuzzy sliding surface. Both Mamdani (Kim and Lee 1995; Ngo et al. 2017; Al-Holou et al. 2002; Hušek 2016) and Sugeno-type structures (Roopaei and Jahromi 2009; Yau and Chen 2006) have been deployed. In adaptive control, a major problem is parameter convergence and the input signal to the system must satisfy the persistent excitation (PE) condition. Pan and Yu proposed a method called composite learning (Pan and Yu 2016) where recorded and instantaneous data are used to generate prediction errors which are in turn used together with tracking errors to update estimates of parameters. This technique guarantees parameter convergence without the PE condition. Also, Pan et al. (2017) have developed a backstepping-based strategy for a class of strict-feedback nonlinear systems with functional uncertainties using composite learning concept (NNCLC).

The second crucial factor is accurate estimation of dynamics of the system to be controlled. Guo et al. (2006) have introduced an RBF-SMC for a chaotic system to approximate the control signal where the adjustment rules for weights of the NN structure are derived depending on the reaching condition in SMC. Baric et al. (2005) employed an MLP-NN to identify the uncertainties in system dynamics. Thus, by taking the uncertainties into account, the convenient SMC is designed for the controlled system. Sun et al. (2011) proposed to use a NN to identify the required system dynamics to design equivalent controller part of SMC. Fei and Ding (2012) have proposed an RBF-based adaptive SMC to combine adaptive sliding mode control and the nonlinear approximation ability of NN which is employed to adaptively identify model uncertainties and external disturbances to restrain the chattering of sliding modes. By combining the learning ability of NN and powerful sides of FL, ANFIS-based SMC structures have also been developed for nonlinear systems (Hung and Chung 2007; Lin and Shen 2006; Lin et al. 2001). Ertugrul and Kaynak (2000) deployed two NN structures to estimate both the equivalent and switching control signals.

SVR proposed by Vapnik is one of the most effective regression techniques in recent years. Owing to their superior generalization performances, SVR-based structures have frequently been proposed for SMC of nonlinear systems (Li et al. 2008a, b; Li and Li 2008a, b; Tokat et al. 2009a, b; Tokat 2006). Li et al. (2008a) have developed a chattering-free SMC based on LS-SVM for uncertain discrete systems with input saturation. LS-SVM structure is deployed in place of the sign function of reaching law in conventional SMC to obtain switching control law. Li and Li (2008a, b) and Li et al. (2008b) have developed chattering-free SMC architectures which combine linear matrix inequality (LMI) approach and

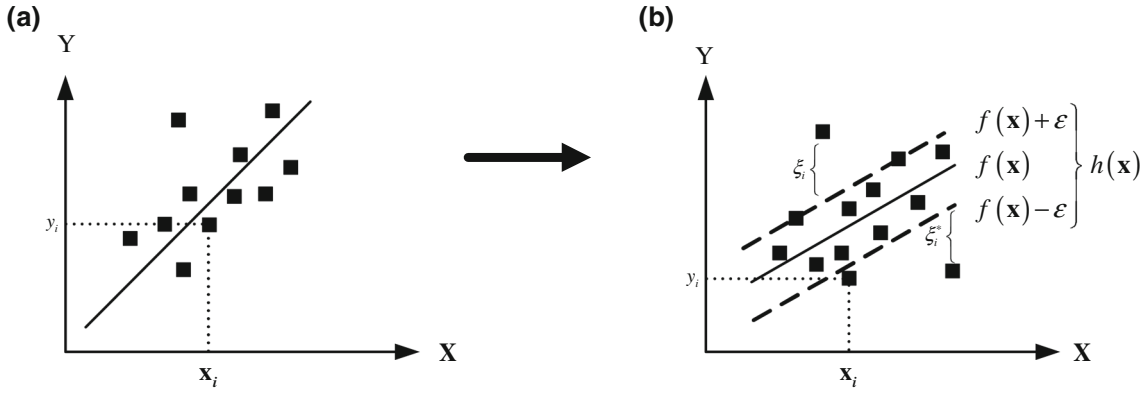

Fig. 1 Support vector regression

SVR for uncertain time delay systems. Tokat et al. (2009a) and Tokat (2006) proposed an SVR-based SMC where the parameters of the time-varying sliding surface are approximated via SVR structure. Tokat et al. (2009b) have introduced an output feedback sliding mode control based on SMC where it is assumed that the mathematical functions of the subdynamics of the system are known and SVR is deployed to identify the known subdynamics of the controlled system. Since, in practice, it is difficult to acquire the mathematical model of the subdynamics, it is identified without the mathematical function representing the behaviour between input–output of the subdynamics in Tokat et al. (2009b).

In this paper, a novel adaptive SMC based on online SVR has been introduced for nonlinear dynamical systems. The proposed method brings forward three main novelties for SMC design.

- The parameters of the sliding surface, which are required to drag the system dynamics to the sliding surface and also the parameter of the switching function so as to palliate the chattering, are approximated via separate SVR structures.
- "Online SVR" is utilized to approximate the optimal parameter values of the SMC. For this purpose, the "closed-loop margin" notion proposed in Uçak and Günel (2016) and Uçak and Günel (2017a) has been expanded to optimize SMC parameters.
- The main distinguishing feature of the proposed method with respect to previous studies combining SVR with SMC summarized above is the utilization of SVR directly to approximate the SMC parameters.

The performance of the proposed SMC has been evaluated on a nonlinear continuously stirred tank reactor (CSTR) system. The results show that the proposed novel SMC structure with online SVR model attains good modelling and closed-loop control performances.

The organization of the paper is presented as follows: Sect. 2 describes the basic principles of online SVR. Constitution of regression and optimization problem in order to deploy SVR directly as a parameter estimator and the proposed SMC architecture are detailed in Sect. 3. In Sect. 4, the simulation results to evaluate the performance of the proposed SMC adjustment mechanism are given. The study is briefly concluded in Sect. 5.

2 Online support vector regression

Support vector regression, first asserted by Cortes and Vapnik (1995), Drucker et al. (1997) and Vapnik et al. (1997), is one of the most effective data sampled regression methods among machine learning algorithms. In SVR, the aim is to obtain a regression function which optimally represents the given samples. Let us consider a training sample data set ($\mathbf{T}$) illustrated in Fig. 1a and given as follows:

$$\mathbf{T} = \{\mathbf{x_i}, y_i\}_{i=1}^{N} \quad \mathbf{x_i} \in \mathbf{X} \subseteq R^n, y_i \in R \tag{1}$$

where $\mathbf{x_i}$ are the input instances, y_i are the corresponding output samples, n is the dimension of the input sample and N is the number of the training data pairs. The linearly distributed samples can be represented using a linear SVR. However, linear SVRs remain incapable of modelling nonlinearly distributed samples. Therefore, nonlinearly distributed samples in input space ($\mathbf{I}$) are mapped via kernel functions to high-dimensional feature space ($\mathbf{F}$) where training samples can be linearly represented and can be separated by using linear SVR algorithm. The samples in (1) given in input space $\mathbf{I}$ can be modelled via SVR function in feature space $\mathbf{F}$ as follows:

$$y_i = f(\mathbf{x_i}) = \mathbf{w}^{\mathrm{T}}\mathbf{\Phi}(\mathbf{x_i}) + b, \quad i = 1, 2, \ldots, N \tag{2}$$

where $\mathbf{w}$ denotes the weights of the SVR, $\mathbf{\Phi}(\mathbf{x_i})$ is the image of input data in feature space and b is the bias of the regressor. Since SVR is constructed upon support vector classification (SVC) problem, it aims to obtain the optimal separator as in SVC. In SVC, the optimization problem is based on the max-

imization of the margin between two different classes and design of the optimal separator between these two classes. As in SVC, the objective of SVR is to obtain the optimal separator. However, SVR cannot be a margin as in SVC, from the nature of the problem, so an artificial margin is defined using a predefined ε tube. Thus, the optimization problem is transformed to obtain an optimal separator in ε tube which represents all samples with at most ε precision. The samples which deviate from the ε tube as illustrated in Fig. 1b can be represented using slack variables $(\xi_i, \xi_i^\star)$. Thus, the primal form of the optimization problem for SVR can be expressed as follows (Iplikci 2006; Smola and Schölkopf 2004):

$$\min_{\mathbf{w},b,\xi,\xi^\star} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^\star) \tag{3}$$

subject to

$$\begin{aligned} &y_i - \mathbf{w}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{x_i}) - b \le \varepsilon + \xi_i \\ &\mathbf{w}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{x_i}) + b - y_i \le \varepsilon + \xi_i^\star \\ &\xi_i, \xi_i^\star \ge 0\ ,\ i = 1, 2, \ldots N \end{aligned} \tag{4}$$

where ε stands for the maximum tolerable error and ξ's and $\xi^\star$'s are the slack variables representing the deviations from ε tube (Iplikci 2006; Smola and Schölkopf 2004). The primal form of the problem is non-convex. Therefore, in order to convert the optimization problem to a convex one, a Lagrangian function is attained using the objective function and corresponding constraints of the problem in (4). Thus, Lagrangian function can be derived as follows using Lagrange multiplier method:

$$\begin{aligned} L =& \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^\star) \\ &- \sum_{i=1}^{N}\beta_i(\varepsilon + \xi_i - y_i + \mathbf{w}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{x_i}) + b) \\ &- \sum_{i=1}^{N}\beta_i^\star(\varepsilon + \xi_i^\star + y_i - \mathbf{w}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{x_i}) - b) - \sum_{i=1}^{N}(\eta_i\xi_i + \eta_i^\star\xi_i^\star) \end{aligned} \tag{5}$$

where β, $\beta^\star$, η and $\eta^\star$ denote Lagrange multipliers (Uçak and Günel 2016; Iplikci 2006; Smola and Schölkopf 2004). According to optimization theory, the first-order optimality conditions can be derived via (5) as (Uçak and Günel 2016; Iplikci 2006; Smola and Schölkopf 2004)

$$\frac{\partial L_p}{\partial \mathbf{w}} = 0 \longrightarrow \mathbf{w} - \sum_{i=1}^{N}\beta_i\mathbf{w}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{x_i}) = 0 \tag{6}$$

$$\frac{\partial L_p}{\partial b} = 0 \longrightarrow \sum_{i=1}^{N}(\beta_i - \beta_i^\star) = 0 \tag{7}$$

$$\frac{\partial L_p}{\partial \xi_i} = 0 \longrightarrow C - \beta_i - \eta_i = 0\ ,\ \ i = 1, 2, \ldots N \tag{8}$$

$$\frac{\partial L_p}{\partial \xi_i^\star} = 0 \longrightarrow C - \beta_i^\star - \eta_i^\star = 0\ ,\ \ i = 1, 2, \ldots N \tag{9}$$

By substituting optimality conditions (6–9) in (5), dual representation of the optimization problem in (3,4) can be formulated in (10)–(11):

$$\begin{aligned} D =& \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\beta_i - \beta_i^\star)(\beta_j - \beta_j^\star)K_{ij} + \varepsilon\sum_{i=1}^{N}(\beta_i + \beta_i^\star) \\ &- \sum_{i=1}^{N} y_i\left(\beta_i - \beta_i^\star\right) \end{aligned} \tag{10}$$

subject to

$$\begin{aligned} &0 \le \beta_i \le C\ ,\ 0 \le \beta_i^\star \le C \\ &\sum_{i=1}^{N}(\beta_i - \beta_i^\star) = 0\ ,\ \ i = 1, 2, \ldots N \end{aligned} \tag{11}$$

where $K_{ij} = \boldsymbol{\Phi}(\mathbf{x_i})^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{x_j})$ (Uçak and Günel 2016; Iplikci 2006). As can be seen from (10) and (11), the dual form has a convex objective function with linear constraints, which ensures the global minimum. The quadratic programming (QP) problem in (10) and (11) can be solved using any QP solver. Thus, using obtained Lagrange values $(\beta_i, \beta_i^\star)$ in (2) and (6), the regression function can be rewritten as (Uçak and Günel 2016, 2017a)

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{N}\lambda_i K(\mathbf{x}_i, \mathbf{x}) + b\ ,\ \ \lambda_i = \beta_i - \beta_i^\star \tag{12}$$

As can be seen in Fig. 1b, the samples have different characteristics depending on their locations with respect to the ε tube. Let us define an error margin function $h(\mathbf{x_i})$ for the ith sample $\mathbf{x_i}$ as (Uçak and Günel 2016, 2017a; Ma et al. 2003; Wang et al. 2009):

$$h(\mathbf{x_i}) = f(\mathbf{x_i}) - y_i = \sum_{j=1}^{N}\lambda_j K_{ij} + b - y_i \tag{13}$$

Thus, the samples in Fig. 1b can be classified into three subsets, namely error support vectors (**E**), margin support vectors (**S**) and remaining samples (**R**), according to their Lagrange multipliers and error margin function values (Uçak and Günel 2016, 2017a; Ma et al. 2003; Wang et al. 2009) as

$$\begin{aligned} \mathbf{E} &= \{i \mid |\lambda_i| = C, |h(\mathbf{x_i})| \ge \varepsilon\} \\ \mathbf{S} &= \{i \mid 0 < |\lambda_i| < C, |h(\mathbf{x_i})| = \varepsilon\} \\ \mathbf{R} &= \{i \mid |\lambda_i| = 0, |h(\mathbf{x_i})| \le \varepsilon\} \end{aligned} \tag{14}$$

The optimization problem formulated in (10) and (11) is convenient for offline training. In online learning, when a new training instance is received, the distribution of all samples changes. Therefore, it is required to adjust all Lagrange parameters of the SVR. Depending on this adjustment, the slope [in other words the weight vector ($\mathbf{w}$)] and bias of the regressor alternate. As a result of this alternation, some samples may immigrate to other classes as depicted in Fig. 2. Therefore, when a new sample is learned by the regressor, the value of the corresponding Lagrange variable for this new sample is determined by taking into consideration all possible immigrations among classes. Let us assume that the error margin function at time index n is

$$h^{\text{old}}(\mathbf{x_i}) = f^{\text{old}}(\mathbf{x_i}) - y_i = \sum_{j=1}^{N} \lambda_j^{\text{old}} K_{ij} + b^{\text{old}} - y_i \tag{15}$$

When a new sample is introduced, a new Lagrange parameter (λ_c^{new}) is assigned to the new corresponding sample. Thus, the error margin function values of all samples including the last one are expressed as

$$h^{\text{new}}(\mathbf{x_i}) = f^{\text{new}}(\mathbf{x_i}) - y_i = K_{ic} \underbrace{(\lambda_c^{\text{old}} + \Delta\lambda_c)}_{\lambda_c^{\text{new}}} + \sum_{j=1}^{N} \underbrace{(\lambda_j^{\text{old}} + \Delta\lambda_j)}_{\lambda_j^{\text{new}}} K_{ij} + \underbrace{(b^{\text{old}} + \Delta b)}_{b_{new}} - y_i \tag{16}$$

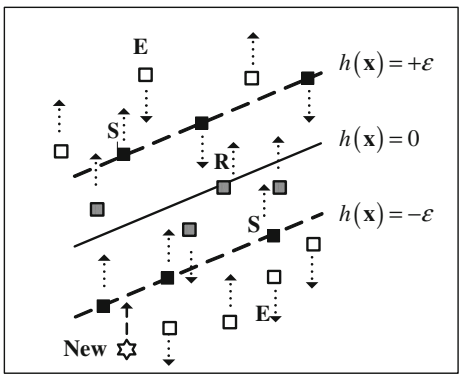

Fig. 2 Migrations among subsets **E**, **R** and **S**

Thus, the alternation in error margin values can be acquired as

$$\begin{aligned} \Delta h(\mathbf{x_i}) &= h^{\text{new}}(\mathbf{x_i}) - h^{\text{old}}(\mathbf{x_i}) \\ &= K_{ic}\Delta\lambda_c + \sum_{j=1}^{N} K_{ij}\Delta\lambda_j + \Delta b \\ \Delta\lambda_c &= \lambda_c^{\text{new}} - \lambda_c^{\text{old}} \ , \\ \Delta\lambda_j &= \lambda_j^{\text{new}} - \lambda_j^{\text{old}} \ , \ \Delta b = b^{\text{new}} - b^{\text{old}} \end{aligned} \tag{17}$$

