Chapter 2
Feedforward Control Based on Inverse Hysteresis Models

This chapter presents the rate-dependent hysteresis compensation of a piezoelectric nanopositioning stage using the feedforward control based on an inverse hysteresis model. Three different controllers are realized and compared, which employ Bouc–Wen model, modified Prandtl–Ishlinskii (MPI) model, and least squares support vector machines (LSSVM)-based intelligent model, respectively. Experimental studies demonstrate the superiority of LSSVM model in hysteresis modeling and compensation tasks.

2.1 Introduction

In order to compensate for the hysteresis effect, the hysteresis behavior is usually modeled by Preisach model [3, 6], Prandtl–Ishlinskii model [7], Bouc–Wen model [11, 12], Maxwell-based model [8], Dahl model [18], polynomial model [9], etc. Then, an inverse hysteresis model is constructed and utilized as a feedforward controller to cancel the hysteresis effect. However, the hysteresis effect is dependent not only on the amplitude but also on the frequency of input signals. It is very difficult to capture the complicated rate-dependent hysteretic behavior precisely. In addition, majority of the existing models employ a large number of parameters to describe the rate-dependent hysteresis [1, 21], which may block their applications in high-speed real-time control as an adverse effect.

Recently, it has been shown that ANN provides an efficient way to model the nonlinear hysteresis [5, 20]. Nevertheless, there is no universal method to determine an optimal ANN structure in terms of the number of hidden layers and number of neurons in each layer. Moreover, ANN exhibits the problems of overfitting and sinking into local optima, which are their major drawbacks in practical implementation. Alternatively, SVM gives a promising way to estimate nonlinear system models accurately. Based on the statistical learning theory and structural risk minimization principle, the SVM approach is capable of modeling nonlinear systems by trans-

Q. Xu and K.K. Tan, *Advanced Control of Piezoelectric Micro-/Nano-Positioning Systems*, Advances in Industrial Control,
DOI 10.1007/978-3-319-21623-2_2

forming the regression problem into a convex quadratic programming (QP) problem and then solving it with a QP solver. As compared with the conventional ANN, SVM exhibits the major advantages of global optimization and higher generalization capability. As a variant of SVM, LSSVM is introduced in 1999 [13]. LSSVM utilizes equality constraints instead of inequality constraints as in the ordinary SVM. Hence, it simplifies the regression to a problem that can be easily solved from a set of linear equations [17]. As a result, the LSSVM has a relatively low complexity and are more computationally efficient than the standard SVM [13]. Another advantage of LSSVM lies in that it has fewer parameters to tune [19]. For more details about LSSVM algorithm, the reader may refer to the book [14].

In this chapter, the LSSVM is introduced to the domain of hysteresis modeling and compensation for a nanopositioning stage, which is driven by piezoelectric stack actuators. A LSSVM model is proposed and trained by introducing the current input value and input variation rate as the input data set to formulate a one-to-one mapping. By adopting the radial basis function (RBF) as kernel function, the LSSVM model only has two free hyperparameters, which are optimally tuned by resorting to Bayesian inference framework. The effectiveness of the presented model is verified by carrying out experimental studies. For a comparison study, two state-of-the-art approaches, namely, Bouc–Wen model and MPI model, are employed. Both Bouc–Wen and MPI models are identified by resorting to a global search approach of particle swarm optimization (PSO).

In addition, a LSSVM inverse model-based feedforward control in combination with a proportional-integral-derivative (PID) feedback control is designed to compensate for the hysteresis nonlinearity. The feasibility of the presented modeling and control approaches are validated by a series of simulation and experimental studies.

2.2 System Description and Hysteresis Characterization

The test bed as employed in this work and its hysteresis properties are described in this section.

2.2.1 Experimental Setup

The experimental setup is shown in Fig. 2.1. A commercial two-axis nanopositioning stage (model: P-734.2CL, from Physik Instrumente Co., Ltd.) is adopted. It is driven by two piezoelectric stack actuators (PSAs) through a high-voltage amplifier (model: E-503.00, from Physik Instrumente Co., Ltd.). The integrated capacitive sensors are used for displacement measurement. The displacement sensors provide the accuracy and repeatability in nanometer and sub-nanometer levels, respectively. The output signals of the two capacitive sensors are processed by a signal conditioner module (model: E-509.C2A, from Physik Instrumente Co., Ltd.). In addition,

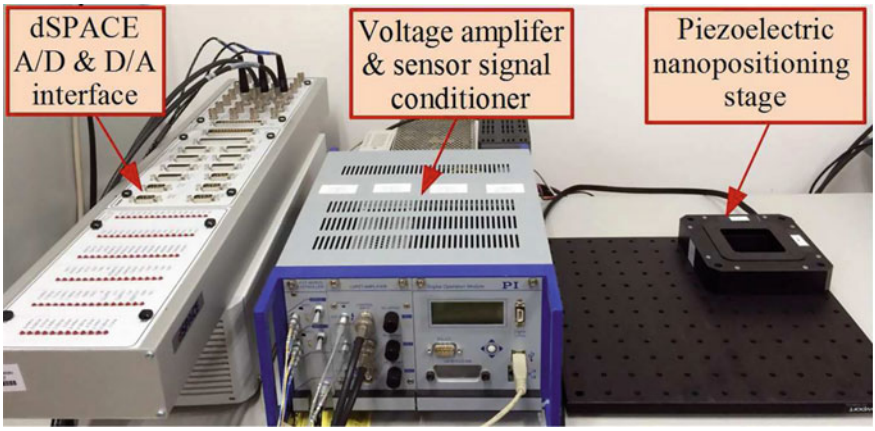

Fig. 2.1 Experimental setup of a piezoelectric nanopositioning system

the control hardware consists of a digital signal processor (DSP)-based real-time controller (model: DS1103, from dSPACE GmbH) with equipped 16-bit A/D and D/A converters. Control algorithms are developed with MATLAB® software and then downloaded to DS1103 board via the ControlDesk interface to implement the real-time control.

2.2.2 Hysteresis Characterization

By applying a triangular-wave voltage with the constant feed-rate and variable amplitudes to PSA #1, the open-loop position responses of the XY nanopositioning stage in the two working axes are shown in Fig. 2.2. It is seen that the cross-talk between the two axes is about 1.17 %, which indicates that the output motions in the two axes are approximately decoupled. Owing to the decoupling property of the two working axes, the motion control in one axis (x-axis) of the system is carried out in the following discussion.

The x-axis hysteresis curves in Fig. 2.2c reveal that the shape of the hysteresis loop is dependent on the amplitude of the input. The larger the amplitude of the input signal, the wider the output hysteresis loop. In addition, by applying sine waves with the same amplitude and varying frequencies (1–10 Hz) as shown in Fig. 2.3a, the x-axis displacement output is depicted in Fig. 2.3b. It can be observed from Fig. 2.3c that the hysteresis shape also relies on the input rate. The higher the frequency of the input signal, the larger the width of the output hysteresis.

The foregoing open-loop testing results indicate that the complicated hysteresis effects are dependent not only on the amplitude but also on the frequency of the input signal. The modeling and compensation of the hysteretic nonlinearity are carried out in the following sections.

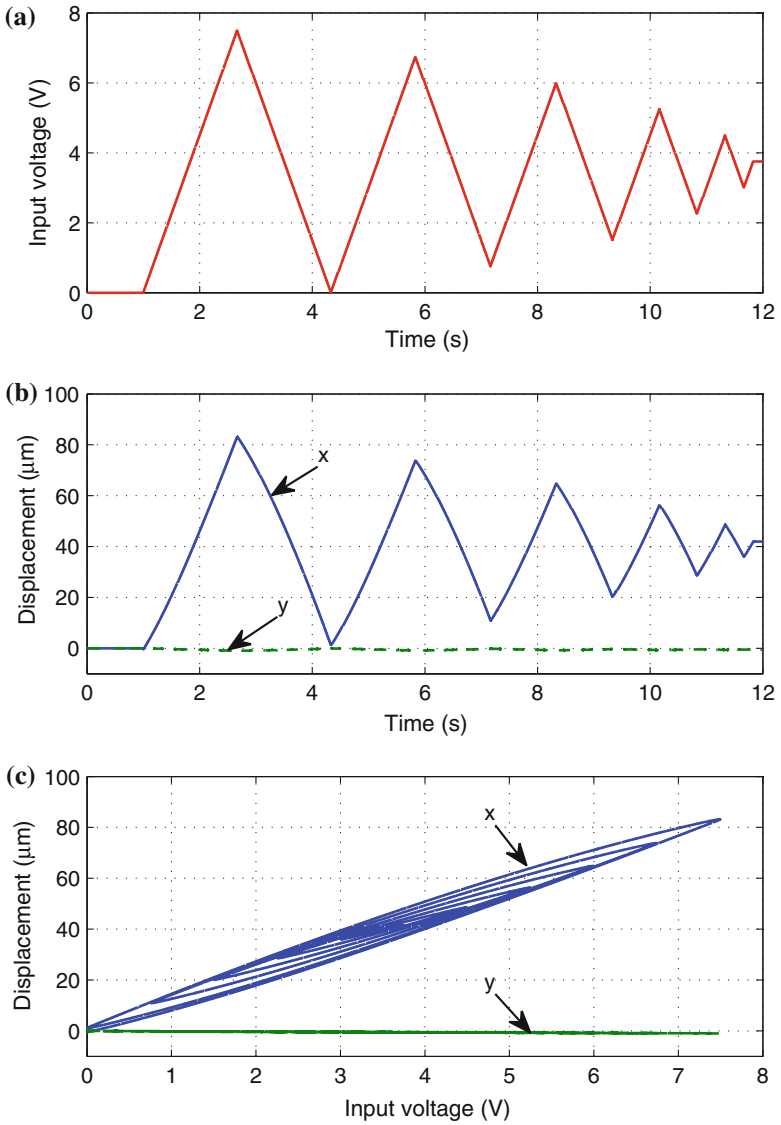

Fig. 2.2 Hysteresis curves of the XY nanopositioning stage with triangular-wave inputs. **a** Input voltage to the voltage amplifier. **b** Time history of the displacements x and y. **c** Output displacements x and y versus input voltage

