Available online at www.sciencedirect.com
SCIENCE DIRECT®

Engineering Structures 28 (2006) 1728–1735

ENGINEERING STRUCTURES

www.elsevier.com/locate/engstruct

Online weighted LS-SVM for hysteretic structural system identification

He-Sheng Tang[a,*], Song-Tao Xue[a,b], Rong Chen[a], Tadanobu Sato[c]

[a] Research Institute of Structural Engineering and Disaster Reduction, Tongji University, Shanghai 200092, China
[b] Department of Architecture, School of Science and Engineering, Kinki University, Osaka 577-0056, Japan
[c] Disaster Prevention Research Institute, Kyoto University, Uji, Kyoto 611-0011, Japan

Received 9 June 2005; received in revised form 24 February 2006; accepted 6 March 2006
Available online 2 May 2006

Abstract

The identification of structural damage is an important objective of health monitoring for civil infrastructures. Frequently, damage to a structure may be reflected by a change of some system parameters, such as a degradation of the stiffness. In this paper, we propose an online sequential weighted Least Squares Support Vector Machine (LS-SVM) technique to identify the structural parameters and their changes when vibration data involve damage events. It efficiently updates a trained LS-SVM by means of incremental updating and decremental pruning algorithms whenever a sample is added to, or removed from, the training set, and robustness is improved by the use of an additional weighted LS-SVM step. This method overcomes the drawback of sparseness lost within the LS-SVM and makes LS-SVM for online system identification possible. The proposed method is capable of tracking abrupt or slow time changes of the system parameters from which the damage event and the severity of the structural damage can be detected and evaluated. Simulation results for tracking the parametric non-stationary changes of non-linear hysteretic structures are presented to demonstrate the application and effectiveness of the proposed technique in detecting the structural damage.

Keywords: Online; System identification; Structural health monitoring; Sequential weighted Least Squares Support Vector Machine

1. Introduction

In the field of civil engineering, real-time structural identification of dynamic systems subjected to earthquake motion has been focused on accurate prediction as well as structural health monitoring and damage assessment. System identification and damage detection based on measured vibration data have received intensive studies recently. As an online identification method, the Kalman filter (KF) has received much attention and has been successful in the parameter estimation problems over the past years [1–4]. Similar methods such as the least squares estimation method (LSE) [5–9], suboptimal H_∞ filter method [10], unscented Kalman filter [11], and particle filter [12] have been developed in some useful forms for solving many practical problems in civil engineering. To date, the online detection of the changes of structural parameters due to structural damage during a severe event, such as an earthquake, is still a challenging problem.

Recently, a least squares version of the SVM (LS-SVM) technique has received some attention for system estimation, function estimation, and non-linear system optimal control problems [13–16]. In the LS-SVM, Vapnik's ε-insensitive loss function has been replaced by a sum-squared error (SSE) cost function. Moreover, the LS-SVM considers equality type constraints instead of inequalities as in the classic SVM approach [17–19]. This reformulation greatly simplifies a problem such that the LS-SVM solution follows directly from solving a set of linear equations rather than from a convex quadratic program (QP).

Most existing algorithms for the SVM and LS-SVM require that training samples be delivered in a single batch; i.e., they are offline algorithms. Offline training algorithms, however, do not fit practical applications such as online system identification and control problems, in which the data are entered sequentially. The standard SVM (also called QP-based SVM, or QP-SVM for short) method generally has been used only for static problems, such as those of classification,

* Corresponding author.
E-mail address: thstj@mail.tongji.edu.cn (H.-S. Tang).

0141-0296/$ - see front matter
doi:10.1016/j.engstruct.2006.03.008

regression, and function estimation. As an application to such dynamic problems as online system identification and online control problems, the method is inefficient because each time the training set is modified, it must be retrained from scratch. A new online training model therefore is required each time a new sample is added to, or an existing sample is removed from, the training set. Approximate online training algorithms previously proposed for the SVM are incremental training algorithms [20] and sequential gradient methods [21], the main drawback of which is that the training process converges slowly. An improved iterative block training method for support vector classifiers based on weighted LSE (WLS-SVC for short) has been presented by Navia-Vázquez et al. [22] and Engel [23]. In [24], the author implemented an online adaptive selective Kernel learning algorithm for SVM using a window where training samples are selectively added and deleted.