As a result of the inclusion of the new sample, the dual constraints of the problem are given as

$$\left.\begin{aligned} &\textbf{KKT Condition (Step } n\textbf{)} : && \sum_{j=1}^{N} \lambda_j^{\text{old}} = 0 \\ &\textbf{KKT Condition (Step } n+1\textbf{)} : && \underbrace{(\lambda_c^{\text{old}} + \Delta\lambda_c)}_{\lambda_c^{\text{new}}} + \sum_{j=1}^{N} \underbrace{(\lambda_j^{\text{old}} + \Delta\lambda_j)}_{\lambda_j^{\text{new}}} = 0 \end{aligned}\right\} \quad \Delta\lambda_c + \sum_{j=1}^{N} \lambda_j = 0 \tag{18}$$

Thus, the rule derived from dual constraints of the problem can be obtained with respect to the Lagrange parameter of the new sample as

$$\lambda_c + \sum_{j=1}^{N} \lambda_j = 0 \tag{19}$$

As can be clearly seen from (14), if any vector related to previous or new data is an element of the subset **E** or **R**, the corresponding value of the Lagrange multiplier (λ_c) is set to "0" or "C". The alternation of error margin function of samples in **S** is $\Delta h(\mathbf{x_i}) = 0$ (Ma et al. 2003; Wang et al. 2009; Martin 2002). By isolating the $\Delta\lambda_c$ term in Eqs. (17) and (19), the update rule for the data in subset **S** can be easily derived with respect to obtained $\Delta\lambda_c$ as follows (Martin 2002):

$$\begin{aligned} \sum_{j=1}^{N} K_{ij}\Delta\lambda_j + \Delta b &= -K_{ic}\Delta\lambda_c \\ \sum_{j\in SV} \Delta\lambda_j &= -\Delta\lambda_c \end{aligned} \tag{20}$$

The summation terms in (20) can be given in matrix form as (Uçak and Günel 2016, 2017a; Ma et al. 2003)

$$\begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 & K_{s_1s_1} & \cdots & K_{s_1s_k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K_{s_ks_1} & \cdots & K_{s_ks_k} \end{bmatrix} \begin{bmatrix} \Delta b \\ \Delta\lambda_{s_1} \\ \vdots \\ \Delta\lambda_{s_k} \end{bmatrix} = - \begin{bmatrix} 1 \\ K_{s_1c} \\ \vdots \\ K_{s_kc} \end{bmatrix} \Delta\lambda_c \tag{21}$$

Thus, the update rule $\Delta\boldsymbol{\lambda}$ can be rewritten as

$$\Delta\boldsymbol{\lambda} = \begin{bmatrix} \Delta b \\ \Delta\lambda_{s_1} \\ \vdots \\ \Delta\lambda_{s_k} \end{bmatrix} = \boldsymbol{\beta}\Delta\lambda_c \tag{22}$$

where

$$\boldsymbol{\beta} = \begin{bmatrix} \beta \\ \beta_{s_1} \\ \vdots \\ \beta_{s_k} \end{bmatrix} = -\boldsymbol{\Theta}\begin{bmatrix} 1 \\ K_{s_1 c} \\ \vdots \\ K_{s_k c} \end{bmatrix}, \quad \boldsymbol{\Theta} = \begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 & K_{s_1 s_1} & \cdots & K_{s_1 s_k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K_{s_k s_1} & \cdots & K_{s_k s_k} \end{bmatrix}^{-1} \tag{23}$$

as given in Uçak and Günel (2016, 2017a) and Ma et al. (2003). The alternation in error margin values for non-support samples (**E** and **R**) can be computed with respect to the Lagrange parameter of the new sample as follows:

$$\begin{bmatrix} \Delta h(\mathbf{x}_{z_1}) \\ \Delta h(\mathbf{x}_{z_2}) \\ \vdots \\ \Delta h(\mathbf{x}_{z_m}) \end{bmatrix} = \boldsymbol{\gamma}\Delta\lambda_c, \quad \boldsymbol{\gamma} = \begin{bmatrix} K_{z_1 c} \\ K_{z_2 c} \\ \vdots \\ K_{z_m c} \end{bmatrix} + \begin{bmatrix} 1 & K_{z_1 s_1} & \cdots & K_{z_1 s_l} \\ 1 & K_{z_2 s_1} & \cdots & K_{z_2 s_l} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K_{z_m s_1} & \cdots & K_{z_m s_l} \end{bmatrix}\boldsymbol{\beta} \tag{24}$$

where $z_1, z_2, \ldots, z_m$ are the indices of non-support samples, $\boldsymbol{\gamma}$ are margin sensitivities and $\gamma_i = 0$ for samples in **S** (Uçak and Günel 2016, 2017a; Ma et al. 2003). Thus, the update rules given for support set samples (**S**) and non-support samples (**E** and **R**) enable us to adjust all λ_i and $h(\mathbf{x_i})$ via given $\Delta\lambda_c$ (Ma et al. 2003). The $\Delta\lambda_c$ term is obtained by considering all possible immigrations. The bookkeeping procedure to trace the immigrations of the all samples is detailed in Ma et al. (2003), Wang et al. (2009) and Martin (2002).

3 Sliding mode controller structure based on SVR

3.1 An overview of sliding mode control

Sliding mode control (SMC) is a robust control technique which is based on forcing system states onto a predefined sliding surface and keeping the states on this surface thereafter. Once on the sliding surface, the system is said to be in sliding mode and the dynamics of the system are represented by the equation of the sliding surface (Liu and Wang 2012; Slotine and Li 1991). The graphical illustration of SMC is presented in Fig. 3. The behaviour of SMC can be examined in two modes: reaching and sliding modes. In reaching mode,

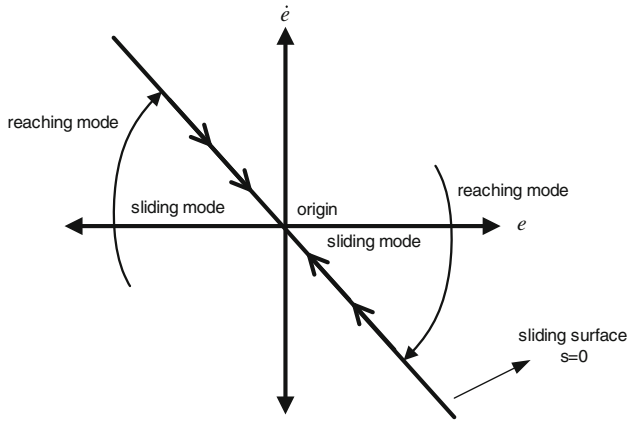

Fig. 3 Graphical interpretation of SMC (Tsai et al. 2004)

the states of the system are wafted to the sliding surface from any arbitrary point. As can be explicitly seen from Fig. 3, the reaching time of the system states to sliding surface depends on the slope of the sliding surface and the accuracy of the system model since the system model is employed to derive the control law. In sliding mode, the aim is to hold the dynamics of the system on the predefined sliding surface in order to force the error dynamics to origin. Because of the model uncertainty or external disturbances, etc., the system states tend to deviate from the sliding surface. Therefore, in sliding mode, a switching control law is utilized to retract the deviant states back to the sliding surface. Let us consider a second-order nonlinear system in order to derive the control laws for reaching and sliding modes

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= -f(\mathbf{X}(t)) + g(\mathbf{X}(t))u(t) = \ddot{x}_1(t) \\ y(t) &= x_1(t) \end{aligned} \tag{25}$$

where $u(t)$ denotes the control signal, $y(t)$ is the controlled output of the system, $f(\mathbf{X}(t))$ and $g(\mathbf{X}(t))$ are nonlinear functions and $\mathbf{X} \in R^n$ is the state vector. It is required that $g(\mathbf{X}(t)) \neq 0$ for the system in (25) to be controllable (Hua et al. 2015). The PD-type sliding surface can be defined as

$$s(t) = \dot{e}(t) + \theta e(t) \tag{26}$$

where $e(t)$ is tracking error given in (27), θ stands for the slope of the PD-type sliding surface and $\theta > 0$ must satisfy Hurwitz condition (Liu and Wang 2012).

$$e(t) = r(t) - y(t) \tag{27}$$

where $r(t)$ represents the desired output of the system. In order to derive the stable control law, the Lyapunov function can be defined with respect to sliding surface ($s(t)$) as

$$V(t) = \frac{1}{2}[s(t)]^2 \tag{28}$$

For stability, the Lyapunov function must satisfy $V(t) > 0$ and $\dot{V}(t) < 0$ for $s(t) \neq 0$ (Ertuğrul et al. 1995; Zribi and Oteafy 2006; Derdiyok and Levent 2000). Thus, $\dot{V}(t)$ can be acquired as

$$\dot{V}(t) = s(t)\dot{s}(t) < 0 \tag{29}$$

The system must satisfy (29), and the signs of $s(t)$ and $\dot{s}(t)$ functions should be opposite to ensure finite time reaching (Bandyopadhyay et al. 2009) and convergence to sliding surface. For this purpose, the following conditions should be satisfied (Bandyopadhyay et al. 2009)

$$\begin{aligned} \lim_{s\to 0_+} \dot{s}(t) < 0 \\ \lim_{s\to 0_-} \dot{s}(t) > 0 \end{aligned} \tag{30}$$

Since the states of the system can be forced from an arbitrary point to the sliding surface as long as (29) is satisfied, the condition in (29) is called as *reachability condition* (Tsai et al. 2004; Bandyopadhyay et al. 2009). Condition (29) ensures only asymptotic reaching to the sliding surface (Bandyopadhyay et al. 2009). A stronger condition for finite time reaching, known as η-*reachability condition*, is given as follows:

$$s(t)\dot{s}(t) < -\eta|s(t)| \tag{31}$$

where $\eta > 0$ (Bandyopadhyay et al. 2009). After the system states are on the sliding surface, the alternation of the system dynamics on sliding surface must be zero ($\dot{s}(t) = 0$), so that the system dynamics is held on the sliding surface. For this purpose, the control signal maintaining the system states on the sliding surface which is called as equivalent control signal (u_{eq}) can be derived as

$$\begin{aligned} \dot{s}(t) &= \ddot{e}(t) + \theta\dot{e}(t) = \ddot{r}(t) - \ddot{y}(t) + \theta\dot{e}(t) = \ddot{r}(t) \\ &\quad + f(\mathbf{X}(t)) - g(\mathbf{X}(t))u_{\text{eq}}(t) + \theta\dot{e}(t) = 0 \\ u_{\text{eq}}(t) &= \frac{1}{g(\mathbf{X}(t))}\left[\ddot{r}(t) + f(\mathbf{X}(t)) + \theta\dot{e}(t)\right] \end{aligned} \tag{32}$$

Owing to the external disturbances or uncertainties in system dynamics, the states of the system may digress from sliding surface. Therefore, in order to retract system behaviour to the sliding surface, a switching control function satisfying the Lyapunov stability theory and reachability condition in (29) can be derived as

$$u_{\text{sw}}(t) = \frac{1}{g(\mathbf{X}(t))}\left[\mu\ \text{sgn}(s(t))\right] \tag{33}$$

Thus, the control signal applied to the controlled system can be acquired as the combination of equivalent control signal (32) and switching control signal (33) as follows:

$$\begin{aligned} u(t) &= u_{\text{eq}}(t) + u_{\text{sw}}(t) \\ &= \frac{1}{g(\mathbf{X}(t))}\left[\ddot{r}(t) + f(\mathbf{X}(t)) + \theta\dot{e}(t) + \mu\ \text{sgn}(s(t))\right] \end{aligned} \tag{34}$$

By substituting (34) in (29), the obtained control signal can be evaluated as to whether reachability condition is satisfied or not as follows

$$\begin{aligned} \dot{V}(t) &= s(t)\left[-\mu\ \text{sgn}(s(t))\right] \\ &= -\mu\, s(t)\ \text{sgn}(s(t)) < 0, \quad \mu > 0, \quad s(t) \neq 0 \end{aligned} \tag{35}$$

The structure of the SMC is illustrated in Fig. 4 where $u_{\text{eq}}(t)$ is equivalent control signal, $u_{\text{sw}}(t)$ represents the switching control signal, $r(t)$ denotes the reference signal system output which is forced to track and $y(t)$ stands for the system output. As mentioned before, u_{sw} is utilized to ensure stability of the closed-loop system and u_{eq} forces the system states to the origin on the sliding surface. Since the sliding surface is constituted via Lyapunov's stability theory, robust tracking is assured when system states are on the sliding surface, whereas robustness is not guaranteed during the reaching phase (Bartoszewicz 1996). Because of the discontinuity in the sign function in u_{sw}, chattering phenomenon stimulating the high-frequency unmodelled dynamics of the system is generally observed on the sliding surface and the produced control signal becomes more sensitive to measurement noise; also fragility of the controller against measurement noise increases. Various solutions have been proposed to suppress chattering by introducing continuous functions in place of sign function in order to provide smooth transition. Several functions utilized to overcome chattering are illustrated in Fig. 5. Thus, the control signal in (34) can be rewritten as follows:

$$\begin{aligned} u(t) &= u_{\text{eq}}(t) + u_{\text{sw}}(t) \\ &= \frac{1}{g(\mathbf{X}(t))}\left[\ddot{r}(t) + f(\mathbf{X}(t)) + \theta\dot{e}(t) + \mu f_{\text{sw}}(s(t), \rho(t))\right] \end{aligned} \tag{36}$$

where $f_{\text{sw}}(s(t), \rho(t))$ is the switching function and $\rho(t)$ is the parameter of the switching function. As can be explicity seen from the control signal in (36), the closed-loop tracking performance of the SMC depends on the parameters of the control law in (36). Thus, the control law can be rewritten with respect to adjustable parameters as follows:

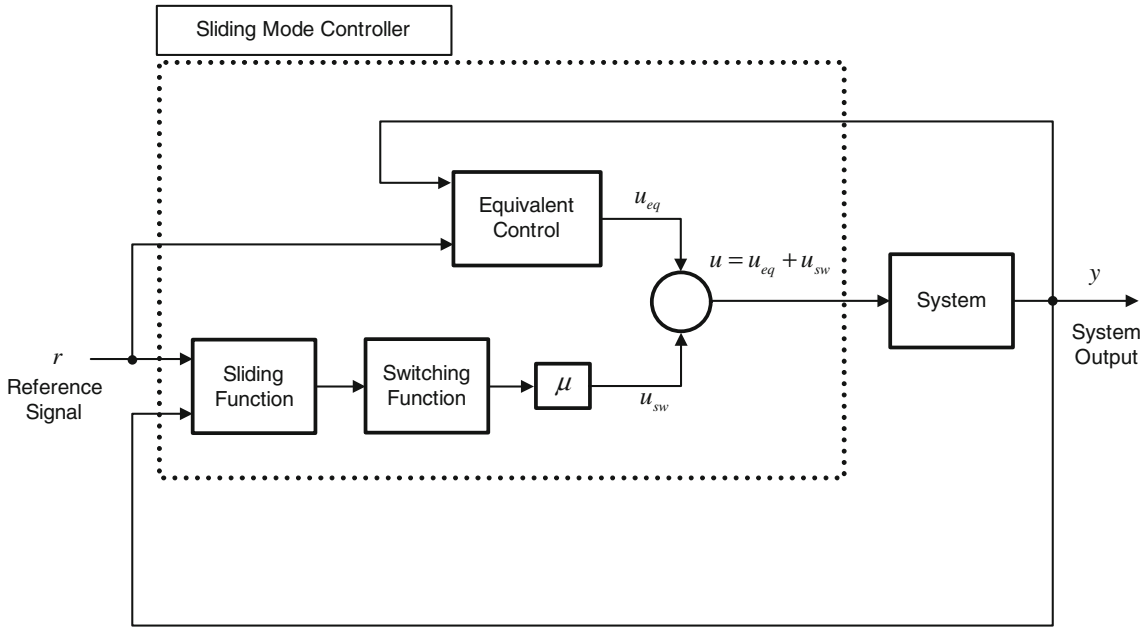

Fig. 4 Sliding mode controller

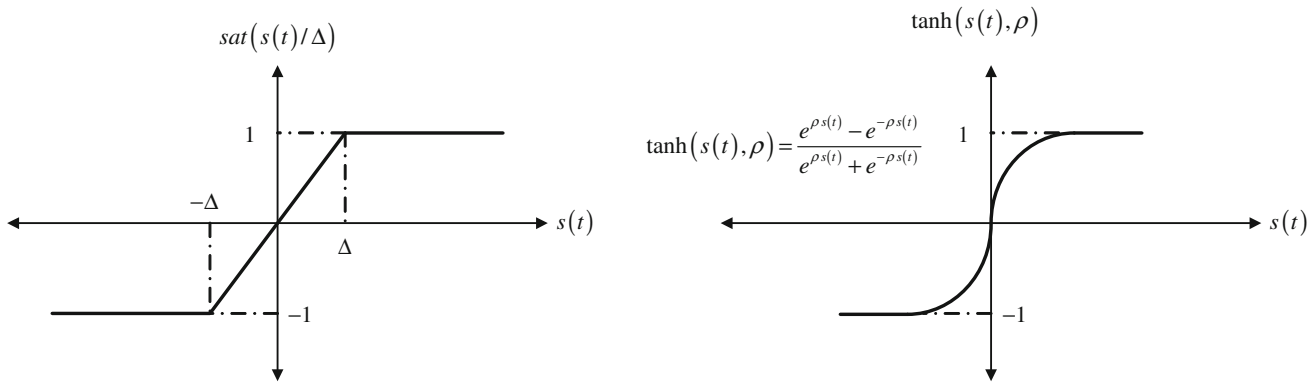

Fig. 5 Various switching functions to avoid chattering

$$u(t) = u_{\text{eq}}(t) + u_{\text{sw}}(t) = \frac{1}{g(\mathbf{X}(t))}\left[\ddot{r}(t) + f(\mathbf{X}(t)) + \theta(t)\dot{e}(t) + \mu(t) f_{\text{sw}}(s(t), \rho(t))\right] \cong \frac{1}{g}\left[\ddot{r} + \hat{f} + \hat{\theta}\dot{e} + \hat{\mu} f_{\text{sw}}(s, \hat{\rho})\right] \quad (37)$$

where $\theta(t)$, $\mu(t)$, $\rho(t)$ are unknown tunable parameters, $\hat{f}$ denotes the estimated system dynamics, and $\hat{\theta}$, $\hat{\mu}$, $\hat{\rho}$ are approximated values.

3.2 Sliding mode control input derivation based on SVR

The adjustable parameters of the SMC in (37) can be optimized using artificial intelligence (AI)-based models. The superior generalization capability of SVR makes it a very good candidate to solve regression problems among other intelligent techniques. Therefore, in this paper, we employ SVR models to identify both the dynamics of the controlled system and the adjustable parameters of the SMC law. The overall architecture of the proposed SMC based on SVR is depictured in Fig. 6. The adjustment mechanism is composed of two separate online trained SVR structures: $\text{SVR}_{\text{estimator}}$ to identify the adjustable parameters of the SMC in (37) and $\text{SVR}_{\text{model}}$ which approximates the future behaviour of the controlled system in response to the adjustments in control law. Owing to the multi-input single-output (MISO) structure of SVR, a separate $\text{SVR}_{\text{estimator}}$ is employed for each approximated component of the SMC (Uçak and Günel 2017a). Therefore, the regression functions in $\text{SVR}_{\text{estimator}}$ structure are given as follows:

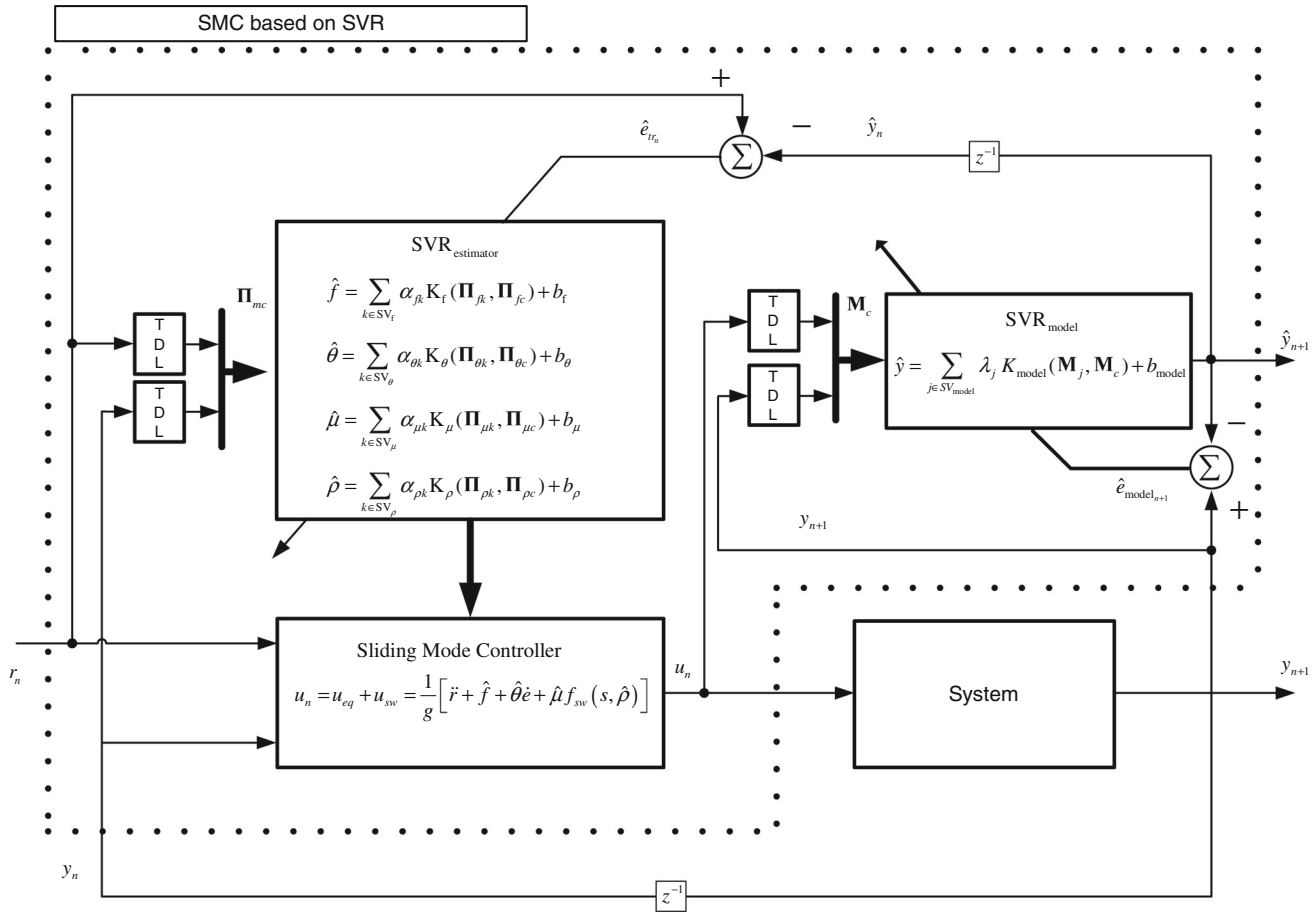

Fig. 6 Adaptive SMC based on online SVR

$$\hat{f} = \sum_{k\in \mathrm{SV}_f} \alpha_{fk} K_f\left(\mathbf{\Pi}_{fk}, \mathbf{\Pi}_{fc}\right) + b_f$$

$$\hat{\theta} = \sum_{k\in \mathrm{SV}_\theta} \alpha_{\theta k} K_\theta\left(\mathbf{\Pi}_{\theta k}, \mathbf{\Pi}_{\theta c}\right) + b_\theta$$

$$\hat{\mu} = \sum_{k\in \mathrm{SV}_\mu} \alpha_{\mu k} K_\mu\left(\mathbf{\Pi}_{\mu k}, \mathbf{\Pi}_{\mu c}\right) + b_\mu$$

$$\hat{\rho} = \sum_{k\in \mathrm{SV}_\rho} \alpha_{\rho k} K_\rho\left(\mathbf{\Pi}_{\rho k}, \mathbf{\Pi}_{\rho c}\right) + b_\rho \tag{38}$$

where α_{fk}, $\alpha_{\theta k}$, $\alpha_{\mu k}$, $\alpha_{\rho k}$ denote the kth Lagrange parameters, $\mathbf{\Pi}_{fk}$, $\mathbf{\Pi}_{\theta k}$, $\mathbf{\Pi}_{\mu k}$, $\mathbf{\Pi}_{\rho k}$ are the corresponding support vectors, $\mathbf{\Pi}_{fc}$, $\mathbf{\Pi}_{\theta c}$, $\mathbf{\Pi}_{\mu c}$, $\mathbf{\Pi}_{\rho c}$ stand for the current input feature vectors of estimators, $K_f(,)$, $K_\theta(,)$, $K_\mu(,)$, $K_\rho(,)$ are the kernel function, and b_f, b_θ, b_μ, b_ρ represent the bias of the regressors. Thus, the control signal produced by SMC can be expressed as:

$$u(t) = u_{\mathrm{eq}}(t) + u_{\mathrm{sw}}(t) = \frac{1}{g(\mathbf{X}(t))}\left[\ddot{r}(t) + \hat{f}\left(\mathbf{\Pi}_{fc}\right) + \hat{\theta}\left(\mathbf{\Pi}_{\theta c}\right)\dot{e}(t) + \hat{\mu}\left(\mathbf{\Pi}_{\mu c}\right) f_{\mathrm{sw}}\left(s(t), \hat{\rho}\left(\mathbf{\Pi}_{\rho c}\right)\right)\right] \tag{39}$$

$\mathrm{SVR}_{\mathrm{model}}$ is employed to predict the effects of adjustment in control parameters on system dynamics in advance. The output of $\mathrm{SVR}_{\mathrm{model}}$ is calculated as

$$\hat{y}_{n+1} = f_{\mathrm{model}}\left(\mathbf{M}_{\mathrm{c}}\right) = \sum_{j\in \mathrm{SV}_{\mathrm{model}}} \lambda_j K_{\mathrm{model}}\left(\mathbf{M}_j, \mathbf{M}_{\mathrm{c}}\right) + b_{\mathrm{model}} \tag{40}$$

where λ_j's and $\mathbf{M}_j$'s denote the Lagrange parameters and corresponding support vectors, respectively, $\mathbf{M}_{\mathrm{c}}$ is the current input, b_{model} is the bias of the regressor and $K_{\mathrm{model}}(,)$ is the kernel function (Uçak and Günel 2016, 2017a). $\mathrm{SVR}_{\mathrm{estimator}}$ and $\mathrm{SVR}_{\mathrm{model}}$ are both deployed online to perform learning, prediction and control consecutively (Uçak and Günel 2016, 2017a). Both $\mathrm{SVR}_{\mathrm{estimator}}$ and $\mathrm{SVR}_{\mathrm{model}}$ have two phases: prediction and training(learning) phases. In training phase of $\mathrm{SVR}_{\mathrm{estimator}}$, $\mathrm{SVR}_{\mathrm{model}}$ is employed in prediction phase and vice versa. The adjustment mechanism can be briefly summarized as follows: Firstly, in training phase of the $\mathrm{SVR}_{\mathrm{estimator}}$, using the previously calculated parameters of the $\mathrm{SVR}_{\mathrm{estimator}}$, SMC parameters are estimated and the approximate control signal (u_n) is computed. In order to observe the possible impact of the computed control sig-

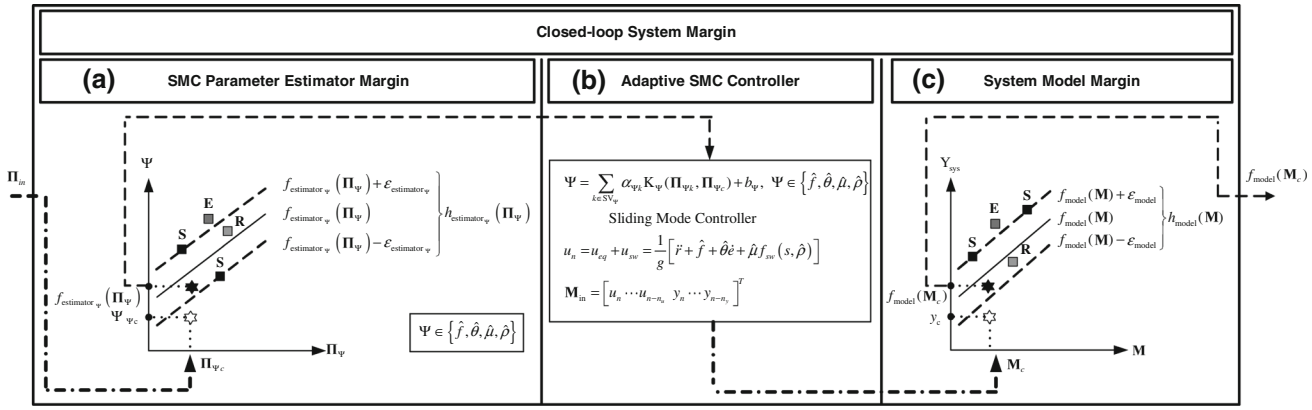

Fig. 7 Margins of controller parameter estimator − $\text{SVR}_{\text{estimator}}$ (**a**), adaptive sliding mode controller (**b**) and system model-$\text{SVR}_{\text{model}}$ (**c**)

nal (u_n) on system behaviour, the obtained control signal is applied to the $\text{SVR}_{\text{model}}$ since ideally, during the course of online working, it is expected that $\hat{y}_{n+1}$ will eventually converge to y_{n+1} (Uçak and Günel 2017a). Then, utilizing the approximated tracking error, the adjustable parameters of the $\text{SVR}_{\text{estimator}}$ are optimized. Thus, the training phase of the $\text{SVR}_{\text{estimator}}$ can be accomplished. In training phase of $\text{SVR}_{\text{model}}$, using the trained $\text{SVR}_{\text{estimator}}$ parameters, the optimized control signal is applied to both the real system and $\text{SVR}_{\text{model}}$. Thus, the current input of system model $\mathbf{M}_c$ and output y_{n+1} can be acquired for training phase of $\text{SVR}_{\text{model}}$, the prediction accuracy of the $\text{SVR}_{\text{model}}$ can be evaluated, and the parameters of the $\text{SVR}_{\text{model}}$ can be adjusted using modelling error. Hence, one cycle of the control algorithm can be summarized as above. The pseudoalgorithm for the SMC adjustment mechanism is detailed in Sect. 3.3. The adjustment structure for SMC illustrated in Fig. 6 can be represented with respect to regression margins of $\text{SVR}_{\text{estimator}}$ and $\text{SVR}_{\text{model}}$ as in Fig. 7. As can be seen from Fig. 7, since the training data pairs ($\mathbf{M}_c$, y_{n+1}) for $\text{SVR}_{\text{model}}$ are available during online operation, the training process is carried out in a straightforward manner as explained in Sect. 2 (Uçak and Günel 2017a). In training phase of the $\text{SVR}_{\text{model}}$, the output of the $\text{SVR}_{\text{model}}$ ($\hat{y}_{n+1}$) is forced to track actual system output, so ($\mathbf{M}_c$, y_{n+1}) is utilized as training data pair (Uçak and Günel 2016, 2017a). However, training of $\text{SVR}_{\text{estimator}}$ presents some difficulties. Whereas the input data ($\mathbf{\Pi}_{\Psi c}$, $\Psi \in \{\hat{f}, \hat{\theta}, \hat{\mu}, \hat{\rho}\}$) for $\text{SVR}_{\text{estimator}}$ are procurable, the desired outputs of the $\text{SVR}_{\text{estimator}}$, namely the SMC parameters ($\Psi \in \{\hat{f}, \hat{\theta}, \hat{\mu}, \hat{\rho}\}$), are not known by the designer in advance (Uçak and Günel 2016, 2017a). Therefore, the parameters of the $\text{SVR}_{\text{estimator}}$ can be optimized without the explicit information of desired output training data, using the closed-loop margin notion proposed in Uçak and Günel (2016, 2017a). In closed-loop system, the aim of the controller is to compel the output of the system (y_{n+1}) to track reference signal (r_{n+1}). Therefore, ($\mathbf{\Pi}_{\Psi c}$, r_{n+1}) data pair is deployed as training data for $\text{SVR}_{\text{estimator}}$. Thus, the proposed parameter estimator in Uçak and Günel (2017a) can be deployed to approximate the parameters of the SMC. The constitution of the closed-loop margin is detailed in Uçak and Günel (2016, 2017a).