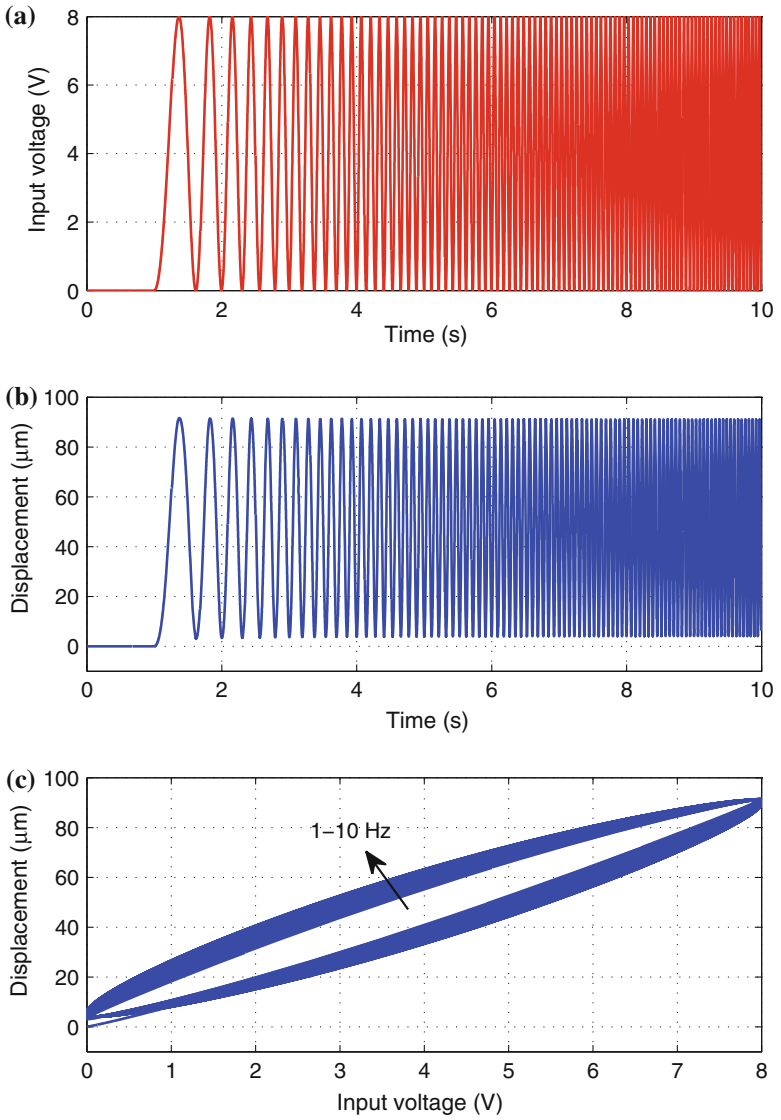

Fig. 2.3 Hysteresis curves of the XY nanopositioning stage with sinusoidal-wave inputs. **a** Input voltage to the voltage amplifier. **b** Time history of the displacement x. **c** Output displacement x versus input voltage

2.3 Hysteresis Modeling

The purpose of hysteresis modeling is to capture the hysteresis behavior of the system. First, the hysteresis is modeled using the popular Bouc–Wen, MPI, and LSSVM models. In addition, the corresponding inverse hysteresis models are also derived for the purpose of hysteresis compensation.

2.3.1 Hysteresis Modeling with the Bouc–Wen Model

Thanks to a fewer number of parameters, Bouc–Wen model has been widely employed in piezoelectric hysteresis modeling. In particular, the entire dynamics model of the piezoelectrically driven system with Bouc–Wen type of hysteresis can be established as follows [11, 12]:

$$m\ddot{y}(t) + b\dot{y}(t) + ky(t) = k[du(t) - h(t)] \tag{2.1}$$

$$\dot{h}(t) = \alpha d\dot{u}(t) - \beta|\dot{u}(t)|h(t)|h(t)|^{n-1} - \gamma\dot{u}(t)|h(t)|^n \tag{2.2}$$

where t is the time variable; parameters m, b, k, and y represent the mass, damping coefficient, stiffness, and displacement response of the piezostage, respectively; d is the piezoelectric coefficient; u denotes the input voltage; and h indicates the hysteretic loop in terms of displacement whose magnitude and shape are determined by parameters α, β, γ, and the order n, with n governing the smoothness of the transition from elastic to plastic response. For the elastic structure and material, $n = 1$ is assigned in Eq. (2.2).

To determine the model parameters, the seven parameters ($m, b, k, d, \alpha, \beta$, and γ) of the Bouc–Wen model are identified by minimizing the following fitness function [11]:

$$F(m, b, k, d, \alpha, \beta, \gamma) = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - x_i^{BW}\right)^2 \tag{2.3}$$

where N denotes the total number of samples, and $x_i - x_i^{BW}$ represents the error of the i-th sample which is calculated as the deviation of Bouc–Wen model output (x_i^{BW}) from experimental result (x_i).

The hysteresis model describes the relationship between the input voltage and output displacement of the system. The dynamics model with Bouc–Wen hysteresis model can be realized using MATLAB®/Simulink® blocks as shown in Fig. 2.4. On the other hand, the input voltage used to produce a desired displacement value is solved by the inverse hysteresis model [11]. Specifically, the inverse dynamic model can be derived from Eqs. (2.1) and (2.2). It is realized using MATLAB/Simulink blocks as shown in Fig. 2.5, where the "Bouc–Wen model" subsystem is depicted in Fig. 2.4b.

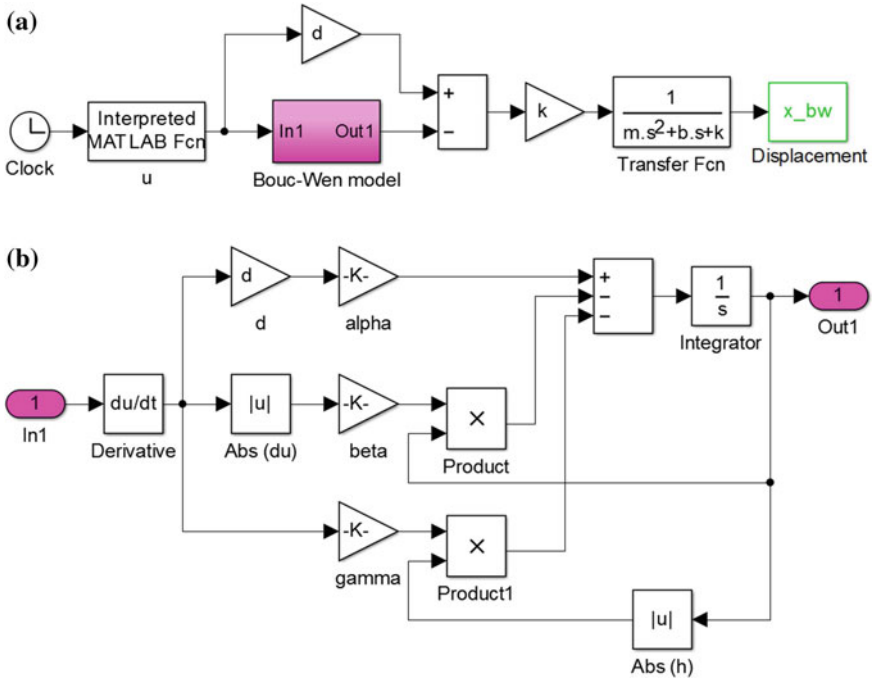

Fig. 2.4 Dynamic model implemented with MATLAB/Simulink blocks. **a** Dynamic model of the entire system. **b** Bouc–Wen hysteresis model

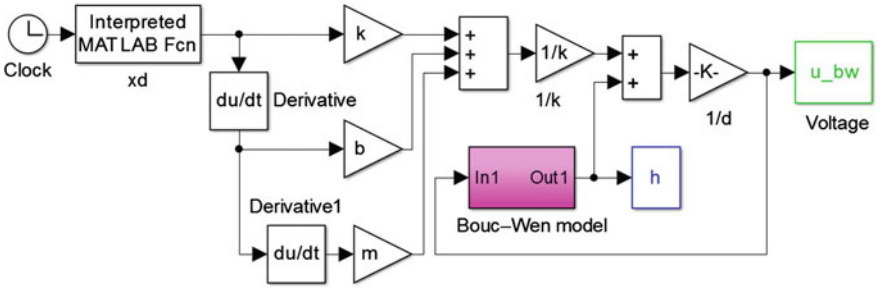

Fig. 2.5 The inverse dynamic model implemented with MATLAB/Simulink blocks

2.3.2 Hysteresis Modeling with the MPI Model

The hysteresis effect is described using MPI model, which has more parameters than the Bouc–Wen model. Then, an inverse hysteresis model is derived to compensate for the hysteresis effect.

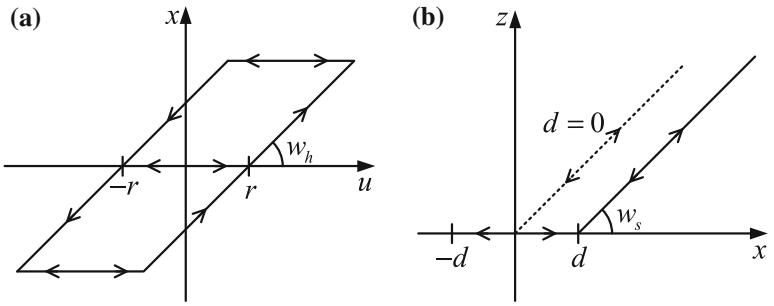

Fig. 2.6 Illustrations of elementary PI operators. **a** Backlash operator H_r. **b** Dead-zone operator S_d

2.3.2.1 PI Model

First, the Prandtl–Ishlinskii (PI) model is described briefly. As a subclass of the Preisach hysteresis model, the PI model is a superposition of elementary backlash or play operators. The backlash operator H_r is illustrated in Fig. 2.6a, which is defined by

$$\begin{aligned} y(t) &= H_r[u, x_0](t) \\ &= \max\{u(t) - r, \min\{u(t) + r, y(t - T)\}\} \end{aligned} \tag{2.4}$$

where u is the voltage control input, y is the displacement response of the stage, r is the control input threshold value or magnitude of the backlash, and T is the sampling time interval.

The initial condition of Eq. (2.4) is given by

$$y(0) = \max\{u(0) - r, \min\{u(0) + r, y_0\}\} \tag{2.5}$$

where y_0 is usually but not necessarily initialized to zero.

A generalized backlash operator can be obtained by multiplying H_r with a weight value w_h, i.e.,

$$y(t) = w_h H_r[u, y_0](t) \tag{2.6}$$

where the weight w_h describes the gain of the backlash operator.

Using a linearly weighted superposition of multiple backlash operators with different thresholds and weights, the complex hysteresis can be modeled by

$$y(t) = \mathbf{w}_h^T \mathbf{H}_r[u, \mathbf{y}_0](t) \tag{2.7}$$

where $\mathbf{w}_h^T = [w_{h0}, w_{h1}, \ldots, w_{hn}]$ is the weight vector, $\mathbf{H}_r[u, \mathbf{y}_0](t) = [H_r[u, y_{00}](t), H_r[u, y_{01}](t), \ldots, H_r[u, y_{0n}](t)]^T$ with the threshold vector $\mathbf{r} = [r_0, r_1, \ldots,$

$r_n]^T$ and the initial state vector $\mathbf{x}_0 = [x_{00}, x_{01}, \ldots, x_{0n}]^T$, for $0 = r_0 < r_1 < \cdots < r_n < +\infty$. The control input thresholds r_i can be chosen as equal intervals between the minimum and maximum voltage control input values of the piezoelectric actuation stage.