Despite the computationally attractive features of LS-SVM algorithms, their solutions have potential drawbacks [15]. One is that sparseness is lost in the LS-SVM solutions. Another is that Vapnik's ε-insensitive loss function, instead of an SSE cost function, without regularization might lead to estimates that are less robust in the LS-SVM algorithms. Actually, a common assumption underlying most process modeling methods, including linear and non-linear least squares regression, is that variance is constant throughout the range of the measured variables. If this is not so, then using weighted least squares may yield the most precise parameter estimates possible. This is done by attempting to give each data point its proper amount of influence in parameter estimates.

In this paper, we propose a robust tracking technique, based on the online sequential weighted least squares support vector machine (SWLS-SVM for short) regression, to track the system parameters and their changes due to damage. The tracking algorithm is based on the adaptation of incremental and decremental pruning algorithms and an additional weighted LS-SVM step for the standard LS-SVM to update the parameter variations whenever a sample is added to, or removed from, the training set. The decremental pruning algorithm is based on sorted support values (SVs), by omitting a relative, small amount of the least meaningful support values. Such an adaptive tracking technique yields a sparse approximation and gives a larger importance to more recent data in order to cope with the system parameter's variations. The proposed technique is capable of tracking the non-stationary changes of system parameters from which the event and severity of structural damage may be detected online. In particular, the proposed method has an excellent capability to identify the abrupt or slow time change of system parameters relating to structural damage. Simulation results demonstrate that the proposed method is suitable for tracking the changes of system parameters for hysteretic structures.

The paper is organized as follows: Section 2 considers the weighted LS-SVM for function estimation. In Section 3 the SWLS-SVM is formulated. Section 4 discusses the identification of non-linear hysteric structural systems by the SWLS-SVM technique and some illustrative examples are presented. A conclusion is given in Section 5.

2. The batch weighted LS-SVM algorithm

Before presenting the weighted LS-SVM, basic formulation of the standard LS-SVM [19] for function estimation is briefly reviewed. Consider a given training set of N data points $\{(x_k, y_k)\}_{k=1}^{N}$ with the input $x_k \in R^n$ and output $y_k \in R$. The following regression model is used;

$$y(x) = \mathbf{w}^{\mathrm{T}} \cdot \boldsymbol{\varphi}(x) + b \tag{1}$$

where $\boldsymbol{\varphi}(*)$ maps the input data to a higher dimensional feature space, $\mathbf{w}$ is a weight vector, and b the bias. In the LS-SVM for function estimation, the objective function of the optimization problem, is defined as

$$\min_{w,b,e} J(w, e) = \frac{C}{2}\sum_{k=1}^{N} e_k^2 + \frac{1}{2}\|w\|^2 \tag{2}$$

subject to the constraints

$$y_k = \mathbf{w}^{\mathrm{T}}\boldsymbol{\varphi}(x_k) + b + e_k, \quad k = 1, 2, \ldots, N \tag{3}$$

where C is the user-defined regularization constant which balances the model's complexity and approximation accuracy, and e_k the approximation error.

Estimation of support values in the LS-SVM is optimal only when there is a Gaussian distribution of error variables. When, however, a Gaussian assumption for error variables is not realistic, it may lead to less robust estimates. This is because the SSE cost function of the LS-SVM, which assigns an equal weight to error at all times, treating all data equally, gives less precisely measured points more influence than they should have and highly precise points too little influence [15,19]. To obtain a robust estimate when the distribution is not a normal Gaussian one, a correction must be made by defining weights based on the error distribution; the so-called weighted LS-SVM method.

To modify these weights to obtain a robust estimate based on the previous LS-SVM solution, one weights the error variables, e_k, from the condition $\alpha_k = Ce_k$ by the weighting factors υ_k. This leads to a new optimization problem;

$$\min_{w,b,e} J^*(w, e) = \frac{C}{2}\sum_{k=1}^{N} \upsilon_k e_k^2 + \frac{1}{2}\|w\|^2. \tag{4}$$