3.3 Adaptive control algorithm for the SMC based on SVR

In this section, the step-by-step algorithm of the proposed control procedure of SMC based on SVR illustrated in Fig. 6 is presented. In the algorithm given below u_n^- represents the computed control signal using the SMC controller parameters obtained at the previous step and u_n^+ stands for the control signal calculated via trained SMC controller parameters at the current step. The performances of the regressors for both $\text{SVR}_{\text{model}}$ and $\text{SVR}_{\text{estimator}}$ are closely related to the input features chosen to construct input feature vectors. Since input and output samples of controlled system are the features of open-loop system, the input feature vector for $\text{SVR}_{\text{model}}$ is constructed using input–output samples of the open-loop system as $\mathbf{M}_c = \left[u_n \ldots u_{n-n_u}, y_n \ldots y_{n-n_y}\right]^T$ where n_u and n_y denote the number of the past instances of features. Similarly, the input feature vector of the $\text{SVR}_{\text{estimator}}$ is constituted using closed-loop system features such as reference signal, system output, control signal and tracking error. Input feature vector of parameter estimator ($\mathbf{\Pi}$) should contain convenient feature variables that can well represent the closed-loop system's operating conditions (Uçak and Günel 2017a). In the proposed SMC, mainly reference signal (r) and system output (y) can be deployed as input features. However, when they are inadequate in closed-loop tracking performance, new variables that are functions of reference and system output such as tracking error, integral of tracking error and derivative of tracking

error can be utilized in order to enhance SMC performance (Uçak and Günel 2016, 2017a). Some examples for parameter estimator feature vectors are given as $\mathbf{\Pi}_c = [r_n \dots r_{n-n_r}, y_n \dots y_{n-n_y}]^{\mathrm{T}}$ or $\mathbf{\Pi}_c = [P_n, I_n, D_n]^{\mathrm{T}}$ where $P_n = e_n - e_{n-1}$,$I_n = e_n$, $D_n = e_n - 2e_{n-1} + e_{n-2}$ and $e_n = r_n - y_n$ (Uçak and Günel 2016, 2017a). Combination of the reference signal, system output and controller output can also be utilized in the feature vector as $\mathbf{\Pi}_c = [\, P_n, I_n, D_n, r_n \dots r_{n-n_r}, y_n \dots y_{n-n_y}, u_{n-1} \dots u_{n-n_u}\,]^{\mathrm{T}}$ where n_r, n_y and n_u represent the number of the past instances of features included in the feature vector (Uçak and Günel 2016, 2017a).

Step 1 Initialization of $\mathrm{SVR}_{\mathrm{estimator}}$ and $\mathrm{SVR}_{\mathrm{model}}$ parameters.

- $\mathrm{SVR}_{\mathrm{estimator}}$(estimator) parameters : $\alpha_{\Psi k} = b_{\Psi} = 0$, $\Psi \in \{\hat{f}, \hat{\theta}, \hat{\mu}, \hat{\rho}\}$
- $\mathrm{SVR}_{\mathrm{model}}$ (system model) parameters : $\lambda_j = b_{model} = 0$

Step 2 Prediction step for parameter estimator ($\Psi^- \in \{\hat{f}^-, \hat{\theta}^-, \hat{\mu}^-, \hat{\rho}^-\}$)

- Set time step n.
- Constitute feature vector for parameter estimator ($\mathbf{\Pi}_{\Psi c}$).
- Calculate the approximated controller parameters $\Psi^- \in \{\hat{f}^-, \hat{\theta}^-, \hat{\mu}^-, \hat{\rho}^-\}$ by $\mathrm{SVR}_{\mathrm{estimator}}$ trained at previous step $(n-1)$ via (38).

Step 3 Computation of control signal (u_n^-) and prediction step for system model($\hat{y}_{n+1}^-$)

- Compute the control signal u_n^- via (38) and (39).
- Constitute feature vector for $\mathrm{SVR}_{\mathrm{model}}$ ($\mathbf{M}_{\mathrm{c}}$).

$$\mathbf{M}_{\mathrm{c}} = [u_n^- \dots u_{n-n_u}, y_n \dots y_{n-n_y}]$$

- Apply u_n^- to $\mathrm{SVR}_{\mathrm{model}}$ and calculate $\hat{y}_{n+1}^-$ by (40).

Step 4 Training step for parameter estimator

- Calculate $\hat{e}_{tr_{n+1}} = r_{n+1} - \hat{y}_{n+1}^-$

If $|\hat{e}_{tr_{n+1}}| > \varepsilon_{\text{closed-loop}}$
Train $\mathrm{SVR}_{\mathrm{estimator}}$ parameters via $\hat{e}_{tr_{n+1}} = r_{n+1} - \hat{y}_{n+1}^-$
else
Continue with $\mathrm{SVR}_{\mathrm{estimator}}$ parameters trained at previous step
end

Step 5 Prediction step for trained parameter estimator ($\Psi^+ \in \{\hat{f}^+, \hat{\theta}^+, \hat{\mu}^+, \hat{\rho}^+\}$) and computation of control input by trained estimator (u_n^+)

- Calculate the controller parameters by trained $\mathrm{SVR}_{\mathrm{estimator}}$ via (38).
- Calculate the control signal u_n^+ produced by the controller using the parameters obtained by trained $\mathrm{SVR}_{\mathrm{estimator}}$ via (38) and (39).

Step 6 Application of the control signal produced by SMC controller

- Apply u_n^+ to system to calculate y_{n+1}.

Step 7 Prediction and training step for $\mathrm{SVR}_{\mathrm{model}}$ ($\hat{y}_{n+1}^+$)

- Apply u_n^+ to $\mathrm{SVR}_{\mathrm{model}}$ and calculate $\hat{y}_{n+1}^+$ via (40).
- Calculate $e_{\mathrm{model}_{n+1}} = y_{n+1} - \hat{y}_{n+1}$
  If $|e_{\mathrm{model}_{n+1}}| > \varepsilon_{\mathrm{model}}$
  Train $\mathrm{SVR}_{\mathrm{model}}$ where $e_{\mathrm{model}_{n+1}} = y_{n+1} - \hat{y}_{n+1}$
  else
  Continue with $\mathrm{SVR}_{\mathrm{model}}$ parameters obtained at previous step
  end

Step 8 Incrementation of time step

- Increment $n = n + 1$ and back to step 2.

3.4 Online support vector regression for parameter estimator

As mentioned in Sect. 3.2, both $\mathrm{SVR}_{\mathrm{estimator}}$ utilized to tune the controller parameters and $\mathrm{SVR}_{\mathrm{model}}$ used to identify the dynamics of the controlled system are deployed in online manner. In this subsection, online tuning rules for $\mathrm{SVR}_{\mathrm{estimator}}$ parameters are derived. Let us consider the training data set used for the closed-loop system:

$$\begin{aligned} &\mathbf{T} = \{\mathbf{\Pi}_{\Psi i}, r_{i+1}\}_{i=1}^{N}\,,\ \mathbf{\Pi}_{\Psi i} \in \mathbf{\Pi} \subseteq R^n, r_{i+1} \in R \\ &\Psi \in \{\hat{f}, \hat{\theta}, \hat{\mu}, \hat{\rho}\} \end{aligned} \tag{41}$$

where N and n denote the number of training samples and dimension of the input samples, respectively, $\mathbf{\Pi}_{\Psi i}$ indicates the input feature vector of corresponding parameter estimator and r_{i+1} is the reference signal that system is forced to chase. The closed-loop error margin function of the system for the ith sample $\mathbf{\Pi}_{\Psi i}$, $\Psi \in \{\hat{f}, \hat{\theta}, \hat{\mu}, \hat{\rho}\}$ can be defined as

$$\begin{aligned} h_{\text{closed-loop}}\left(\left[\mathbf{\Pi}_{\hat{f}i}\ \mathbf{\Pi}_{\hat{\theta}i}\ \mathbf{\Pi}_{\hat{\mu}i}\ \mathbf{\Pi}_{\hat{\rho}i}\right]\right) &= \hat{y}_{i+1} - r_{i+1} \\ &= f_{\mathrm{model}}(\mathbf{M}_i) - r_{i+1} \end{aligned} \tag{42}$$

where

$$\hat{y}_{i+1} = f_{\text{model}}(\mathbf{M}_i) = \sum_{j \in \text{SV}_{\text{model}}} \lambda_j K_{\text{model}}(\mathbf{M}_j, \mathbf{M}_i) + b_{\text{model}}$$
$$\lambda_j = \beta_j - \beta_j^{\star}$$
$$\mathbf{M}_i = [u_i \ldots u_{i-n_u},\ y_i \ldots y_{i-n_y}]$$
$$u_i = \frac{1}{g}\big[\ddot{r} + \hat{f}(\mathbf{\Pi}_{fi}) + \hat{\theta}(\mathbf{\Pi}_{\theta i})\dot{e}(t) + \hat{\mu}(\mathbf{\Pi}_{\mu i}) f_{\text{sw}}(s(t), \hat{\rho}(\mathbf{\Pi}_{\rho i}))\big]$$
$$\hat{f}(\mathbf{\Pi}_{fi}) = \sum_{k \in \text{SV}_f} \alpha_{fk} K_f(\mathbf{\Pi}_{fk}, \mathbf{\Pi}_{fi}) + b_f$$
$$\hat{\theta}(\mathbf{\Pi}_{\theta i}) = \sum_{k \in \text{SV}_\theta} \alpha_{\theta k} K_\theta(\mathbf{\Pi}_{\theta k}, \mathbf{\Pi}_{\theta i}) + b_\theta$$
$$\hat{\mu}(\mathbf{\Pi}_{\mu i}) = \sum_{k \in \text{SV}_\mu} \alpha_{\mu k} K_\mu(\mathbf{\Pi}_{\mu k}, \mathbf{\Pi}_{\mu i}) + b_\mu$$
$$\hat{\rho}(\mathbf{\Pi}_{\rho i}) = \sum_{k \in \text{SV}_\rho} \alpha_{\rho k} K_\rho(\mathbf{\Pi}_{\rho k}, \mathbf{\Pi}_{\rho i}) + b_\rho$$
$$\mathbf{\Pi}_{fi} = [r_i \ldots r_{i-n_{r_f}},\ y_i \ldots y_{i-n_{y_f}},\ u_{i-1} \ldots u_{i-n_{u_f}}]^{\text{T}}$$
$$\mathbf{\Pi}_{\theta i} = [r_i \ldots r_{i-n_{r_\theta}},\ y_i \ldots y_{i-n_{y_\theta}},\ u_{i-1} \ldots u_{i-n_{u_\theta}}]^{\text{T}}$$
$$\mathbf{\Pi}_{\mu i} = [r_i \ldots r_{i-n_{r_\mu}},\ y_i \ldots y_{i-n_{y_\mu}},\ u_{i-1} \ldots u_{i-n_{u_\mu}}]^{\text{T}}$$
$$\mathbf{\Pi}_{\rho i} = [r_i \ldots r_{i-n_{r_\rho}},\ y_i \ldots y_{i-n_{y_\rho}},\ u_{i-1} \ldots u_{i-n_{u_\rho}}]^{\text{T}} \tag{43}$$

Since $\text{SVR}_{\text{model}}$ and $\text{SVR}_{\text{estimator}}$ are deployed consecutively, the parameters of the $\text{SVR}_{\text{model}}$ are fixed and known and the sole unknown variables are the parameters of the $\text{SVR}_{\text{estimator}}$ in the training phase of the $\text{SVR}_{\text{estimator}}$ (Uçak and Günel 2017a). Therefore, the closed-loop error margin function can be expressed with respect to an input–output training data pair of closed-loop system ($\mathbf{\Pi}_{\Psi i}$, $\Psi \in \{f, \theta, \mu, \rho\}$, r_{i+1}) as

$$h_{\text{closed-loop}}\left(\left[\mathbf{\Pi}_{\hat{f}i}\ \mathbf{\Pi}_{\hat{\theta}i}\ \mathbf{\Pi}_{\hat{\mu}i}\ \mathbf{\Pi}_{\hat{\rho}i}\right]\right) = \hat{y}_{i+1} - r_{i+1} = f_{\text{closed-loop}}\left(\left[\mathbf{\Pi}_{\hat{f}i}\ \mathbf{\Pi}_{\hat{\theta}i}\ \mathbf{\Pi}_{\hat{\mu}i}\ \mathbf{\Pi}_{\hat{\rho}i}\right]\right) - r_{i+1} \tag{44}$$

Then, using data pair ($\mathbf{\Pi}_{\Psi i}$, $\Psi \in \{f, \theta, \mu, \rho\}$, r_{i+1}) and closed-loop margin given in (42,44), the incremental update rules for the parameters of the $\text{SVR}_{\text{estimator}}$ can be derived. When the regressor is exposed to new data ($\mathbf{\Pi}_{\Psi c}$, $\Psi \in \{f, \theta, \mu, \rho\}$), a new Lagrange multiplier $\alpha_{\Psi c}$ corresponding to this new sample is assigned and the coefficient $\alpha_{\Psi c}$ should be adjusted in a finite number of discrete steps until it meets the KKT conditions while ensuring the existing samples in $\mathbf{T}$ continue to satisfy the KKT conditions at each step (Ma et al. 2003). During this learning phase, some samples may migrate to other classes and there may be transitions between classes.