2.3.2.2 MPI Model

The PI operator has the same symmetric property as the backlash operator with respect to the center point of the loop formed by the operator. Hence, the model accuracy of PI operator will be reduced for the situations where the hysteresis loops are not symmetric. In order to overcome this issue, a saturation operator is adopted to connect in serial with the hysteresis operator.

Specifically, the saturation operator is a weighted linear superposition of linear-stop or one-sided dead-zone operators given as follows:

$$S_d[y](t) = \begin{cases} \max\{y(t) - d, 0\}, & d > 0 \\ y(t), & d = 0 \end{cases} \tag{2.8}$$

where the dead-zone operator S_d is depicted in Fig. 2.6b.

Based on the aforementioned dead-zone operator, the saturation operator is expressed by

$$z(t) = \mathbf{w}_s^T \mathbf{S}_d[y](t) \tag{2.9}$$

where y is the output of the hysteresis operator, z is the stage response, $\mathbf{w}_s^T = [w_{s0}, w_{s1}, \ldots, w_{sm}]$ is the weight vector, $\mathbf{S}_d[y](t) = [S_{d0}[y](t), S_{d1}[y](t), \ldots, S_{dm}[y](t)]^T$ with the threshold vector $\mathbf{d} = [d_0, d_1, \ldots, d_m]^T$, for $0 = d_0 < d_1 < \cdots < d_m < +\infty$.

Therefore, the modified PI operator is derived as follows:

$$z(t) = \Gamma[u](t) = \mathbf{w}_s^T \mathbf{S}_d\left[\mathbf{w}_h^T \mathbf{H}_r[u, \mathbf{y}_0]\right](t). \tag{2.10}$$

A realization of the MPI model using MATLAB/Simulink blocks is shown in Fig. 2.7.

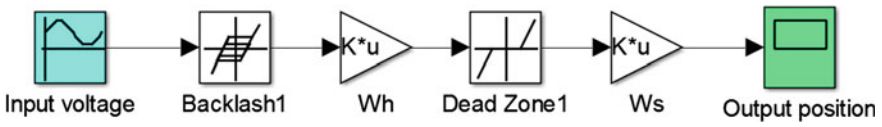

Fig. 2.7 MPI model implemented with MATLAB/Simulink blocks

2.3.2.3 Inverse MPI Model

In order to use the MPI model for the feedforward hysteresis compensation, an inverse model is required to express the voltage as a function of the position.

It has been shown that the inverse MPI model is also PI-type, which can be calculated by

$$u(t) = \Gamma^{-1}[z](t) = \mathbf{w}_h^{*T}\mathbf{H}_r^*\left[\mathbf{w}_s^{*T}\mathbf{S}_d^*[z], \mathbf{y}_0^*\right](t) \tag{2.11}$$

where $\mathbf{w}_h^{*T} = [w_{h0}^*, w_{h1}^*, \ldots, w_{hn}^*]$ and $\mathbf{w}_s^{*T} = [w_{s0}^*, w_{s1}^*, \ldots, w_{sm}^*]$ are the weight vectors, $\mathbf{y}_0^* = [y_{00}^*, y_{01}^*, \ldots, y_{0n}^*]^T$ is the initial state vector, and the threshold vectors are $\mathbf{r}^* = [r_0^*, r_1^*, \ldots, r_n^*]^T$ and $\mathbf{d}^* = [d_0^*, d_1^*, \ldots, d_m^*]^T$, respectively.

The inverse MPI model parameters are given as follows [1, 10]:

$$w_{h0}^* = \frac{1}{w_{h0}} \tag{2.12}$$

$$w_{hi}^* = \frac{-w_{hi}}{\left(\sum_{j=0}^{i} w_{hj}\right)\left(\sum_{j=0}^{i-1} w_{hj}\right)}, \quad i = 1, \ldots, n \tag{2.13}$$

$$r_i^* = \sum_{j=0}^{i} w_{hj}(r_i - r_j), \quad i = 0, 1, \ldots, n \tag{2.14}$$

$$y_{0i}^* = \sum_{j=0}^{i} w_{hj} y_{0i} + \sum_{j=i+1}^{n} w_{hj} y_{0j}, \quad i = 0, 1, \ldots, n \tag{2.15}$$

$$w_{s0}^* = \frac{1}{w_{s0}} \tag{2.16}$$

$$w_{sj}^* = \frac{-w_{sj}}{\left(\sum_{k=0}^{j} w_{sk}\right)\left(\sum_{k=0}^{j-1} w_{sk}\right)}, \quad j = 1, \ldots, m \tag{2.17}$$

$$d_j^* = \sum_{k=0}^{j} w_{sk}(d_j - d_k), \quad j = 0, 1, \ldots, m. \tag{2.18}$$

2.3.3 Hysteresis Modeling with the LSSVM

In this section, the hysteresis modeling is treated as a nonlinear regression problem, and the LSSVM is employed to model the piezoelectric hysteresis for the piezoelectric nanopositioning system. Due to the hysteresis effects, one input voltage can produce multiple values of the output displacements. Thus, one challenge lies in how to convert the one-to-many mapping into a one-to-one mapping. Here, both the current input and input variation rates are introduced to form the input data set, which

determines a unique output value. By making use of the input variation rate, the rate dependency of the hysteretic behavior can be captured.

Specifically, the hysteresis model is identified using the input voltage (u) and voltage variation rate ($\dot{u}$) as the inputs and the displacement (y) as the output to train the LSSVM.

2.3.3.1 LSSVM Modeling

LSSVM maps the input data into a high-dimensional feature space and constructs a linear regression function therein. The unknown hysteresis function is approximated by the equation

$$y(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) + b \tag{2.19}$$

with the given training data set $\{\mathbf{x}_i, y_i\}_{i=1}^N$, where N represents the number of training data set, $\mathbf{x}_i = \{u_i, \dot{u}_i\} \in \mathbb{R}^2$ are the input data, and $y_i \in \mathbb{R}$ are the output data. Additionally, the weight vector $\mathbf{w} \in \mathbb{R}^{n_h}$, the nonlinear mapping $\varphi(\cdot) : \mathbb{R}^2 \to \mathbb{R}^{n_h}$ denotes a map from the input space to a feature space, and b is the bias.

The LSSVM approach formulates the regression as an optimization problem in the primal weight space:

$$\min_{\mathbf{w},b,\mathbf{e}} J(\mathbf{w}, \mathbf{e}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{2}\gamma \sum_{i=1}^{N} e_i^2 \tag{2.20}$$

subject to the equality constraints

$$y_i = \mathbf{w}^T \varphi(\mathbf{x}_i) + b + e_i, \quad i = 1, 2, \ldots, N \tag{2.21}$$

where e_i is the residual error.

The weight vector $\mathbf{w}$ may be in high dimension, and it cannot be solved directly. In order to solve the optimization problem, a Lagrangian function is defined as follows:

$$L(\mathbf{w}, b, \mathbf{e}; \boldsymbol{\alpha}) = J(\mathbf{w}, \mathbf{e}) - \sum_{i=1}^{N} \alpha_i [\mathbf{w}^T \varphi(\mathbf{x}_i) + b + e_i - y_i] \tag{2.22}$$

where α_i are the Lagrange multipliers which can be either positive or negative values. The conditions for optimality can be obtained by solving the following partial derivatives:

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \to \mathbf{w} = \sum_{i=1}^{N} \alpha_i \varphi(\mathbf{x}_i) \tag{2.23}$$

$$\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=1}^{N} \alpha_i = 0 \tag{2.24}$$

$$\frac{\partial L}{\partial e_i} = 0 \rightarrow \alpha_i = \gamma e_i, \quad i = 1, 2, \ldots, N \tag{2.25}$$

$$\frac{\partial L}{\partial \alpha_i} = 0 \rightarrow \mathbf{w}^T \varphi(\mathbf{x}_i) + b + e_i - y_i = 0, \quad i = 1, 2, \ldots, N \tag{2.26}$$

which can be used to construct the dual formulation by eliminating $\mathbf{w}$ and e_i, i.e.,

$$\underbrace{\begin{bmatrix} 0 & \mathbf{1}_N^T \\ \mathbf{1}_N & \Omega + \gamma^{-1}\mathbf{I}_N \end{bmatrix}}_{\boldsymbol{\Phi}} \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix} \tag{2.27}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_N]^T$ is called the support vector, whose support values α_i are proportional to the residual errors e_i by the regularization factor $\gamma \in \mathbb{R}$. In addition, $\mathbf{1}_N = [1, 1, \ldots, 1]^T$, $\mathbf{y} = [y_1, y_2, \ldots, y_N]^T$, and $\mathbf{I}_N$ is an identity matrix.

Besides, the kernel trick is employed to derive that:

$$\Omega_{ij} = \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j), \quad i, j = 1, 2, \ldots, N \tag{2.28}$$

where K is a predefined kernel function. The purpose of introducing the kernel function is to avoid the explicit computation of the map $\varphi(\cdot)$ in dealing with the high-dimensional feature space.

It is observed that the LSSVM approach utilizes the equality constraints instead of the inequality constraints as in the ordinary SVM. Thus, it simplifies the regression to a problem that can be easily solved from a set of linear equations. Assume that $\boldsymbol{\Phi}$ is invertible, then b and $\boldsymbol{\alpha}$ can be calculated from Eq. (2.27):

$$\begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \boldsymbol{\Phi}^{-1} \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix}. \tag{2.29}$$

Then, in view of Eq. (2.23), the solution for the regression problem can be derived:

$$y(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b \tag{2.30}$$

where K is the kernel function satisfying Mercer's condition, $\mathbf{x}_i$ is the training data, and $\mathbf{x}$ denotes the new input data.

By adopting the RBF kernel function

$$K(\mathbf{x}, \mathbf{x}_i) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{\sigma^2}\right) \tag{2.31}$$

with $\sigma > 0$ denoting the width parameter (which specifies the kernel sample variance σ^2) and $\|\cdot\|$ representing the Euclidean distance, the LSSVM model for the hysteresis model estimation becomes

$$y(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{\sigma^2}\right) + b. \tag{2.32}$$

Once the regularization parameter γ and kernel parameter σ are assigned, a training process is needed to determine the support values α_i and the bias b.

2.3.3.2 Hyperparameters Tuning with Bayesian Inference

The above modeling procedure shows that there are two hyperparameters (γ and σ) to be adjusted. The high generalization ability of the LSSVM model relies on the appropriate tuning of the two parameters. To select the best hyperparameters, the approach of tenfold cross validation is usually adopted. However, it is a very time-consuming method. Alternatively, it has been shown that the Bayesian framework with three levels of inference is an effective way to infer the optimal hyperparameters of the LSSVM regressors [15].

Given a set of training data, Bayesian inference is a robust framework to determine the distribution of the estimated model parameters based on the data sets. Afterward, the optimal model parameters can be predicted. The basic idea of the hyperparameters inference procedure using Bayesian framework arises from a modified version of the LSSVM regression problem as shown below:

$$\min_{\mathbf{w},b,\mathbf{e}} J(\mathbf{w}, \mathbf{e}) = \mu E_W + \zeta E_D \tag{2.33}$$

subject to:

$$e_i = y_i - [\mathbf{w}^T \varphi(\mathbf{x}_i) + b], \quad i = 1, 2, \ldots, N \tag{2.34}$$

with

$$E_W = \frac{1}{2}\mathbf{w}^T\mathbf{w} \tag{2.35}$$

$$E_D = \frac{1}{2}\sum_{i=1}^{N} e_i^2 = \frac{1}{2}\sum_{i=1}^{N}\left\{y_i - [\mathbf{w}^T \varphi(\mathbf{x}_i) + b]\right\}^2 \tag{2.36}$$

where μ is the new regularization factor and ζ denotes the variance of the noise for the residual error e_i.

The dual program of the above optimization problem is the same as Eq. (2.27). The hyperparameter γ is related to μ and ζ by $\gamma = \zeta/\mu$. It is noticeable that, by substituting Eq. (2.35) and the above relationship into Eq. (2.33), the same problem as described by Eq. (2.20) is generated. Given a training data set, the Bayesian inference algorithm for the calculation of the two optimal hyperparameters is presented in [16]. Here, the procedure is implemented by resorting to a MATLAB toolbox [4].

2.4 Experimental Studies

In this section, the hysteresis modeling for a piezo-driven nanopositioning stage using the Bouc–Wen, MPI, and LSSVM models is carried out by experimental studies.

Without loss of generality, the input voltage signal is chosen below in order to identify the hysteresis model:

$$u(t) = 5e^{-0.11t}[\sin(6\pi t e^{-0.345t} - 1.55) + 1.0] \tag{2.37}$$

which is depicted in Fig. 2.8a. In addition, the experimental output data are obtained as depicted in Fig. 2.8b.

2.4.1 Bouc–Wen Model Results

By setting a time interval of 0.02 s, 500 training data sets are obtained as shown in Fig. 2.9a. Then, the Bouc–Wen model is identified off-line by optimizing the seven parameters to minimize the fitness function in Eq. (2.3).

In view of the superior performance of PSO over alternative methods such as the direct search approach and genetic algorithm (GA), the PSO is adopted for the parameter optimization. The optimization is carried out with a PSO toolbox [2] running in MATLAB environment. The identified model parameters are shown in Table 2.1.

The experimental result and the simulated Bouc–Wen model output are compared in Fig. 2.9a. The plots indicate that the Bouc–Wen model cannot exactly represent the complicated hysteresis of the system. A relative large error exists between the identified model output and the experimental result as shown in Fig. 2.9b. Specifically, the maximum model error is 4.70 μm, which accounts for 5.3 % of the concerned travel range of the nanopositioning stage. It is observed that a smaller model error is obtained when the input has lower magnitude and frequency (2–4 s). Hence, the model error varies greatly at different amplitudes and frequencies of the input signal, which indicates that the Bouc–Wen model cannot capture the rate dependency of the hysteresis precisely.

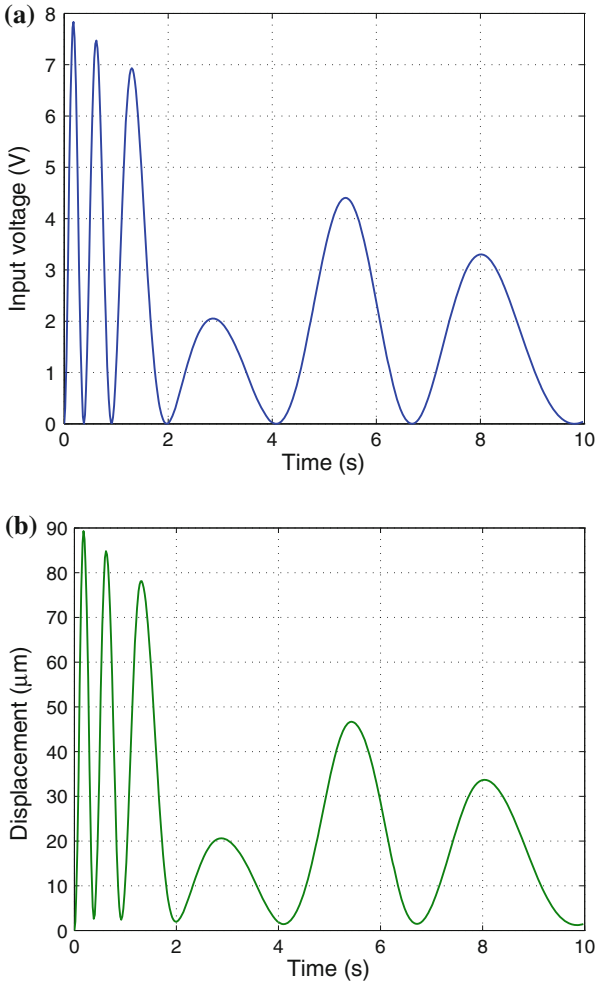

Fig. 2.8 Time history of the training data sets. **a** Input voltage to the high-voltage amplifier. **b** Output displacement of the nanopositioning stage

2.4.2 MPI Model Results

In order to identify the MPI model, the key step lies in the weight parameter determination for matching the model output to the experimental hysteresis data. Based on the input and output data sets, the thresholds r_i and d_i^* are assigned as follows:

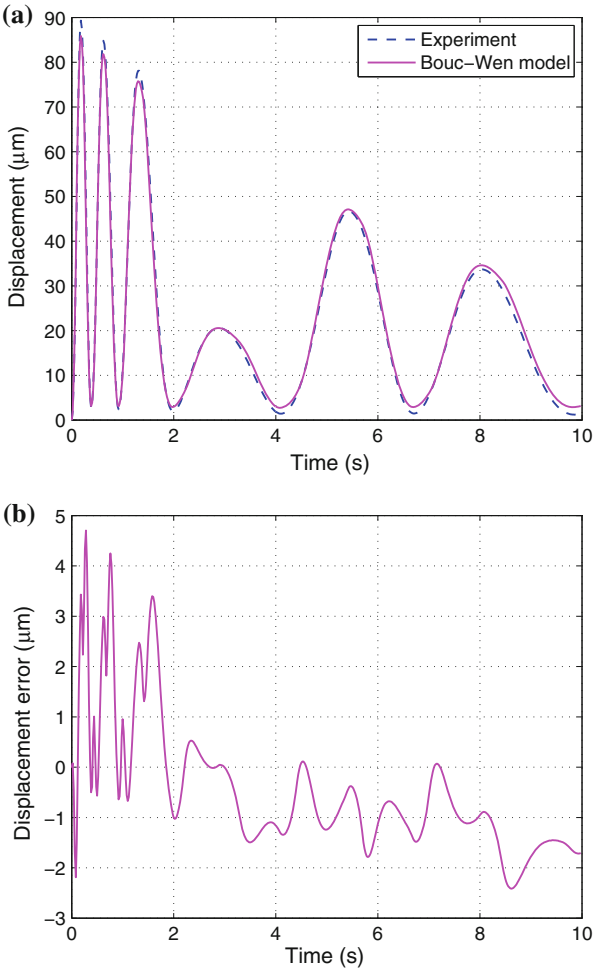

Fig. 2.9 Results of the identified Bouc–Wen model. **a** Experimental result and Bouc–Wen model output. **b** Bouc–Wen model output errors

$$r_i = \frac{i}{n+1} \max |u(t)|, \quad \text{for} \quad i = 0, 1, \ldots, n \tag{2.38}$$

$$d_j^* = \frac{j}{m+1} \max |y(t)|, \quad \text{for} \quad j = 0, 1, \ldots, m \tag{2.39}$$

where u and y denote the voltage input and displacement output data of the piezostage system, respectively. In addition, the initial states are assigned as $y_{0i} = 0, i = 0, 1, \ldots, n$, for simplicity.

Table 2.1 Parameters of the identified Bouc–Wen hysteresis model

Parameter	Search space	Identified value
m	[0, 1]	0.2789 kg
b	$[10^2, 10^6]$	8.9055×10^5 N s/m
k	$[10^3, 10^9]$	3.5244×10^8 N/m
d	[0, 0.1]	1.1351×10^{-5} m/V
α	[0, 1]	0.4754
β	[0, 1]	0.8718
γ	[0, 1]	0.9925

Then, the weights $\mathbf{w}_h$ and $\mathbf{w}_s^*$ are identified by solving an optimization problem of minimizing the model error:

$$E[u, y](\mathbf{w}_h, \mathbf{w}_s^*, t) = \mathbf{w}_h^T \mathbf{H}_r[u, \mathbf{y}_0](t) - \mathbf{w}_s^{*T} \mathbf{S}_d^*[y](t). \tag{2.40}$$

Here, instead of using L_2^2-norm optimization [10], the weight parameters are identified by resorting to the PSO approach. Additionally, $n = 12$ and $m = 6$ are assigned, and the thresholds r_i and d_j^* are allocated according to Eqs. (2.38) and (2.39). Using the same training data sets as shown in Fig. 2.8a, b, the MPI model is identified off-line by optimizing a total of $n + m + 2 = 20$ weight parameters ($\mathbf{w}_h$ and $\mathbf{w}_s^*$) within the search space of [0, 1] using the PSO algorithm [2].

The identified MPI model and inverse MPI model parameters are described in Tables 2.2 and 2.3, respectively. The presence of zero weight values of w_{hi} and w_{sj}^* indicates that the selected n and m are large enough for the modeling process.