Therefore, the convergence of the samples and migrations among classes can be traced via the following convergence conditions (Uçak and Günel 2016; Iplikci 2006; Ma et al. 2003).

$$\begin{aligned}
&h_{\text{closed-loop}}\left(\left[\mathbf{\Pi}_{\hat{f}i}\ \mathbf{\Pi}_{\hat{\theta}i}\ \mathbf{\Pi}_{\hat{\mu}i}\ \mathbf{\Pi}_{\hat{\rho}i}\right]\right) \geq \varepsilon_{\text{closed-loop}}, \alpha_i = -C_{\text{closed-loop}}\\
&h_{\text{closed-loop}}\left(\left[\mathbf{\Pi}_{\hat{f}i}\ \mathbf{\Pi}_{\hat{\theta}i}\ \mathbf{\Pi}_{\hat{\mu}i}\ \mathbf{\Pi}_{\hat{\rho}i}\right]\right) = \varepsilon_{\text{closed-loop}}, -C_{\text{closed-loop}} < \alpha_i < 0\\
&-\varepsilon_{\text{closed-loop}} \leq h_{\text{closed-loop}}\left(\left[\mathbf{\Pi}_{\hat{f}i}\ \mathbf{\Pi}_{\hat{\theta}i}\ \mathbf{\Pi}_{\hat{\mu}i}\ \mathbf{\Pi}_{\hat{\rho}i}\right]\right) \leq \varepsilon_{\text{closed-loop}}, \alpha_i = 0\\
&h_{\text{closed-loop}}\left(\left[\mathbf{\Pi}_{\hat{f}i}\ \mathbf{\Pi}_{\hat{\theta}i}\ \mathbf{\Pi}_{\hat{\mu}i}\ \mathbf{\Pi}_{\hat{\rho}i}\right]\right) = -\varepsilon_{\text{closed-loop}}, 0 < \alpha_i < C_{\text{closed-loop}}\\
&h_{\text{closed-loop}}\left(\left[\mathbf{\Pi}_{\hat{f}i}\ \mathbf{\Pi}_{\hat{\theta}i}\ \mathbf{\Pi}_{\hat{\mu}i}\ \mathbf{\Pi}_{\hat{\rho}i}\right]\right) \leq -\varepsilon_{\text{closed-loop}}, \alpha_i = C_{\text{closed-loop}}
\end{aligned} \tag{45}$$

By substituting α_Ψ, $b_{\text{estimator}_\Psi}$, $h_{\text{closed-loop}}$, $\varepsilon_{\text{closed-loop}}$, $\mathbf{\Pi}_{\Psi i}$ and $K_{\text{estimator}_\Psi}$ in place of λ, b, h, ε, $\mathbf{x}_i$ and K in (12–24), the incremental learning algorithm can be derived for $\text{SVR}_{\text{estimator}}$. Thus, the parameters of the $\text{SVR}_{\text{estimator}}$, $\alpha_{\Psi k}$, $b_{\text{estimator}_\Psi}$ can be optimized such that the maximum tolerable training error will be equal to $\varepsilon_{\text{closed-loop}}$. The update vector ($\Delta\boldsymbol{\alpha}_\Psi$) for Lagrange multipliers of support set samples($\mathbf{S}$) in $\text{SVR}_{\text{estimator}}$ can be acquired with respect to the Lagrange multiplier of the current new sample ($\Delta\alpha_{\Psi c}$) as (Uçak and Günel 2017a):

$$\Delta\boldsymbol{\alpha}_\Psi = \begin{bmatrix} \Delta b_{\text{estimator}_\Psi} \\ \Delta\alpha_{\Psi s_1} \\ \vdots \\ \Delta\alpha_{\Psi s_k} \end{bmatrix} = \boldsymbol{\beta}_\Psi \Delta\alpha_{\Psi c} \tag{46}$$

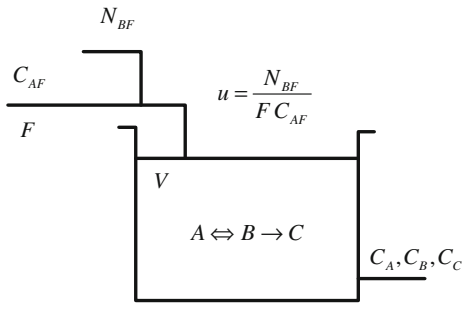

Fig. 8 CSTR system (Uçak and Günel 2017b, c; Kravaris and Palanki 1988)

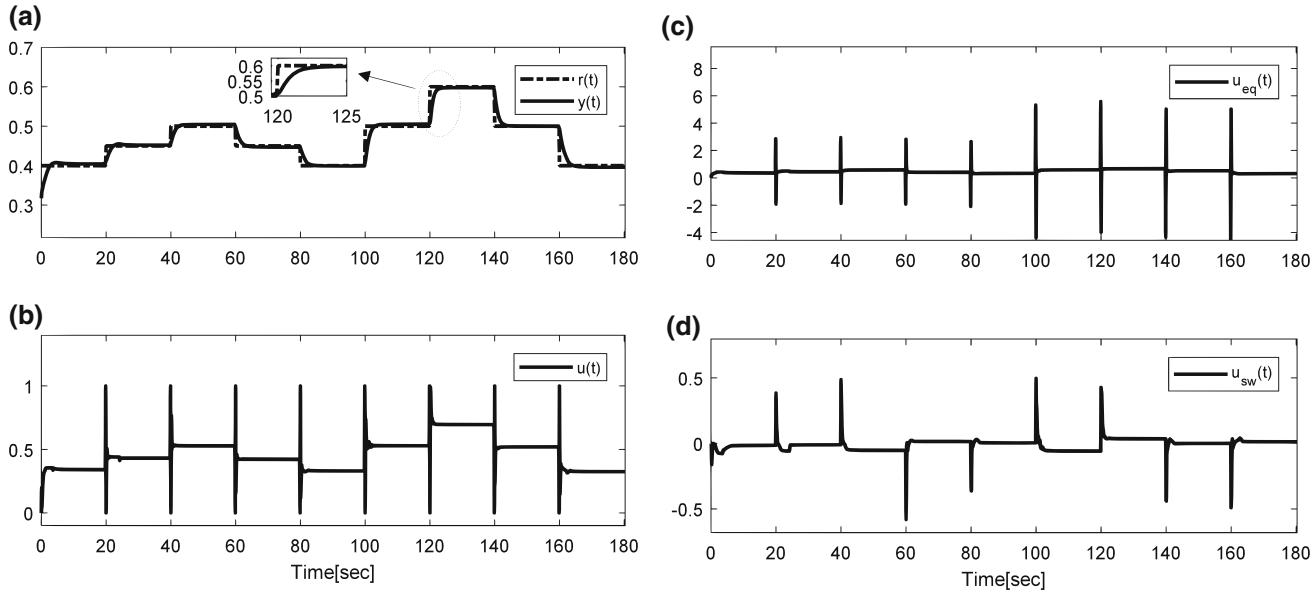

Fig. 9 System output (**a**), control signal (**b**), equivalent control (**c**) and switching control (**d**) for staircase input (without noise and parametric uncertainty case)

where

$$\boldsymbol{\beta}_{\Psi} = \begin{bmatrix} \beta \\ \beta_{\Psi s_1} \\ \vdots \\ \beta_{\Psi s_k} \end{bmatrix} = -\boldsymbol{\Theta}_{\Psi} \begin{bmatrix} 1 \\ K_{\text{estimator}_{\Psi_{s_1 c}}} \\ \vdots \\ K_{\text{estimator}_{\Psi_{s_k c}}} \end{bmatrix},$$

$$\boldsymbol{\Theta}_{\Psi} = \begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 & K_{\text{estimator}_{\Psi_{s_1 s_1}}} & \cdots & K_{\text{estimator}_{\Psi_{s_1 s_k}}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K_{\text{estimator}_{\Psi_{s_k s_1}}} & \cdots & K_{\text{estimator}_{\Psi_{s_k s_k}}} \end{bmatrix}^{-1} \tag{47}$$

The margin values of non-support samples can be derived in terms of $\Delta\alpha_{\Psi c}$ as follows:

$$\begin{bmatrix} \Delta h_{\text{closed-loop}}(\left[\boldsymbol{\Pi}_{\Psi z_1}\right]) \\ \Delta h_{\text{closed-loop}}(\left[\boldsymbol{\Pi}_{\Psi z_2}\right]) \\ \vdots \\ \Delta h_{\text{closed-loop}}(\left[\boldsymbol{\Pi}_{\Psi z_r}\right]) \end{bmatrix} = \boldsymbol{\gamma}_{\Psi} \Delta\alpha_{\Psi c},$$

$$\boldsymbol{\gamma}_{\Psi} = \begin{bmatrix} K_{\text{estimator}_{\Psi_{z_1 c}}} \\ K_{\text{estimator}_{\Psi_{z_2 c}}} \\ \vdots \\ K_{\text{estimator}_{\Psi_{z_r c}}} \end{bmatrix} + \begin{bmatrix} 1 & K_{\text{estimator}_{\Psi_{z_1 s_1}}} & \cdots & K_{\text{estimator}_{\Psi_{z_1 s_l}}} \\ 1 & K_{\text{estimator}_{\Psi_{z_2 s_1}}} & \cdots & K_{\text{estimator}_{\Psi_{z_2 s_l}}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K_{\text{estimator}_{\Psi_{z_r s_1}}} & \cdots & K_{\text{estimator}_{\Psi_{z_r s_l}}} \end{bmatrix} \boldsymbol{\beta}_{\Psi} \tag{48}$$

where $\boldsymbol{\Pi}_{\Psi z_r} = \left[\boldsymbol{\Pi}_{f z_r}, \boldsymbol{\Pi}_{\theta z_r}, \boldsymbol{\Pi}_{\mu z_r}, \boldsymbol{\Pi}_{\rho z_r}\right], z_1, z_2, \ldots, z_r$ are the indices of non-support samples, $\boldsymbol{\gamma}_{\Psi}$ are margin sensitivities (Uçak and Günel 2016, 2017a).

4 Simulation results

The control performance evaluation of the proposed SMC is carried out on a nonlinear CSTR system. CSTR is a widely utilized chemical reactor system in industry, mainly used to produce polymers, pharmaceuticals and other various chemical products (Uçak and Günel 2016, 2017b, c). The schematic diagram of the CSTR system is illustrated in Fig. 8 (Uçak and Günel 2017b, c; Kravaris and Palanki 1988). In CSTR system, isothermal, liquid-phase, successive multicomponent chemical reactions can be performed (Kravaris and Palanki 1988; Wu and Chou 1999). Let us consider that a chemical reaction given as follows is carried out in CSTR:

$$A \leftrightharpoons B \rightarrow C \tag{49}$$

where A, B are the inlet reactants mixed in a vessel with constant volume via an agitator and transume to the product C (Uçak and Günel 2016, 2017b, c; Wu and Chou 1999; Uçak and Günel 2019). The reaction in (49) is composed of two sides: first is among A–B, and second one is between B and C (Uçak and Günel 2016, 2017b, c, 2019). Therefore, in CSTR, the aim is to control the concentration of product C by adjusting the molar feed rate of reactant B (Uçak and Günel 2016, 2017b, c; Kravaris and Palanki 1988; Wu and Chou 1999; Uçak and Günel 2019). The differential

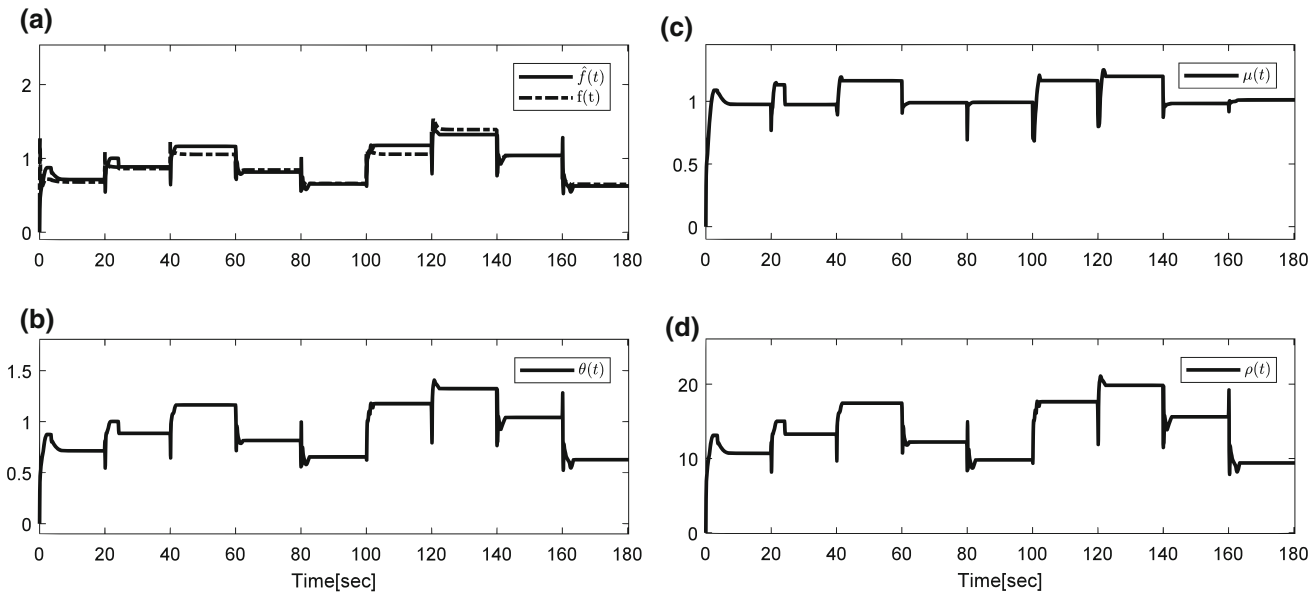

Fig. 10 Adaptive sliding mode controller parameters for staircase input (without noise and parametric uncertainty case)