The comparisons of the experimental output and simulated MPI model output are illustrated in Fig. 2.10. It is observed from Fig. 2.10a that the MPI model cannot

Table 2.2 Parameters of the identified MPI hysteresis model

i	r_i	w_{hi}	j	d_j	w_{sj}
0	0	0.0677	0	0	99.9999
1	0.6036	0.0228	1	0.1276	-1.0010×10^{-5}
2	1.2072	6.5979×10^{-6}	2	0.2552	-3.3798×10^{-5}
3	1.8108	0.0196	3	0.3827	-1.5611×10^{-5}
4	2.4145	0.0037	4	0.5103	0
5	3.0181	4.0252×10^{-6}	5	0.6379	-2.6054×10^{-4}
6	3.6217	1.0681×10^{-6}	6	0.7655	-6.8291×10^{-5}
7	4.2253	6.7711×10^{-8}			
8	4.8289	6.1372×10^{-8}			
9	5.4325	3.8439×10^{-8}			
10	6.0362	2.9690×10^{-8}			
11	6.6398	8.4374×10^{-9}			
12	7.2434	0			

Table 2.3 Parameters of the inverse MPI hysteresis model

i	r_i^*	w_{hi}^*	j	d_j^*	w_{sj}^*
0	0	14.7797	0	0	0.0100
1	0.0408	−3.7195	1	12.7581	1.0010×10^{-9}
2	0.0954	-8.0705×10^{-4}	2	25.5163	1.3798×10^{-9}
3	0.1500	−1.9672	3	38.2744	1.5611×10^{-9}
4	0.2164	−0.2980	4	51.0326	0
5	0.2850	-3.1129×10^{-4}	5	63.7907	2.6054×10^{-8}
6	0.3537	-8.2597×10^{-5}	6	76.5488	6.8292×10^{-9}
7	0.4223	-5.2362×10^{-6}			
8	0.4909	-4.7460×10^{-6}			
9	0.5596	-2.9725×10^{-6}			
10	0.6282	-2.2959×10^{-6}			
11	0.6969	-6.5247×10^{-7}			
12	0.7655	0			

exactly represent the complicated hysteresis of the nanopositioning system. A relative large error exists between the identified model output and the experimental result as shown in Fig. 2.10b. Specifically, the maximum model error is 6.35 μm which occurs along with the input component of the highest frequency (0–0.38 s). It accounts for 7.1 % of the concerned travel range of the nanopositioning stage. On the other hand, the model error is reduced at lower frequency with smaller amplitude of the input signal, that is, the model errors vary significantly at different frequencies and amplitudes of the input signal, which means that the MPI model cannot capture the rate dependency of the hysteresis either. Due to the existence of zero weights in $\mathbf{w}_h$ and $\mathbf{w}_s^*$, the increases of m and n will not cause significant improvement on the model accuracy. Hence, the capability of the MPI model is limited in modeling the rate-dependent hysteresis.

2.4.3 LSSVM Model Results

To identify the LSSVM model, the 500 sets of input and output data sets as shown in Fig. 2.8 are also adopted to train the LSSVM. To capture the rate dependency behavior, the input rate is numerically calculated by the backward difference equation:

$$\dot{u}(kT) = \frac{u(kT) - u(kT - T)}{T} \tag{2.41}$$

where T is the sampling time and k denotes the index of the time series. The two input variables are shown in Fig. 2.11.

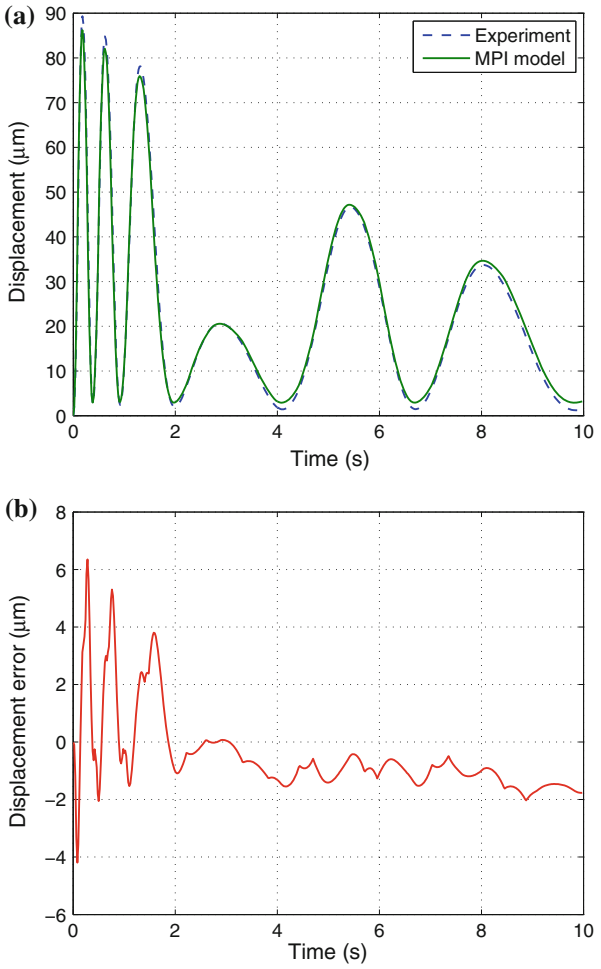

Fig. 2.10 Results of the identified MPI model. **a** Experimental result and MPI model output. **b** MPI model output errors

To facilitate the LSSVM modeling, the data sets are normalized before the training process. Specifically, both the input and output data are normalized within the range of [0, 1] by the following transformation [17]:

$$\mathcal{N}(v) = v^* = \frac{v - v_{\min}}{v_{\max} - v_{\min}} \tag{2.42}$$

where $\mathcal{N}$ represents the normalization operation, and $v_{\min}$ and $v_{\max}$ denote the minimum and maximum values of the input or output data v, respectively. It is notable that, after the training process, the LSSVM output y^* is re-transformed into the actual value y using the inverse normalization operation $\mathcal{N}^{-1}$.

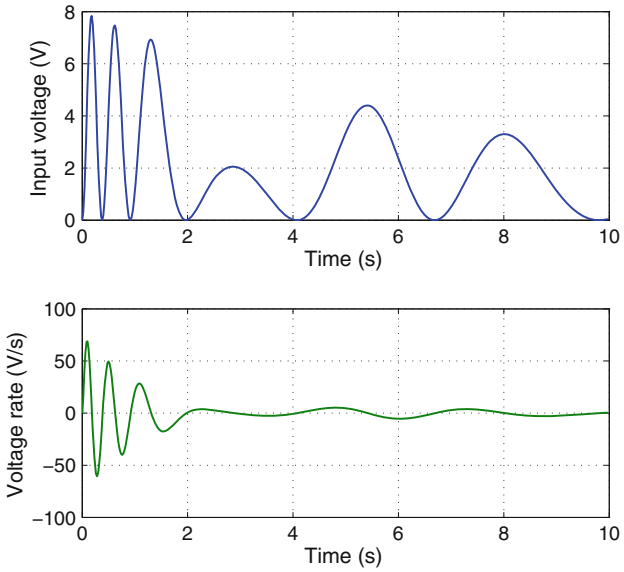

Fig. 2.11 Training data sets of the two input variables for the LSSVM model

Given arbitrary initial values ($\gamma_0 = 100$ and $\sigma_0 = 1$), the optimized LSSVM model hyperparameters are obtained by the Bayesian framework automatically. The optimal hyperparameters are $\gamma = 741.57$ and $\sigma = 7.39$. They are used to train the LSSVM to obtain the values of α_i and b, which minimizes the objection function (2.22).

Once the training process is completed, the LSSVM model produces the outputs as illustrated in Fig. 2.12. It is seen that the maximum model error is only 1.27 μm, i.e., 1.4 % of the motion range. Thus, with the same number of training data sets, the LSSVM model error has been reduced by 79 and 80 % as compared with the Bouc–Wen and MPI model errors, respectively. In addition, Fig. 2.12b exhibits that the LSSVM model errors are more uniformly distributed in comparison with Bouc–Wen and MPI model results as shown in Figs. 2.9b and 2.10b, respectively. Relatively, the LSSVM model errors are not dependent on neither the amplitude nor the frequency of the input signals. Therefore, the trained LSSVM model captures the amplitude- and rate-dependent hysteresis accurately.

2.4.4 Model Capability Comparison

Based on the displacement error $e = y_d - y$, where y_d and y represent the desired and actual displacements, respectively, the mean absolute error (MAE) and root-mean-square error (RMSE) are defined as follows:

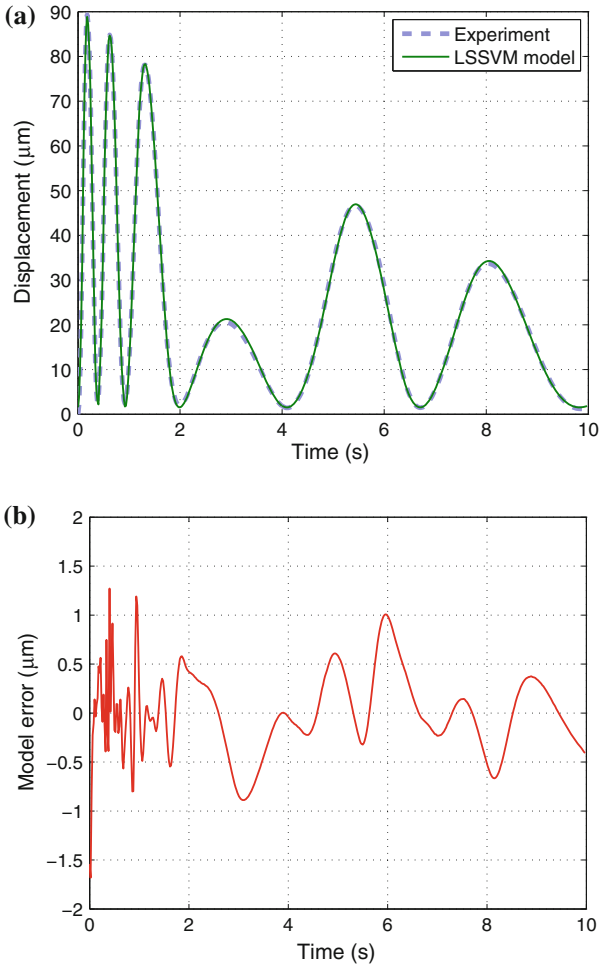

Fig. 2.12 Results of the trained LSSVM model. **a** Experimental result and LSSVM model output. **b** LSSVM model output errors

$$\text{MAE} = \frac{1}{N_t}\sum_{i=1}^{N_t}|e_i| \tag{2.43}$$

$$\text{RMSE} = \sqrt{\frac{1}{N_t}\sum_{i=1}^{N_t}e_i^2} \tag{2.44}$$

where $N_t = 500$ is the number of test data sets in the experiment.

The modeling performances of the three models are tabulated in Table 2.4. Due to the fewer number of parameters, the Bouc–Wen model takes 19.1 % shorter time than

Table 2.4 Modeling performances of Bouc–Wen, MPI, and LSSVM models

Specification	Bouc–Wen	MPI	LSSVM
No. of model parameters	7	20	2
Model identification time (s)	614.89	759.91	2.56
Modeling error			
MAE (%)	1.276	1.358	0.364
RMSE (%)	1.560	1.692	0.470
Generalization error			
MAE (%)	1.253	1.318	0.621
RMSE (%)	1.530	1.621	0.858

the MPI model to identify all the parameters by resorting to the same optimization approach (PSO) in 1500 epoches. On the other hand, the LSSVM model needs a significantly shorter time than both Bouc–Wen and MPI models to identify the model parameters.