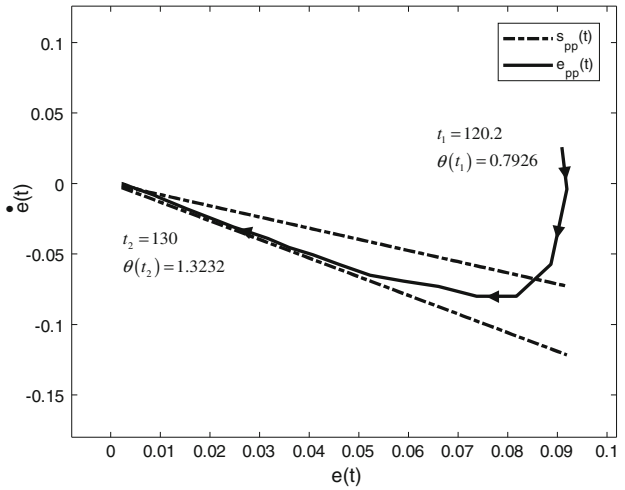

Fig. 11 The phase plane of the error dynamics and $s_{\text{pp}}(t)$ sliding function between 120.2 and 130 s (*pp* phase plane)

equations describing the dynamical behaviour of the system, proposed by Kravaris and Palanki (1988), are expressed as follows:

$$\begin{aligned}
\dot{x}_1(t) &= 1 - x_1(t) - Da_1 x_1(t) + Da_2 x_2^2(t) \\
\dot{x}_2(t) &= -x_2(t) + Da_1 x_1(t) - Da_2 x_2^2(t) \\
&\quad - Da_3 d_2(t) x_2^2(t) + u(t) \\
\dot{x}_3(t) &= -x_3(t) + Da_3 d_2(t) x_2^2(t)
\end{aligned} \tag{50}$$

where $x_1(t), x_2(t)$ and $x_3(t)$ are states obtained from the concentrations of reactant A, middle reactant B and product C, respectively, $Da_1 = 3$, $Da_2 = 0.5$, $Da_3 = 1$, $u(t)$ is the control signal, $x_3(t)$ is the controlled output of the system, $d_2(t)$ is the time-varying parameter of the system which represents the activity of the reaction, the nominal value of which is $d_{2_{\text{nominal}}}(t) = 1$ as given in Uçak and Günel (2016, 2017b, c, 2019), Iplikci (2006, 2010), Kravaris and Palanki (1988) and Wu and Chou (1999). The limitation for control signal is given as $u_{\min} = 0$ and $u_{\max} = 1$, and duration of control signal is set as $\tau_{\min} = \tau_{\max} = T_s = 0.1$ seconds where T_s is sampling time. The performance of the system has been evaluated for three different cases: (1) nominal case: when there is no noise and parametric uncertainty in the system. (2) Measurement noise case: 30 dB Gaussian measurement noise is added to the output of the system. (3) Parametric uncertainty: time-varying parameter is introduced to the system. $\mathbf{M}_c = \left[u_{n-1} \ldots u_{n-n_u}, y_n \ldots y_{n-n_y}\right]^{\text{T}}$ is utilized as the input feature vector for $\text{SVR}_{\text{model}}$ where $n_u = n_y = 2$ (Uçak and Günel 2017b, c). For $\text{SVR}_{\text{estimator}}$, since the control performance is closely associated with the chosen features, it may be essential to utilize various input feature vectors depending on the particulars of the system, whether there is noise, disturbance, parametric uncertainty or not, etc. The input feature vectors for $\text{SVR}_{\text{estimator}}$ are chosen as $\mathbf{\Pi}_{fn} = \mathbf{\Pi}_{\theta n} = \mathbf{\Pi}_{\rho n} = [I_n\ I_{n-1} \ldots I_{n-n_i},\ u_{n-1} \ldots u_{i-n_{u_u}},\ y_n \ldots y_{n-n_y}]^{\text{T}}$, $\mathbf{\Pi}_{\mu n} = [e_{1_n} \ldots e_{1_{n-n_{e_1}}},\ e_{2_n} \ldots e_{2_{n-n_{e_2}}},\ s_n \ldots s_{n-n_s}]^{\text{T}}$ where $I_n = e_n$, $e_n = e_{1_n} = r_n - y_n$, $e_{2_n} = \dot{e1}_n = \dot{r}_n - \dot{y}_n$ and $s_n = \theta_n e_{1_n} + e_{2_n}$. The number of the previous features is assigned as $n_i = 10$, $n_u = 1$, $n_y = n_{e_1} = n_{e_2} = n_s = 0$ for staircase reference signals in nominal and measurement noise cases. For all remaining cases and signals, $n_i = 5$, $n_u = 1$, $n_y = n_{e_1} = n_{e_2} = n_s = 0$ is deployed.

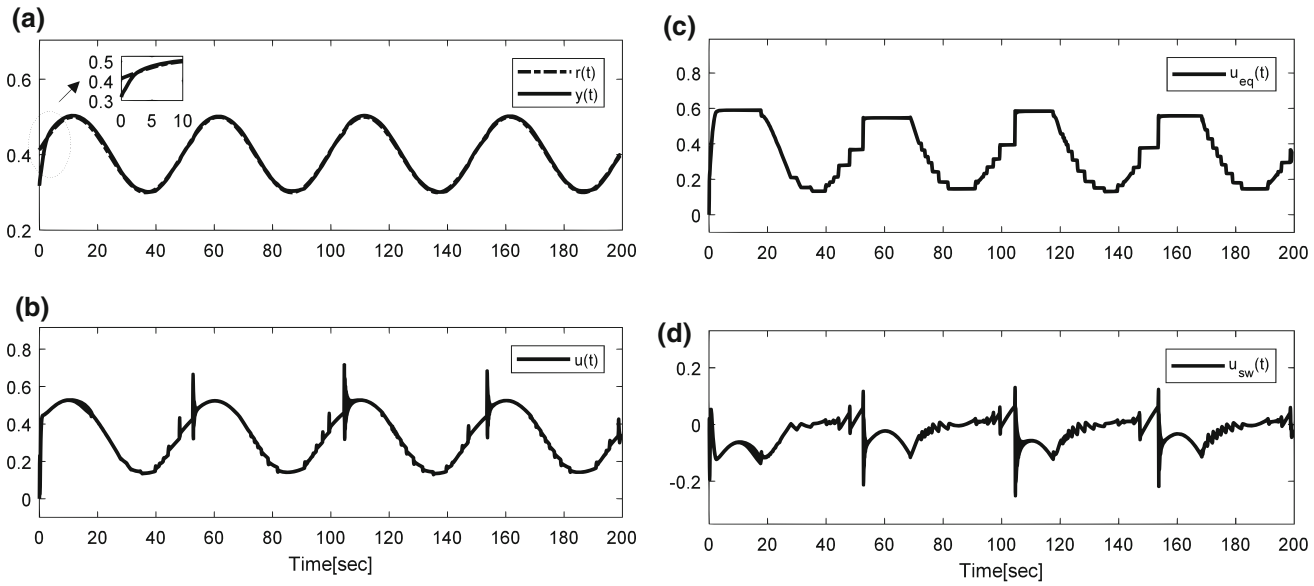

Fig. 12 System output (**a**), control signal (**b**), equivalent control (**c**) and switching control (**d**) for sinusoidal input (without noise and parametric uncertainty case)

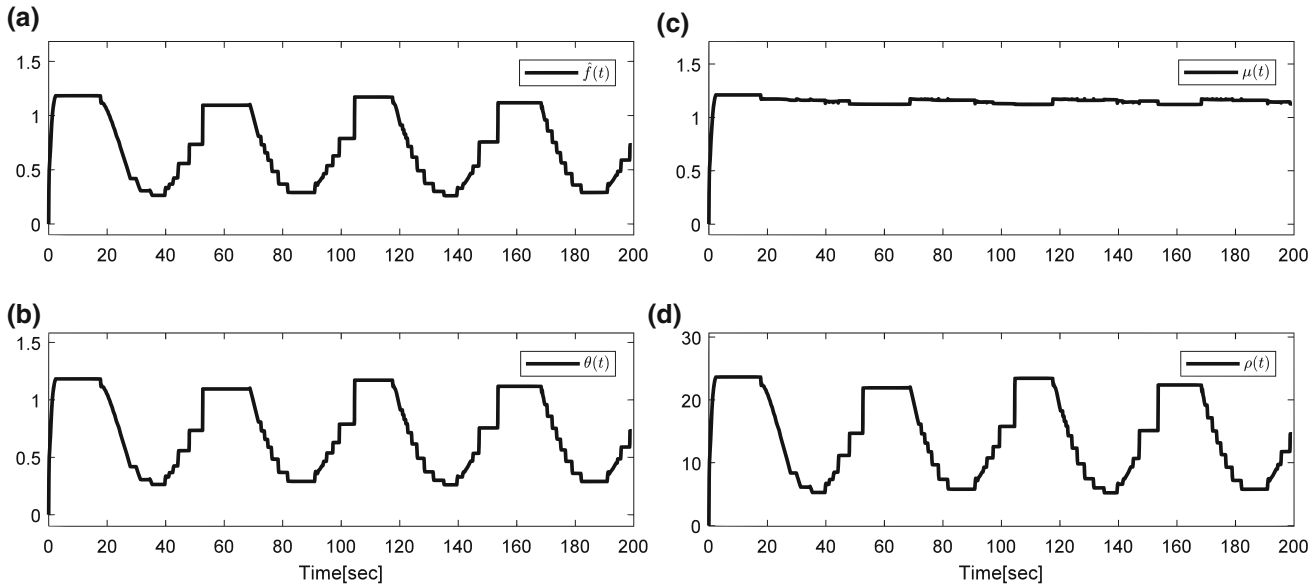

Fig. 13 Adaptive sliding mode controller parameters for sinusoidal input (without noise and parametric uncertainty case)

4.1 Nominal case with no noise and parametric uncertainty

The tracking performance of the closed-loop system in response to staircase reference signal and control signal produced by SMC are given in Fig. 9. As can be seen from tracking performance given Fig. 9a, the controlled system successfully tracks the desired signal. In Fig. 9, equivalent and switching control signals are depicted in Fig. 9c, d. The approximation of SMC parameters via $\text{SVR}_{\text{estimator}}$ is illustrated in Fig. 10. The estimation of the system dynamics($\hat{f}(t)$) via $\text{SVR}_{\text{model}}$ and its actual value($f(t)$) obtained via differential equations in (50) are given in Fig. 10a. It is expected that $|\hat{f} - f| \leq F$ since f is assumed to be bounded by some known function $F = F(x, \dot{x})$ (Slotine and Li 1991). The illustration of the system parameter convergence to the desired sliding function (phase plane of the errors) between 120.2 and 130 s is shown in Fig. 11 where $s_{\text{pp}}(t)$ and $e_{\text{pp}}(t)$ denote the sliding function and tracking error phase plane. In adaptive schemes, exact approximation of uncertainties is a major issue; hence, the SVR-based methodology proposed here can be combined with the meth-

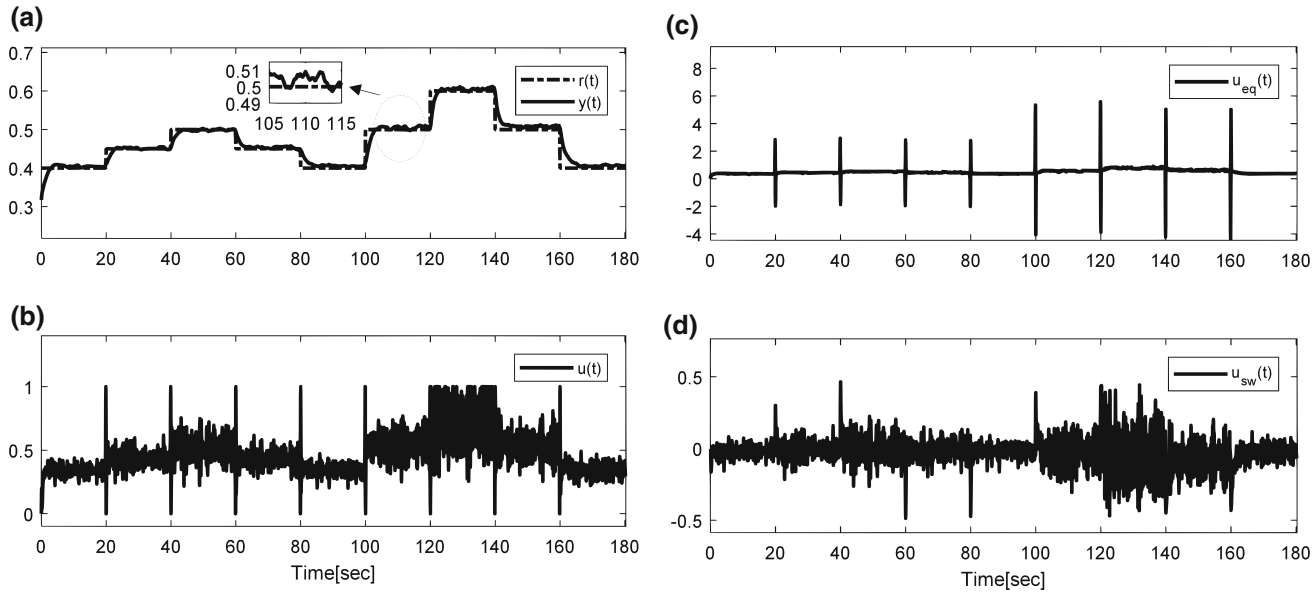

Fig. 14 System output (**a**), control signal (**b**), equivalent control (**c**) and switching control (**d**) for staircase input (measurement noise case)

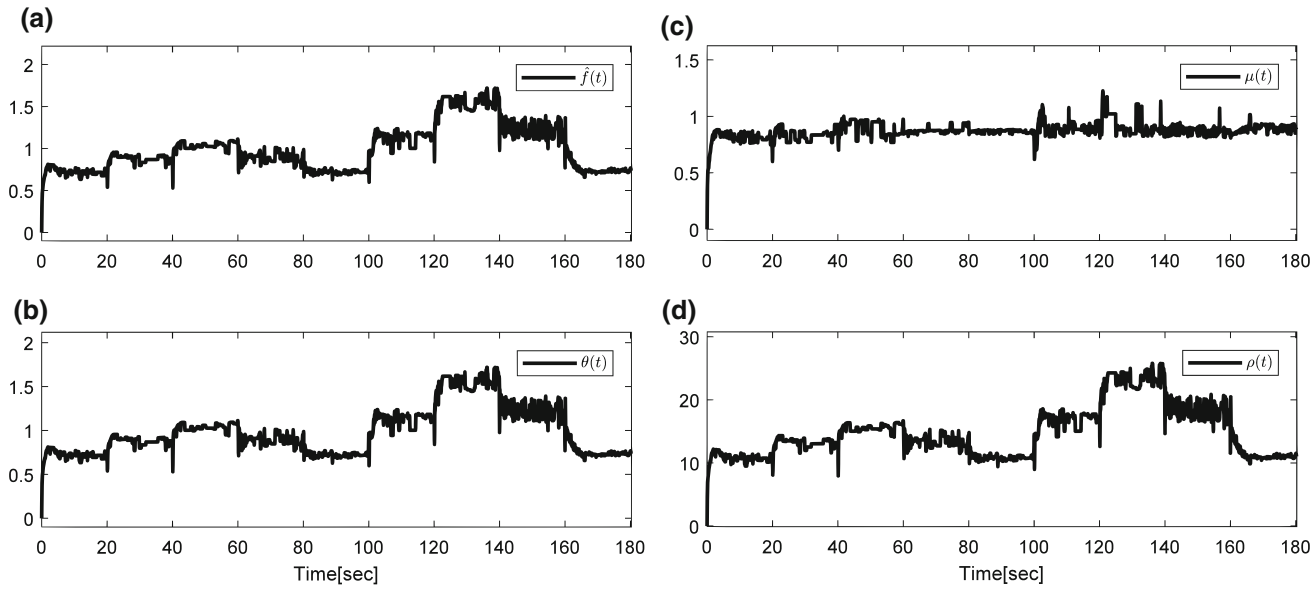

Fig. 15 Adaptive sliding mode controller parameters for staircase input (measurement noise case)

ods suggested in Pan and Yu (2016) and Pan et al. (2017) in future works to assure parametric convergence. The tracking performance of the controller and SMC parameters for sinusoidal input signal is shown in Figs. 12 and 13, respectively.

4.2 Measurement noise

Since control systems are frequently exposed to measurement noise generated by the measurement mechanism, the performance evaluation of the controller with respect to noise is crucial to design robust controllers. For this purpose, the performance evaluation of the system is performed under 30 dB Gaussian measurement noise. The control performance is illustrated in Fig. 14. The alternations of the SMC parameters are given in Fig. 15. The response of the closed-loop system and SMC parameters for sinusoidal input is depictured in Figs. 16 and 17, respectively. As can be seen from Figs. 14 and 16, the system output can accurately track the applied reference input signals.

4.3 Parametric uncertainty

The second important criterion for adaptive controllers is the evaluation of the controller robustness in terms of parametric

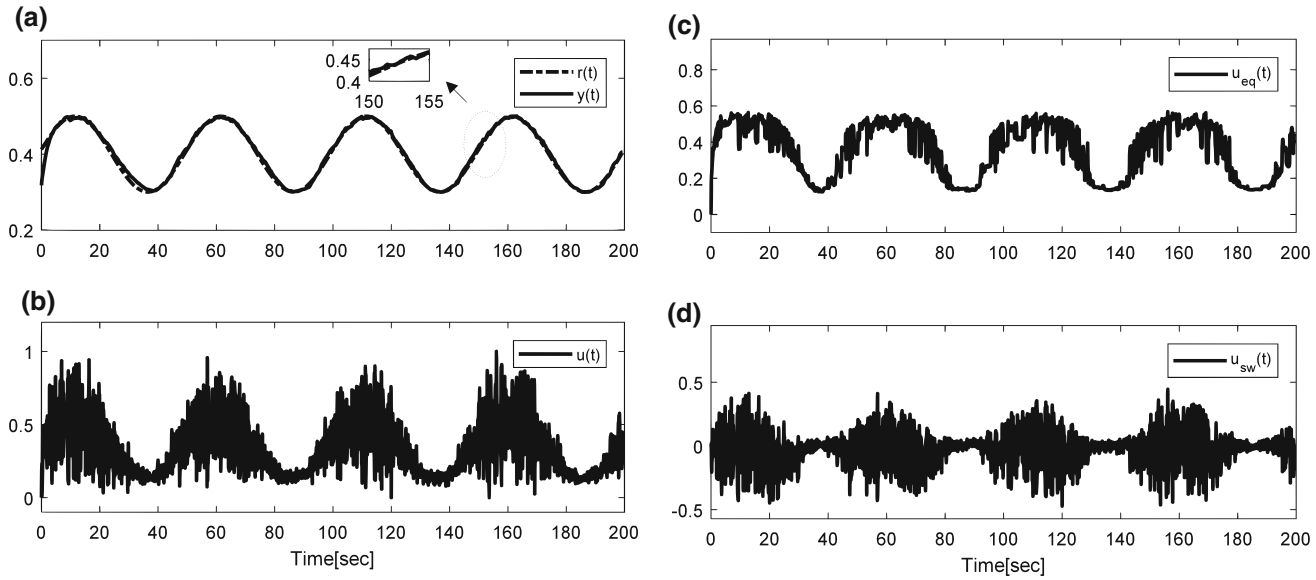

Fig. 16 System output (**a**), control signal (**b**), equivalent control (**c**) and switching control (**d**) for sinusoidal input (measurement noise case)

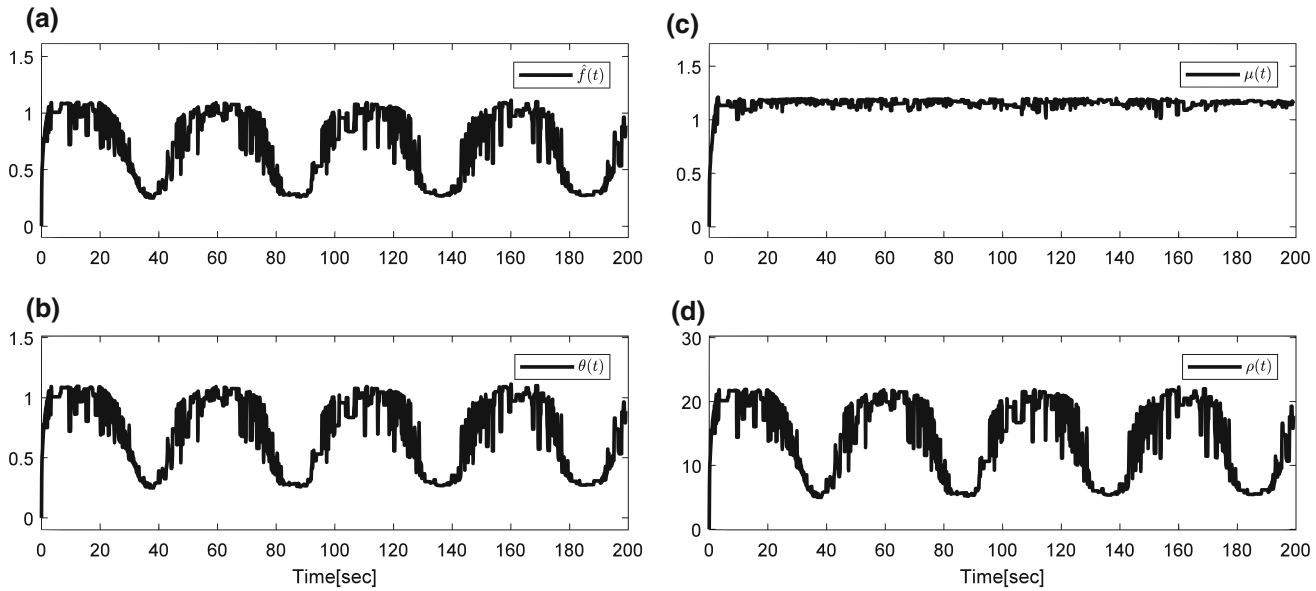

Fig. 17 Adaptive sliding mode controller parameters for sinusoidal input (measurement noise case)

uncertainty. In our simulations, the parameter which represents the activity of the reaction ($d_2(t)$) is considered as the time-varying parameter of the system. It is allowed to vary slowly in the purlieu of its nominal value ($d_{2_{\text{nom}}}(t) = 1$) as $d_2(t) = 1 + 0.1\sin(0.2\pi t)$ (Uçak and Günel 2016, 2017b, c, 2019). The tracking performance of the controller and control signal applied to the system for parametric uncertainty case are depictured in Fig. 18. The convergence of SMC parameters to their optimal values is illustrated in Fig. 19. If the control signal produced for nominal system parameters in Fig. 9 and for the time-varying parameter situation in Fig. 18 is compared, it can be clearly observed how the control signal in Fig. 18 tries to reject the uncertainty resulting from the time-varying system parameter. The response of the closed-loop system and alternation of SMC parameters for sinusoidal input are shown in Figs. 20 and 21, respectively.

5 Conclusion

In this paper, a novel SMC architecture based on SVR is proposed for nonlinear dynamical systems. The closed-loop margin notion proposed in Uçak and Günel (2016, 2017a) has been expanded for SMC. The adjustment mechanism is composed of two main SVR structures: $\text{SVR}_{\text{model}}$ identifies the dynamics of the controlled system, and $\text{SVR}_{\text{estimator}}$

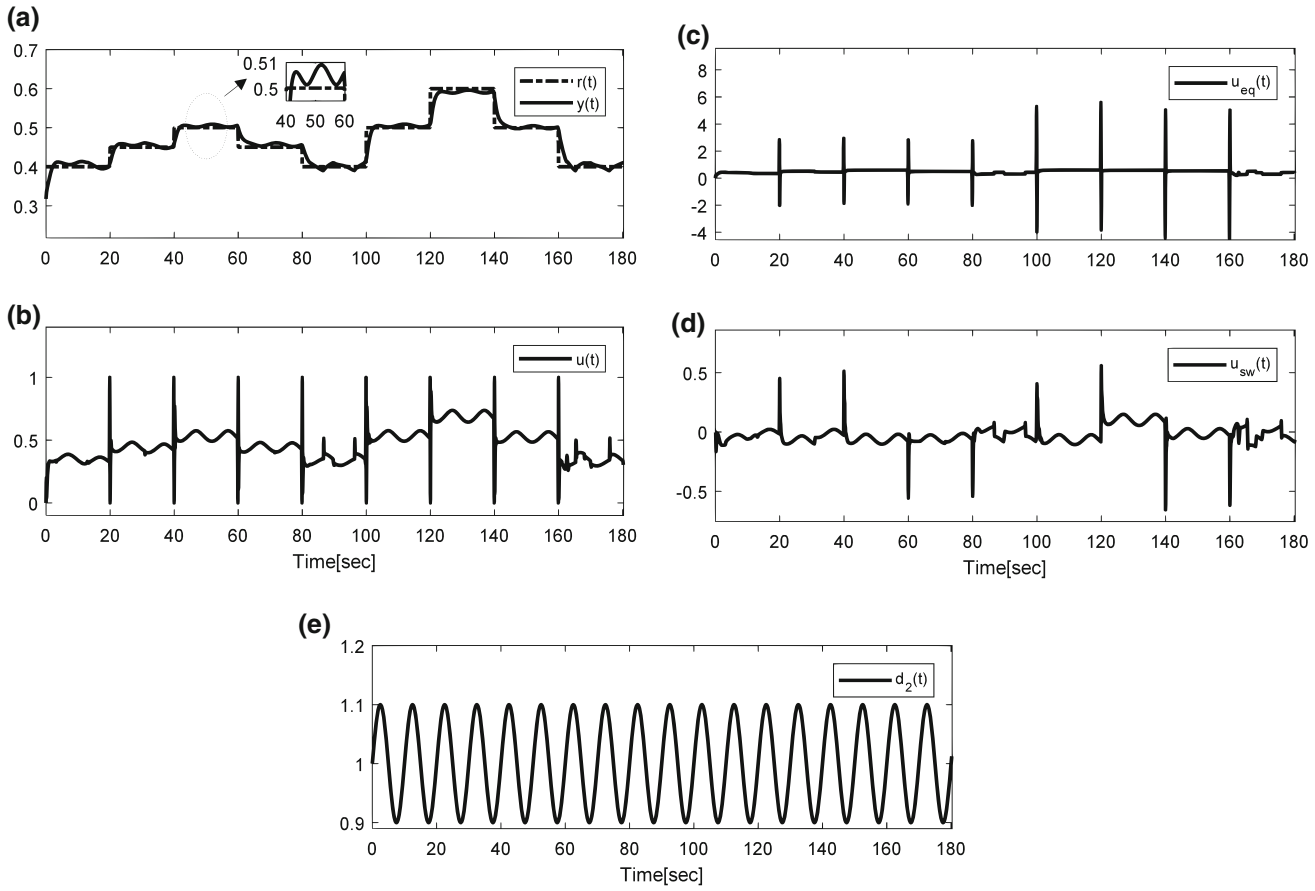

Fig. 18 System output (**a**), control signal (**b**), equivalent control (**c**), switching control (**d**) and time-varying system parameter (**e**) for staircase input (parametric uncertainty case)

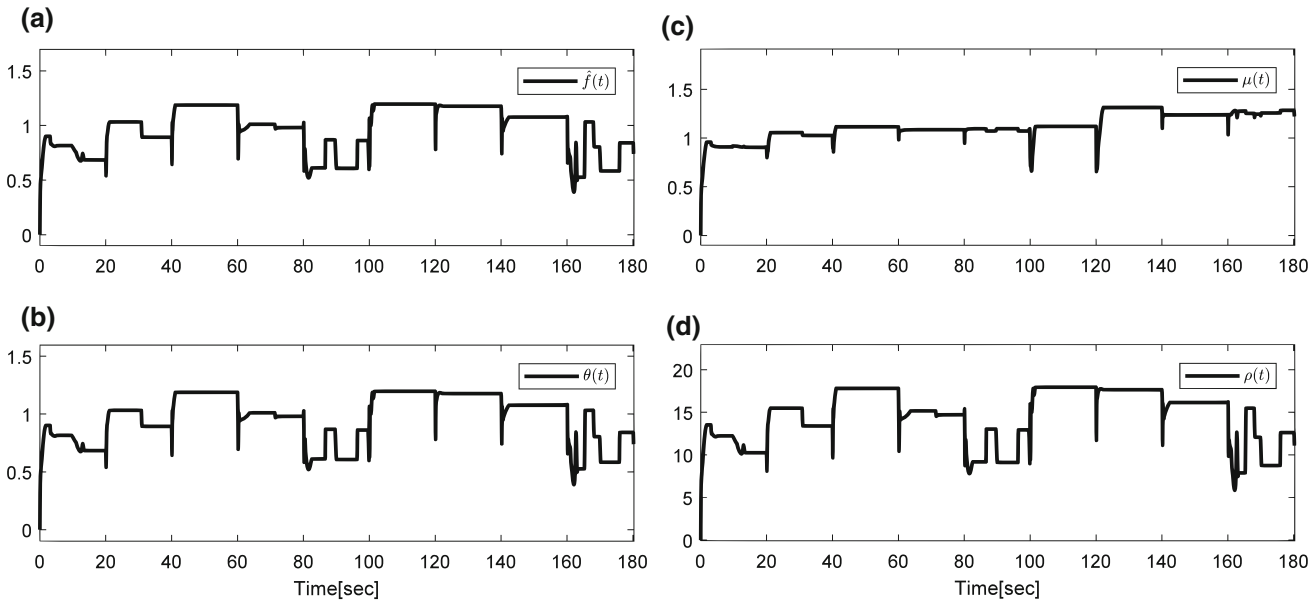

Fig. 19 Adaptive sliding mode controller parameters for staircase input (parametric uncertainty case)

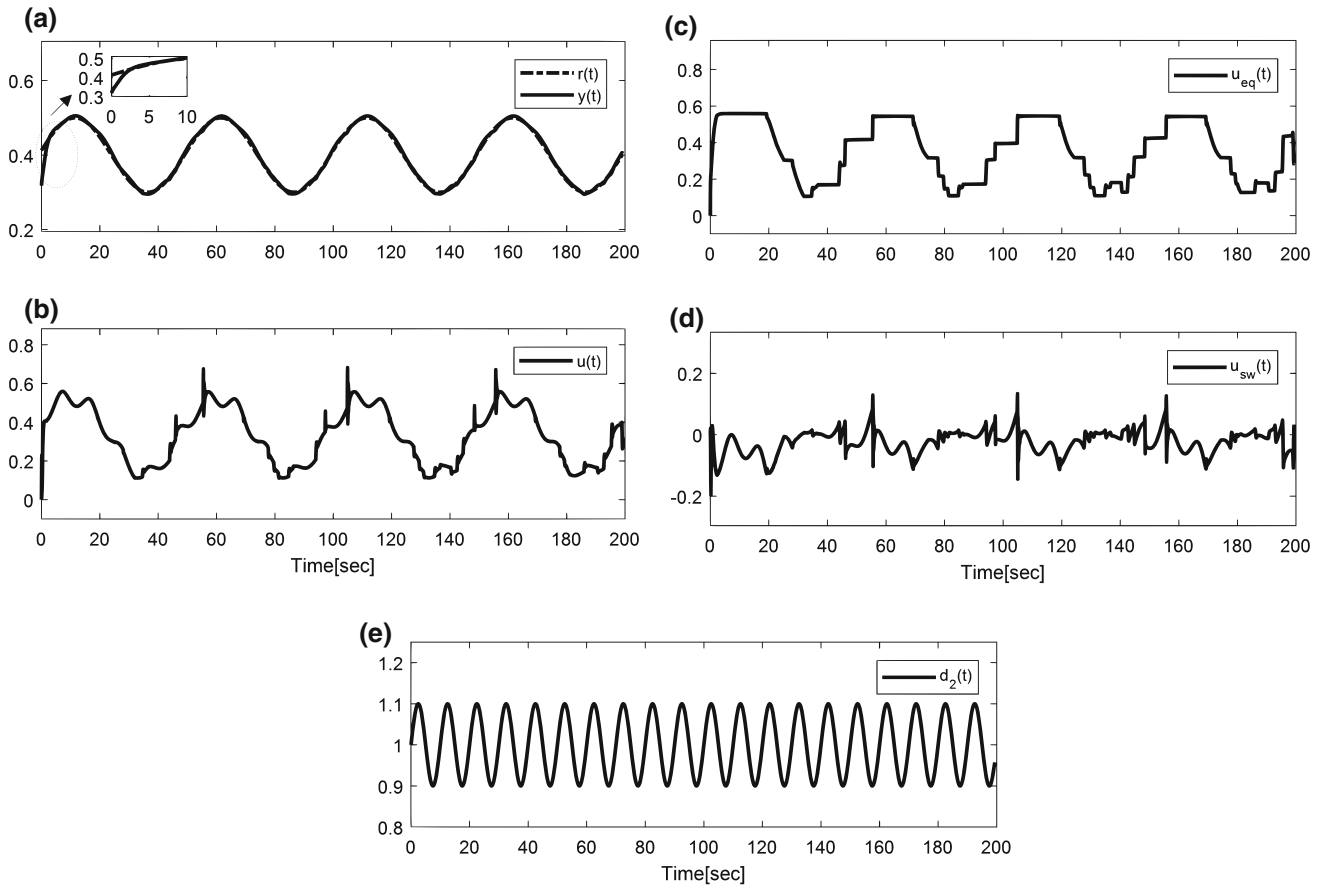

Fig. 20 System output (**a**), control signal (**b**), equivalent control (**c**), switching control (**d**) and time-varying system parameter (**e**) for sinusoidal input (parametric uncertainty case)