Concerning the modeling error, it is observed that the Bouc–Wen model produces 6.0 % lower MAE and 7.8 % lower RMSE than the MPI model. In contrast, the LSSVM model achieves the best results among the three methods. Regarding MAE, the LSSVM has improved the accuracy by 71.5 and 73.2 % in comparison with Bouc–Wen and MPI models, respectively. As far as RMSE is concerned, the LSSVM has enhanced the accuracy by 69.9 and 72.2 % as compared with Bouc–Wen and MPI models, respectively.

2.4.5 Generalization Study

To test the generalization of the obtained models, a new input signal is chosen as follows:

$$u(t) = 5e^{-0.14t}[\cos(3\pi t e^{-0.092t} - 3.15) + 1.0] \tag{2.45}$$

which is depicted in Fig. 2.13a. In addition, the displacement output of the nanopositioning stage is shown in Fig. 2.13b.

The Bouc–Wen, MPI, and LSSVM model outputs are depicted in Fig. 2.14. The model output errors with respect to the actual output obtained by experiments are illustrated in Fig. 2.15. Table 2.4 describes the generalization testing results of the three types of models.

It is found that the Bouc–Wen model produces a MAE of 1.086 μm and a RMSE of 1.326 μm, which accounts for 1.25 and 1.53 % of the motion range, respectively. Using the MPI model, the MAE and RMSE are 1.143 and 1.405 μm, which are equivalent to 1.32 and 1.62 % of the overall motion range, respectively. As compared

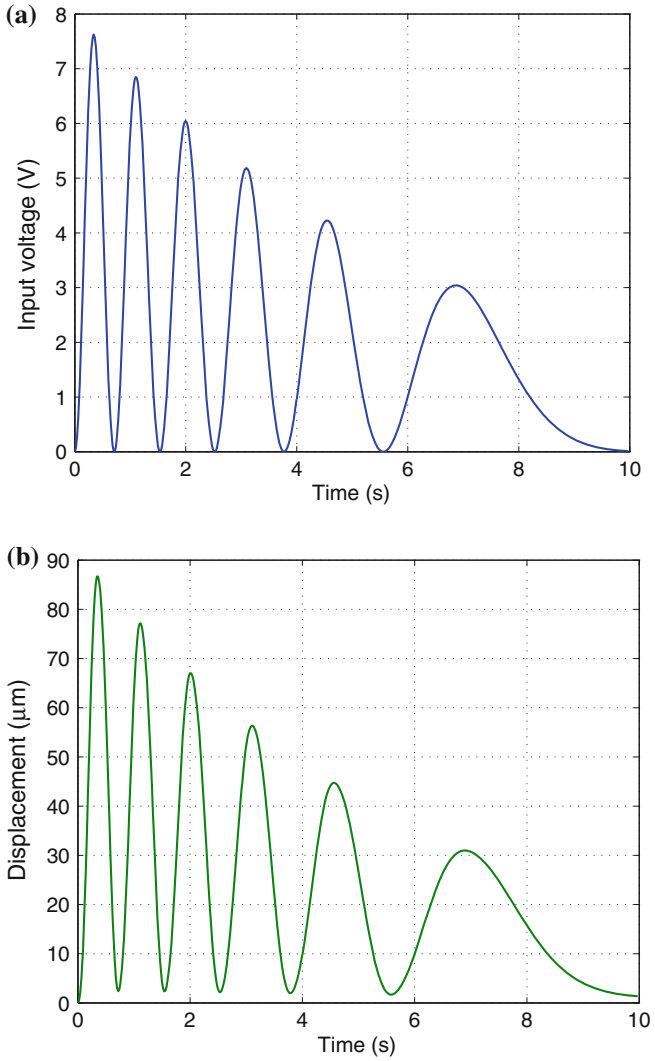

Fig. 2.13 Time history of the testing data sets. **a** Input voltage to the high-voltage amplifier. **b** Output displacement of the nanopositioning stage

with MPI model, the Bouc–Wen model reduces the MAE and RMSE by 4.9 and 5.6 %, respectively. Thus, the generalization ability of the Bouc–Wen model is slightly better than that of the MPI model.

In contrast, the LSSVM model produces a MAE of 0.538 μm and a RMSE of 0.744 μm, i.e., 0.62 and 0.86 % of the overall motion range, which have been reduced by 50.5 and 43.9 % in comparison with the Bouc–Wen model results and by 52.9

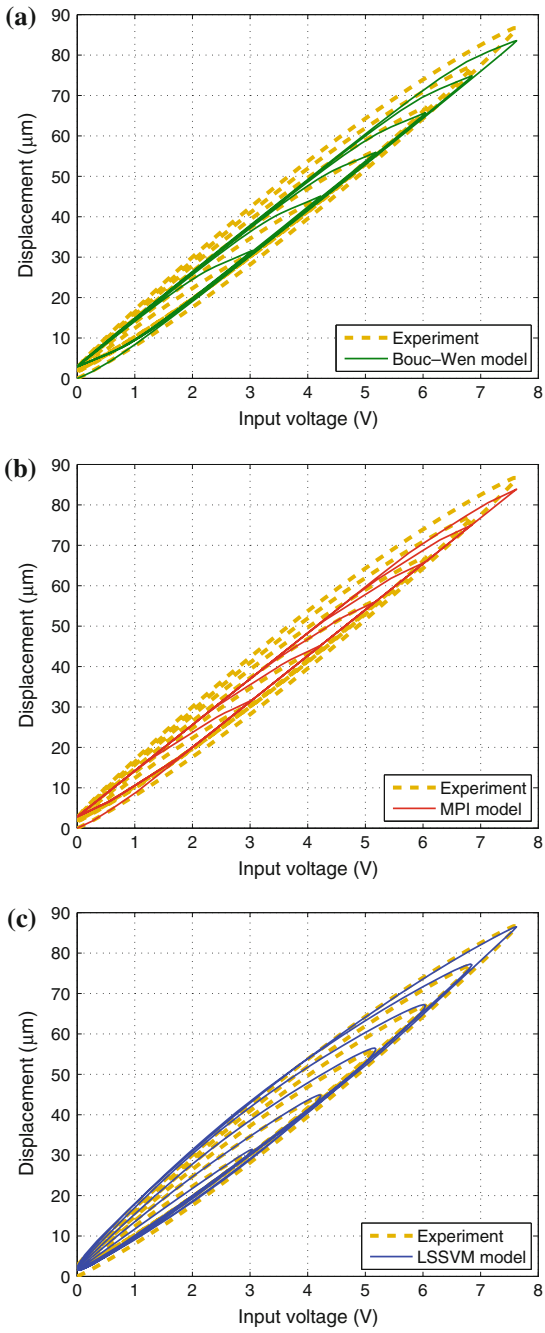

Fig. 2.14 Model generalization testing results. **a** Boun-Wen model output. **b** MPI model output. **c** LSSVM model output

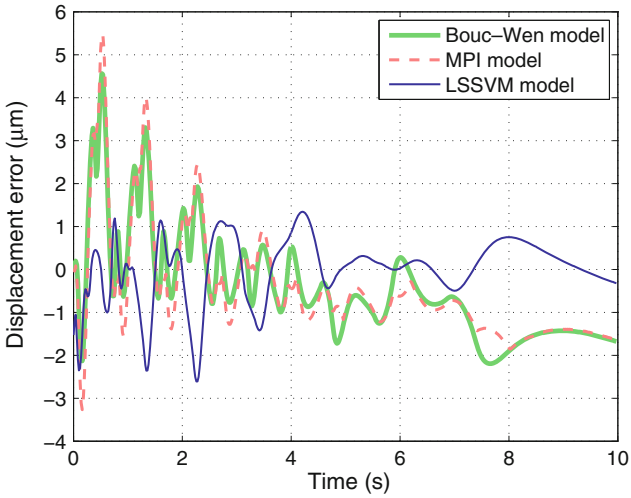

Fig. 2.15 Displacement errors of the Bouc–Wen model, MPI model, and LSSVM model

and 47.0 % in comparison with the MPI model outputs, respectively. Therefore, the experimental results demonstrate a good generalization capacity of the LSSVM hysteresis model.

2.5 Controller Design and Verification

Based on the established hysteresis models, a feedforward plus feedback control scheme is implemented and verified for the hysteresis compensation in this section. Both the inverse Bouc–Wen model and inverse MPI model are adopted for comparisons with the LSSVM inverse model-based control scheme.

2.5.1 Feedforward Controller Design

To compensate for the hysteresis nonlinearity, a feedforward (FF) control based on the inverse hysteresis model can be constructed. The FF control effort u_{FF}^{BW} is created by the inverse Bouc–Wen model. Similarly, the inverse MPI model is used to produce the FF control signal u_{FF}^{MPI}. The overall control scheme is shown in Fig. 2.16a, where the feedback (FB) control u_{FB} is adopted to suppress the residual control error due to the model errors of the identified Bouc–Wen and MPI models.

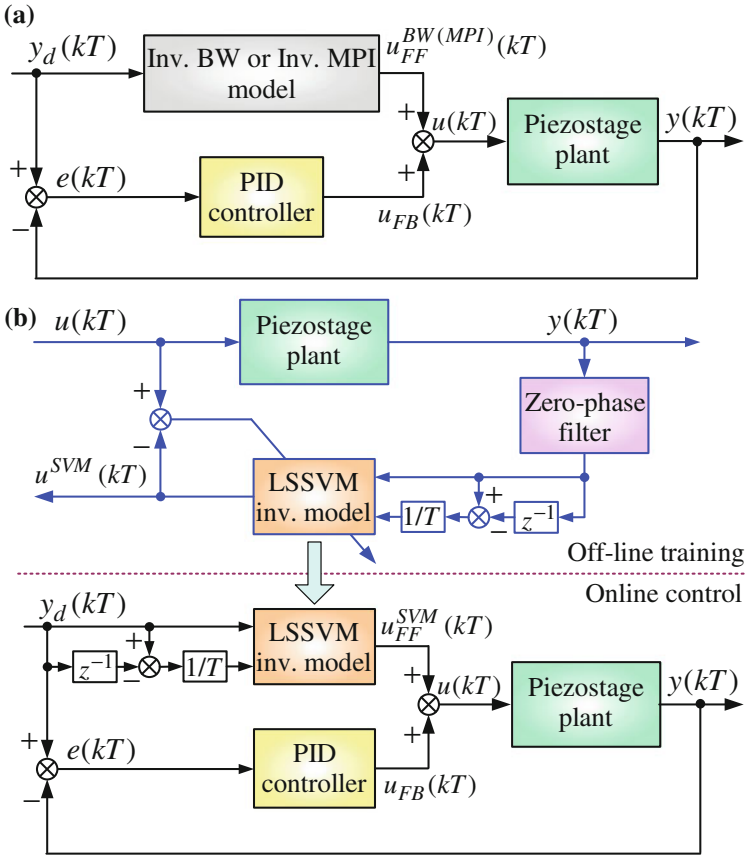

Fig. 2.16 Block diagram of the feedforward (FF) plus feedback (FB) hybrid controller for a piezostage system. **a** Bouc–Wen model or MPI model-based control. **b** LSSVM model-based control

To establish an inverse LSSVM hysteresis model, the displacement (y) and velocity ($\dot{y}$) are selected as the inputs and the corresponding voltage (u) is assigned as the output for the LSSVM training. Preliminary investigations indicate that the noisy sensor readings produce slightly oscillating voltage input by the inverse LSSVM model. Thus, to alleviate the noises in the displacement sensor readings, the displacement data y are filtered by adopting a zero-phase digital filter with 50-Hz cutoff frequency. The filter is implemented with the command "filtfilt" in MATLAB. Afterward, the input data sets are formed by the filtered displacement data and the calculated velocity. Once trained off-line using the scheme as shown in Fig. 2.16b, the LSSVM inverse model provides the online FF control signal u_{FF}^{SVM}.