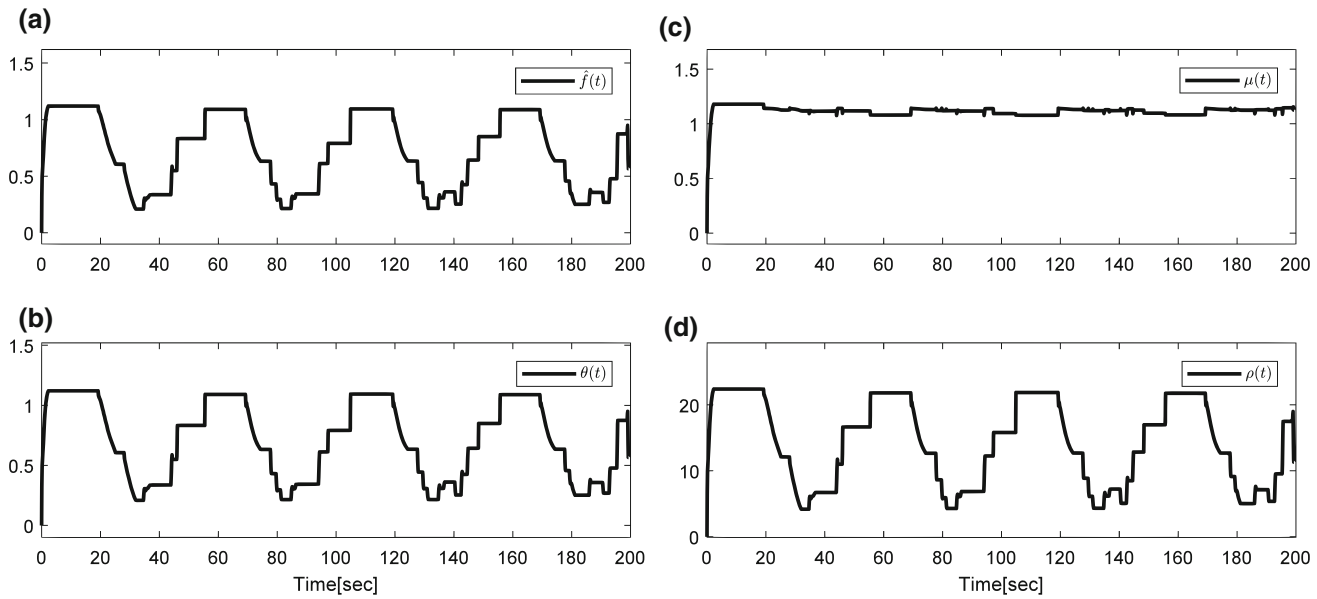

Fig. 21 Adaptive sliding mode controller parameters for sinusoidal input (parametric uncertainty case)

approximates the parameters of SMC. The main contribution of the paper is that SVR is directly deployed to identify the parameters of SMC as opposed to existing works in technical literature where SVRs are generally utilized for modelling to approximate system Jacobians to adjust parameters of conventional controller structures.

The performance of the SMC is examined on a nonlinear continuously stirred tank reactor (CSTR) benchmark system. The robustness of the SMC has been evaluated for the noiseless case and when measurement noise and parametric uncertainty are added. Simulations results prove that proposed control architecture attains successful tracking performance, good noise rejection and high toleration to parametric uncertainties. In future works, it is planned to develop new adaptive control mechanisms for nonlinear systems based on SVR by employing closed-loop margin notions.

Compliance with ethical standards

Conflict of interest The author declares that there is no conflict of interest regarding the publication of this paper.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

Al-Duwaish HN, Al-Hamouz ZM (2011) A neural network based adaptive sliding mode controller: application to a power system stabilizer. Energy Convers Manag 52(2):1533–1538. https://doi.org/10.1016/j.enconman.2010.06.060

Al-Holou N, Lahdhiri T, Joo DS, Weaver J, Al-Abbas F (2002) Sliding mode neural network inference fuzzy logic control for active suspension systems. IEEE Trans Fuzzy Syst 10(2):234–246. https://doi.org/10.1109/91.995124

Bandyopadhyay B, Deepak F, Kim KS (2009) Sliding mode control using novel sliding surfaces. Springer, Berlin

Baric M, Petrovic I, Peric N (2005) Neural network-based sliding mode control of electronic throttle. Eng Appl Artif Intell 18(8):951–961. https://doi.org/10.1016/j.engappai.2005.03.008

Bartoszewicz A (1996) Time-varying sliding modes for second-order systems. IEE Proc Control Theory Appl 143(5):455–462. https://doi.org/10.1049/ip-cta:19960535

Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297

de la Parte MP, Camacho O, Camacho EF (2002) Development of a GPC-based sliding mode controller. ISA Trans 41(1):19–30. https://doi.org/10.1016/S0019-0578(07)60199-2

Derdiyok A, Levent M (2000) Sliding mode control of a bioreactor. Korean J Chem Eng 17(6):619–624. https://doi.org/10.1007/BF02699106

Drucker H, Burges CJC, Kaufman L, Smola A, Vapnik V (1997) Support vector regression machines. In: 10th annual conference on neural information processing systems (NIPS), Denver

Efe MÖ, Kaynak O, Yu XH, Wilamowski BM (2001) Sliding mode control of nonlinear systems using Gaussian radial basis function neural networks. In: International joint conference on neural networks (IJCNN 01), Washington

Emel'yanov SV (1967) Variable structure control systems. Nauka, Moscow

Emel'yanov SV (2007) Theory of variable-structure control systems: inception and initial development. Comput Math Model 18(4):321–331

Ertugrul M, Kaynak O (2000) Neuro sliding mode control of robotic manipulators. Mechatronics 10(1–2):239–263. https://doi.org/10.1016/S0957-4158(99)00057-4

Ertuğrul M, Kaynak O, Şabanoviç A (1995) A comparison of various VSS techniques on the control of automated guided vehicles. In: 1995 IEEE international symposium on industrial electronics (ISIE 95), Athens

Fei JT, Ding HF (2012) Adaptive sliding mode control of dynamic system using RBF neural network. Nonlinear Dyn 70(2):1563–1573. https://doi.org/10.1007/s11071-012-0556-2

Feng Y, Han FL, Yu XH (2014) Chattering free full-order sliding-mode control. Automatica 50(4):1310–1314. https://doi.org/10.1016/j.automatica.2014.01.004

Guo HJ, Lin SF, Liu JH (2006) A radial basis function sliding mode controller for chaotic Lorenz system. Phys Lett A 351(4–5):257–261. https://doi.org/10.1016/j.physleta.2005.10.101

Hua J, An LX, Li YM (2015) Bionic fuzzy sliding mode control and robustness analysis. Appl Math Model 39(15):4482–4493. https://doi.org/10.1016/j.apm.2014.12.017

Hung LC, Chung HY (2007) Decoupled sliding-mode with fuzzy-neural network controller for nonlinear systems. Int J Approx Reason 46(1):74–97. https://doi.org/10.1016/j.ijar.2006.08.002

Hušek P (2016) Adaptive sliding mode control with moving sliding surface. Appl Soft Comput 42:178–183. https://doi.org/10.1016/j.asoc.2016.01.009

Iplikci S (2006) Online trained support vector machines-based generalized predictive control of non-linear systems. Int J Adapt Control Signal Process 20(10):599–621. https://doi.org/10.1002/acs.919

Iplikci S (2010) A comparative study on a novel model-based PID tuning and control mechanism for nonlinear systems. Int J Robust Nonlinear Control 20(13):1483–1501. https://doi.org/10.1002/rnc.1524

Kaynak O, Erbatur K, Ertugrul M (2001) The fusion of computationally intelligent methodologies and sliding-mode control-a survey. IEEE Trans Ind Electron 48(1):4–17. https://doi.org/10.1109/41.904539

Kim SW, Lee JJ (1995) Design of a fuzzy controller with fuzzy sliding surface. Fuzzy Sets Syst 71(3):359–367. https://doi.org/10.1016/0165-0114(94)00276-D

Kravaris C, Palanki S (1988) Robust nonlinear state feedback under structured uncertainty. AIChe J 34(7):1119–1127. https://doi.org/10.1002/aic.690340708

Li LS, Li JN (2008a) Chattering-free support vector regression sliding mode control. In: International conference on computational-intelligence and security, Suzhou

Li LS, Li JN (2008b) Chattering-free sliding mode control based on support vector regression. In: 3rd international conference on intelligent system and knowledge engineering, Xiamen

Li JN, Zhang YB, Pan HP (2008a) Chattering-free LS-SVM sliding mode control. In: 5th international symposium on neural networks, Beijing

Li JN, Zhang YB, Pan HP (2008b) Chattering-free LS-SVM sliding mode control for a class of uncertain discrete time-delay systems with input saturation. In: international conference on intelligent computation technology and automation, Changsha

Lin FJ, Shen PH (2006) Robust fuzzy neural network sliding-mode control for two-axis motion control system. IEEE Trans Ind Electron 53(4):1209–1225. https://doi.org/10.1109/TIE.2006.878312

Lin FJ, Shyu KK, Wai RJ (2001) Recurrent-fuzzy-neural-network sliding-mode controlled motor-toggle servomechanism. IEEE-ASME Trans Mechatron 6(4):453–466. https://doi.org/10.1109/3516.974859

Liu J, Wang X (2012) Advanced sliding mode control for mechanical systems. Springer, Berlin

Ma J, Theiler J, Perkins S (2003) Accurate online support vector regression. Neural Comput 15(11):2683–2703. https://doi.org/10.1162/089976603322385117

Martin M (2002) On-line support vector machine regression. In: 13th European conference on machine learning (ECML 2002), Helsinki

Ngo QH, Nguyen NP, Nguyen CN, Tran TH, Ha QP (2017) Fuzzy sliding mode control of an offshore container crane. Ocean Eng 140:125–134. https://doi.org/10.1016/j.oceaneng.2017.05.019

Pan Y, Yu H (2016) Composite learning from adaptive dynamic surface control. IEEE Trans Autom Control 61(9):2603–2609. https://doi.org/10.1109/TAC.2015.2495232

Pan Y, Sun T, Liu Y, Yu H (2017) Composite learning from adaptive backstepping neural network control. Neural Netw 95(2017):134–142. https://doi.org/10.1016/j.neunet.2017.08.005

Pan YP, Yang CG, Pan L, Yu HY (2018) Integral sliding mode control: performance, modification, and improvement. IEEE Trans Ind Inform 14(7):3087–3096. https://doi.org/10.1109/TII.2017.2761389

Roopaei M, Jahromi MZ (2009) Chattering-free fuzzy sliding mode control in MIMO uncertain systems. Nonlinear Anal Theory Methods Appl 71(10):4430–4437. https://doi.org/10.1016/j.na.2009.02.132

Sabanovic A (2011) Variable structure systems with sliding modes in motion control-a survey. IEEE Trans Ind Inform 7(2):212–223. https://doi.org/10.1109/TII.2011.2123907

Slotine JJE, Li W (1991) Applied nonlinear control. Prentice Hall, Upper Saddle River

Smola AJ, Schölkopf B (2004) A tutorial on support vector regression. Stat Comput 14(3):199–222. https://doi.org/10.1023/B:STCO.0000035301.49549.88

Sun TR, Pei HL, Pan YP, Zhou HB, Zhang CH (2011) Neural network-based sliding mode adaptive control for robot manipulators. Neurocomputing 74(14–15):2377–2384. https://doi.org/10.1016/j.neucom.2011.03.015

Tokat S (2006) Time-varying sliding surface design with support vector machine based initial condition adaptation. J Vib Control 12(8):901–926. https://doi.org/10.1177/1077546306067675

Tokat S, Eksin İ, Güzelkaya M, Söylemez MT (2003) Design of a sliding mode controller with a nonlinear time-varying sliding surface. Trans Inst Meas Control 25(2):145–162. https://doi.org/10.1191/0142331203tm079oa

Tokat S, Iplikci S, Ulusoy L (2009a) Performance based sliding mode controller using support vector machines. In: 2nd IFAC conference on intelligent control systems and signal processing

Tokat S, Iplikci S, Ulusoy L (2009b) Output feedback sliding mode control with support vector machine based observer gain adaptation. In: 8th WSEAS international conference on circuits, systems, electronics, control and signal processing, Puerto de la Cruz

Tsai CH, Chung HY, Yu FM (2004) Neuro-sliding mode control with its applications to seesaw systems. IEEE Trans Neural Netw 15(1):124–134. https://doi.org/10.1109/TNN.2003.811560

Uçak K, Günel GO (2016) An adaptive support vector regressor controller for nonlinear systems. Soft Comput 20(7):2531–2556. https://doi.org/10.1007/s00500-015-1654-0

Uçak K, Günel GO (2017a) Generalized self-tuning regulator based on online support vector regression. Neural Comput Appl 28:S775–S801. https://doi.org/10.1007/s00521-016-2387-4

Uçak K, Günel GO (2017b) PID type STR based on SVR for nonlinear systems. In: International conference on advanced technology and sciences, Istanbul

Uçak K, Günel GO (2017c) Fuzzy PID type STR based on SVR for nonlinear systems. In: 10th international conference on electrical and electronics engineering, Bursa

Uçak K, Günel GO (2019) Model free adaptive support vector regressor controller for nonlinear systems. Eng Appl Artif Intell 81:47–67. https://doi.org/10.1016/j.engappai.2019.02.001

Utkin VI (1992) Sliding modes in control and optimization. Springer, Berlin

Utkin V (1977) Variable structure systems with sliding modes. IEEE Trans Autom Control 22(2):212–222. https://doi.org/10.1109/TAC.1977.1101446

Vapnik V, Golowich SE, Smola A (1997) Support vector method for function approximation, regression estimation, and signal processing. In: 10th annual conference on neural information processing systems (NIPS), Denver

Wang X, Du Z, Chen J, Pan F (2009) Dynamic modeling of biotechnical process based on online support vector machine. J Comput 4(3):251–258. https://doi.org/10.4304/jcp.4.3.251-258

Wu W, Chou YS (1999) Adaptive feedforward and feedback control of non-linear time-varying uncertain systems. Int J Control 72(12):1127–1138. https://doi.org/10.1080/002071799220489

Yau HT, Chen CL (2006) Chattering-free fuzzy sliding-mode control strategy for uncertain chaotic systems. Chaos Solitons Fractals 30(3):709–718. https://doi.org/10.1016/j.chaos.2006.03.077

Zribi M, Oteafy A (2006) Control of a bioreactor using static and dynamic sliding mode controllers. In: 2006 IEEE GCC conference, GCC 2006, Manama