2.5.2 Feedforward Plus Feedback Controller Design

Due to the presence of modeling error, the hysteresis cannot be completely eliminated by the stand-alone inverse model-based FF compensator. Therefore, a FB control u_{FB} is employed to create a hybrid control as described in Fig. 2.16.

Here, the PID control algorithm is implemented as a FB control due to its robustness and popularity. By adopting an incremental PID algorithm, the overall control input can be derived in the discretized form:

$$\begin{aligned} u(kT) &= u_{FF}(kT) + u_{FB}(kT) \\ &= u_{FF}(kT) + u_{FB}(kT - T) \\ &\quad + K_p\,[e(kT) - e(kT - T)] + K_i\,e(kT) \\ &\quad + K_d\,[e(kT) - 2e(kT - T) + e(kT - 2T)] \end{aligned} \tag{2.46}$$

where e represents the displacement error, $u_{FB}(kT - T)$ is the FB control command in the previous time step, and the FF term $u_{FF}(kT)$ is given by the inverse MPI model or LSSVM inverse hysteresis model. Additionally, K_p, K_i, and K_d are the proportional, integral, and derivative gains, respectively.

2.5.3 Controller Verification

In the control experiments, the sampling time interval is assigned as $T = 0.001$ s. Actually, a PI control is employed. The control gains are tuned as $K_p = 0.0864$, $K_i = 31.1040$, and $K_d = 0$ using the Ziegler–Nichols (Z–N) method.

For a reference displacement as shown in Fig. 2.17a, the Bouc–Wen (BW) and MPI model-based FF control results are both shown in Fig. 2.17. The two FF plus PID FB control results are depicted in Fig. 2.18. In addition, the LSSVM inverse hysteresis model is trained with the hyperparameters $\gamma = 741.82$ and $\sigma = 7.69$, which are optimized by the Bayesian inference framework. Using the same reference input as shown in Fig. 2.17a, the control results and tracking errors of the LSSVM model-based FF and FF+FB control are also shown in Figs. 2.17 and 2.18.

Figure 2.17b reveals that the inverse Bouc–Wen model-based FF gives a MAE of 1.199 % and a RMSE of 1.660 % with respect to the motion range. On the other hand, the FF control using the inverse MPI model produces a MAE of 1.080 % and a RMSE of 1.813 %. It is seen that the Bouc–Wen model and MPI model produce similar results. While the former achieves a slightly better RMSE than the latter, the latter produces a better MAE than the former. By contrast, the LSSVM inverse model-based FF approach further reduces the MAE and RMSE to 0.416 and 0.537 %, respectively, which have been significantly improved by 65.3 and 67.7 % in comparison with the inverse Bouc–Wen model results and by 61.5 and 70.4 % in comparison with the inverse MPI model results, respectively.

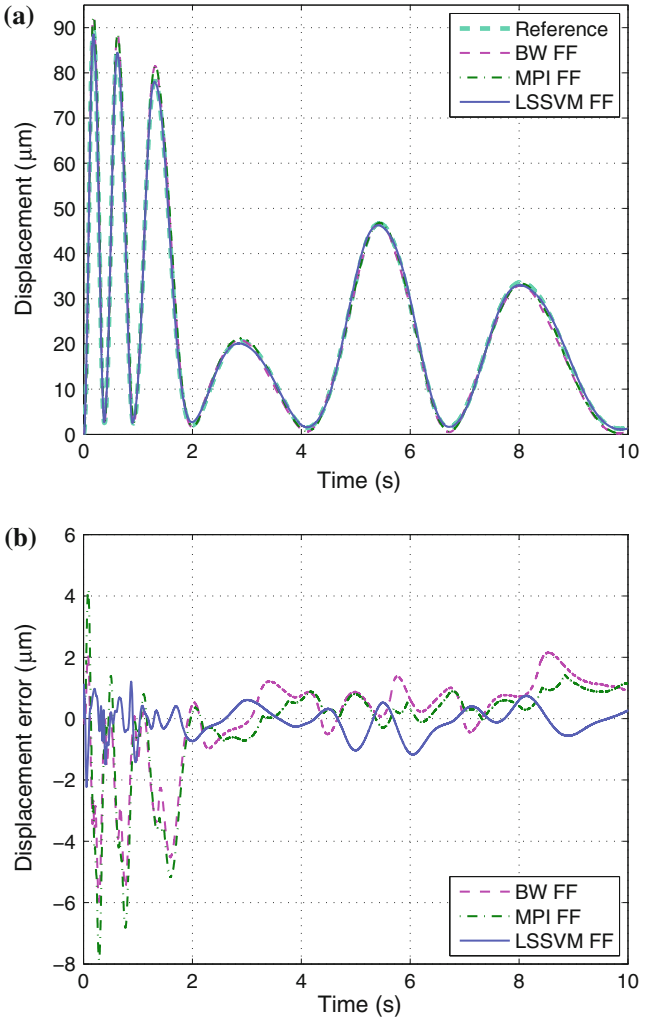

Fig. 2.17 Bouc–Wen, MPI, and LSSVM-based FF control results. **a** Feedforward (FF) control results. **b** Control errors of the three FF methods

Moreover, the hybrid control results as shown in Fig. 2.18b indicate that the inverse Bouc–Wen model-based FF+FB control scheme leads to a MAE of 0.024 % and a RMSE of 0.048 %. In addition, the inverse MPI model-based FF+FB control scheme results in a MAE of 0.023 % and a RMSE of 0.052 %, while the LSSVM-based hybrid control substantially suppresses the MAE to 0.015 % and RMSE to 0.038 %. As compared to the inverse Bouc–Wen hysteresis model, the LSSVM inverse model-based FF+FB control reduces the tracking errors in terms of MAE and RMSE by 37.5 and 20.8 %, respectively. Moreover, the LSSVM inverse model-based FF+FB control

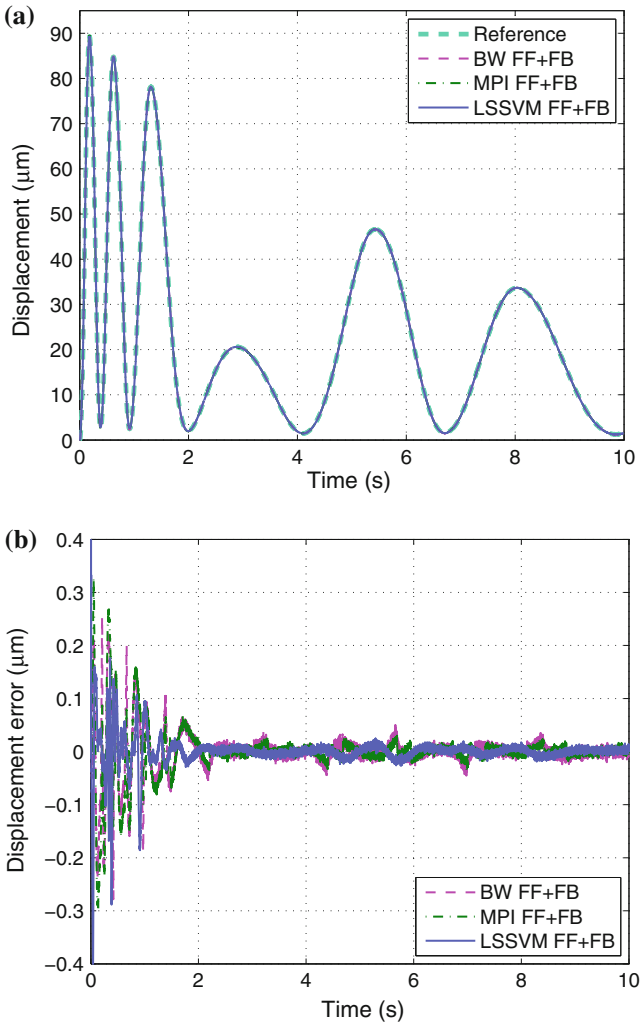

Fig. 2.18 Bouc–Wen, MPI, and LSSVM-based FF plus FB control results. **a** FF plus feedback (FB) control results. **b** Control errors of the three hybrid approaches

suppresses the MAE and RMSE by 34.8 and 26.9 %, respectively, as compared with the inverse MPI model-based control. This is caused by the fact that the model accuracy of LSSVM is superior to that of both Bouc–Wen and MPI models. The effectiveness of the LSSVM-based control is evident from the control results.

In addition, to reveal the efficiency of the hybrid control, the control results of the stand-alone FF and FB and the combined FF+FB control based on the LSSVM model are illustrated in Fig. 2.19. It can be observed from Fig. 2.19b that the FF+FB produces the MAE and RMSE of 0.015 and 0.038 %, respectively, which have been

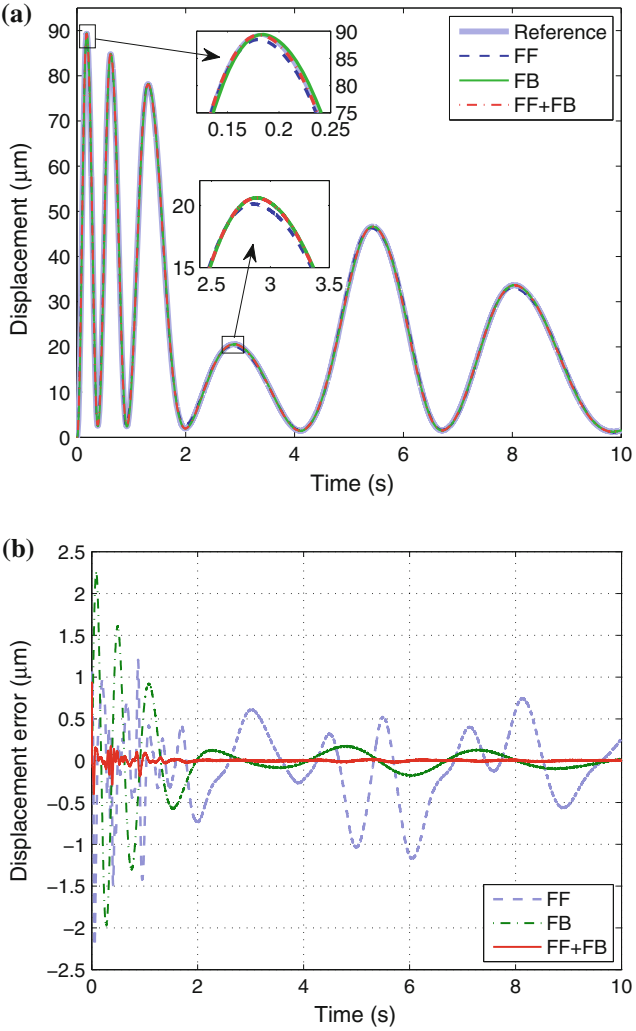

Fig. 2.19 LSSVM-based stand-alone and hybrid control results. **a** FF, FB, and FF+FB results. **b** Tracking errors of the three methods

significantly improved by 96.5 and 92.9 % in comparison with the FF control results, and substantially enhanced by 93.7 and 91.7 % as compared with the stand-alone FB outputs, respectively. As a result, the hysteresis effects have been suppressed by the FF+FB approach to a negligible level as indicated in Fig. 2.20a. Although the PID FB control results can be improved by tuning the control gains larger, it is at the risk of oscillations in the hybrid control results. With the hybrid control, the individual control input components are illustrated in Fig. 2.20b. It is found that the overall control effort is dominated by the FF control, whereas the FB control effort is minor.

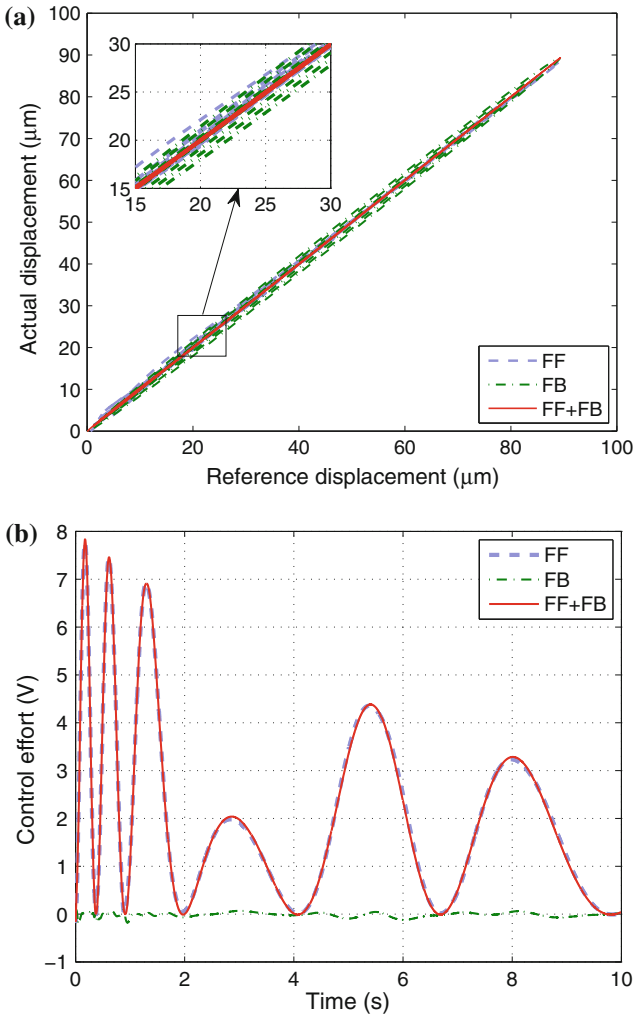

Fig. 2.20 LSSVM-based control results. **a** Actual versus reference displacements obtained by the three approaches. **b** Control signal components of the FF+FB hybrid control

Even so, the FB control is capable of suppressing the residual tracking errors and leading to a better result than the stand-alone FF method, which is verified by the tracking errors as compared in Fig. 2.19b.

For the purpose of comparison, the control results obtained by the Bouc–Wen, MPI, and LSSVM inverse models are tabulated in Table 2.5, where the MAE and RMSE are calculated as percentage values with respect to the motion range. It is obvious that the FF control accuracy of the LSSVM model has been enhanced by over 65 and 60 % as compared with those of the Bouc–Wen and MPI models, respectively.

Table 2.5 Control performances using Bouc–Wen, MPI, and LSSVM models

Controller	Performance (%)	Bouc–Wen	MPI	LSSVM
FF	MAE	1.199	1.080	0.416
	RMSE	1.660	1.813	0.537
FF+FB	MAE	0.024	0.023	0.015
	RMSE	0.048	0.052	0.038

With the FF+FB hybrid control, the LSSVM approach has suppressed the control errors by more than 20 and 25 % in comparison with the Bouc–Wen and MPI models, respectively. Therefore, the superiority of the LSSVM model over the popular Bouc–Wen and MPI models is verified for the hysteresis compensation task.

2.6 Chapter Summary

The results presented in this chapter show that the rate-dependent hysteresis of a piezoelectric nanopositioning stage can be accurately modeled and effectively suppressed by the LSSVM regression model and LSSVM inverse model, respectively. By selecting the input variation rate as an auxiliary input variable, the multi-valued mapping due to the hysteresis nonlinearity is converted into a one-to-one mapping, and the LSSVM is trained to capture the rate-dependent hysteretic behavior. Its superior modeling performance over the popular Bouc–Wen and MPI hysteresis models has been revealed by experimental studies. Results also demonstrate that the hybrid control using the LSSVM inverse model-based feedforward control combined with a PID control is capable of compensating the hysteresis nonlinearity effectively. Due to a simple structure of the presented modeling and control framework, it can be easily extended to the hysteretic systems driven by shape-memory alloy or other types of smart actuators as well. As a future work, an incremental LSSVM model can be established to automatically update the model parameters online once new training data occur.

References

1. Ang, W.T., Khosla, P.K., Riviere, C.N.: Feedforward controller with inverse rate-dependent model for piezoelectric actuators in trajectory-tracking applications. IEEE/ASME Trans. Mechatron. **12**(2), 134–142 (2007)
2. Birge, B.: PSOt—a particle swarm optimization toolbox for use with Matlab. In: Proceedings of IEEE Swarm Intelligence Symposium, pp. 182–186. Indianapolis, Indiana, USA (2003)
3. Boukari, A.F., Carmona, J.C., Moraru, G., Malburet, F., Chaaba, A., Douimi, M.: Piezo-actuators modeling for smart applications. Mechatronics **21**(1), 339–349 (2011)

4. De Brabanter, K., Karsmakers, P., Ojeda, F., Alzate, C., De Brabanter, J., Pelckmans, K., De Moor, B., Vandewalle, J., Suykens, J.A.K.: LS-SVMlab toolbox user's guide version 1.7. Internal Report 10-146, ESAT-SISTA, K.U.Leuven, Leuven, Belgium (2010)
5. Dong, R., Tan, Y., Chen, H., Xie, Y.: A neural networks based model for rate-dependent hysteresis for piezoelectric actuators. Sens. Actuator A-Phys. **143**(2), 370–376 (2008)
6. Ge, P., Jouaneh, M.: Tracking control of a piezoceramic actuator. IEEE Trans. Contr. Syst. Technol. **4**(3), 209–216 (1996)
7. Janaideh, M.A., Rakheja, S., Su, C.Y.: Experimental characterization and modeling of rate-dependent hysteresis of a piezoceramic actuator. Mechatronics **19**(5), 656–670 (2009)
8. Juhasz, L., Maas, J., Borovac, B.: Parameter identification and hysteresis compensation of embedded piezoelectric stack actuators. Mechatronics **21**(1), 329–338 (2011)
9. Kim, J., Kang, B.: Micro-macro linear piezoelectric motor based on self-moving cell. Mechatronics **19**(7), 1134–1142 (2009)
10. Kuhnen, K.: Modeling, identification and compensation of complex hysteretic nonlinearities: a modified Prandtl-Ishlinskii approach. Eur. J. Control **9**(4), 407–421 (2003)
11. Li, Y., Xu, Q.: Adaptive sliding mode control with perturbation estimation and PID sliding surface for motion tracking of a piezo-driven micromanipulator. IEEE Trans. Contr. Syst. Technol. **18**(4), 798–810 (2010)
12. Lin, C.J., Chen, S.Y.: Evolutionary algorithm based feedforward control for contouring of a biaxial piezo-actuated stage. Mechatronics **19**(6), 829–839 (2009)
13. Suykens, J.A.K., Vandewalle, J.: Least squares support vector machine classifiers. Neural Process. Lett. **9**(3), 293–300 (1999)
14. Suykens, J.A.K., Gestel, T.V., Brabanter, J.D., Moor, B.D., Vandewalle, J.: Least Squares Support Vector Machines. World Scientific Publishing Co., Singapore (2002)
15. Van Gestel, T., Suykens, J.A.K., Baestaens, D.E., Lambrechts, A., Lanckriet, G., Vandaele, B., De Moor, B., Vandewalle, J.: Financial time series prediction using least squares support vector machines within the evidence framework. IEEE Trans. Neural Netw. **12**(4), 809–821 (2001)
16. Vong, C.M., Wong, P.K., Li, Y.P.: Prediction of automotive engine power and torque using least squares support vector machines and Bayesian inference. Eng. Appl. Artif. Intell. **19**(3), 277–287 (2006)
17. Wong, P.K., Vong, C.M., Tam, L.M., Li, K.: Data preprocessing and modelling of electronically-controlled automotive engine power performance using kernel principal components analysis and least-square support vector machines. Int. J. Veh. Syst. Model. Test. **3**(4), 312–330 (2008)
18. Xu, Q., Li, Y.: Dahl model-based hysteresis compensation and precise positioning control of an XY parallel micromanipulator with piezoelectric actuation. J. Dyn. Syst. Meas. Control-Trans. ASME **132**(4), 041,011 (2010)
19. Yang, J., Bouzerdoum, A., Phung, S.L.: A training algorithm for sparse LS-SVM using compressive sampling. In: Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing, pp. 2054–2057 (2010)
20. Yu, S., Alici, G., Shirinzadeh, B., Smith, J.: Sliding mode control of a piezoelectric actuator with neural network compensating rate-dependent hysteresis. In: Proc. of IEEE Int. Conf. on Robotics and Automation, pp. 3641–3645 (2005)
21. Yu, Y., Xiao, Z., Naganathan, N.G., Dukkipati, R.V.: Dynamic Preisach modelling of hysteresis for the piezoceramic actuator system. Mech. Mach. Theory **37**(1), 75–89 (2002)

Springer

http://www.springer.com/978-3-319-21622-5

Advanced Control of Piezoelectric
Micro-/Nano-Positioning Systems

Xu, Q.; Tan, K.K.

2016, XX, 257 p., Hardcover
ISBN: 978-3-319-21622-5