The corresponding Lagrangian is given by

$$L(w, b, e, \alpha) = J^*(w, e) - \sum_{k=1}^{N} \alpha_k \left[\mathbf{w}^{\mathrm{T}}\boldsymbol{\varphi}(x_k) + b + e_k - y_k\right] \tag{5}$$

with Lagrange multipliers α_k. The Karush–Kuhn–Tucker (KKT) conditions for optimality [19] are given by

$$\begin{aligned}
&\frac{\partial L}{\partial \mathbf{w}} = 0 \rightarrow \mathbf{w} = \sum_{k=1}^{N} \alpha_k \boldsymbol{\varphi}(x_k) \\
&\frac{\partial L}{\partial e_k} = 0 \rightarrow \alpha_k = Ce_k \\
&\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{k=1}^{N} \alpha_k = 0 \\
&\frac{\partial L}{\partial \alpha_k} = 0 \rightarrow \mathbf{w}^{\mathrm{T}} \boldsymbol{\varphi}(x_k) + b + e_k - y_k = 0
\end{aligned} \tag{6}$$

for $k = 1, \ldots, N$. After elimination of e_k and w, the solution is given by the set of linear equations

$$\mathbf{A}_N \boldsymbol{\alpha}_N = \mathbf{Y}_N \tag{7}$$

where $\mathbf{A}_N = \begin{bmatrix} 0 & \vec{\mathbf{1}}^{\mathrm{T}} \\ \vec{\mathbf{1}} & \boldsymbol{\Omega} + \mathbf{V}_C \end{bmatrix}$, $\mathbf{V}_C = \mathrm{diag}\left\{\frac{1}{C\upsilon_1}, \ldots, \frac{1}{C\upsilon_N}\right\}$, $\boldsymbol{\alpha}_N = \begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix}$, $\mathbf{Y}_N = \begin{bmatrix} 0 \\ \mathbf{Y} \end{bmatrix}$, $\mathbf{Y} = [y_1; y_2; \ldots; y_N]$, $\vec{\mathbf{1}} = [1; 1; \ldots; 1]$, $\boldsymbol{\alpha} = [\alpha_1; \alpha_2; \ldots; \alpha_N]$, $\mathbf{I}$ is an $N \times N$ identity matrix, and $\boldsymbol{\Omega}$ follows Mercer's condition,

$$\Omega_{kl} = K(x_k, x_l) = \boldsymbol{\varphi}(x_k)^{\mathrm{T}} \boldsymbol{\varphi}(x_l), \quad k, l = 1, 2, \ldots, N \tag{8}$$

where Ω_{kl} stands for the item at the k-th row and l-th column of $\boldsymbol{\Omega}$.

Eqs. (1) and (6) provide the final result of the LS-SVM model for function estimation;

$$y(x) = \sum_{k=1}^{N} \alpha_k K(x_k, x) + b \tag{9}$$

where α and b are solutions of Eq. (7) and $\Phi(\cdot, \cdot)$ is the kernel function. As choices of kernel function, there are several possibilities. The kernel selected in this paper is a linear one based on the system identification procedure detailed in the next section.

The weighted LS reflects the behavior of the random errors in the model. Optimizing the weighted fitting criterion to obtain parameter estimates allows the weights to determine the contribution of each observation to the final parameter estimates. One common choice for υ_k has been given by David [25] and Suykens et al. [15]:

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

where $\hat{s} = \frac{\mathrm{IQR}}{2 \times 0.6745}$ is a robust estimate of the standard deviation of the LS-SVM error variable e_k, which denotes how much the estimated error distribution deviates from a Gaussian distribution. The IQR (interquartile range) is the difference between the 75th and 25th percentiles. The constants c_1, c_2 typically are chosen to be $c_1 = 2.5$ and $c_2 = 3$ [26]. Because the SSE cost function in the unweighted LS-SVR is optimal under the assumption of a normal Gaussian distribution for e_k, the weights-selecting criterion given by Eq. (10) provides corrections when the distribution is not normal, leading to a robust estimate.

3. The sequential weighted LS-SVM algorithm

The LSE-based WLS-SVC method [22,23] is the use of a constant-forgetting factor. This leads to online and adaptive implementations of SVC. The drawback of the LSE with constant-forgetting factor method for tracking time-varying parameters is that if the constant forgetting factor is small, it has a better capability of tracking the parametric variation, but it is very sensitive to measurement noise. On the other hand, if the constant forgetting factor used is large (approaches 1.0), its tracking capability is compromised although it is less sensitive to noise [9]. The variable forgetting factor approach [27] replaces the constant forgetting factor that depends on the time k. This approach improves over that of the constant forgetting factor. However, both approaches above can only recognize the time instant of parametric variation without knowing which parameter varies [8,9]. Consequently, when a parameter varies, the predicted results for all parameters exhibit significant oscillations. Hence, this approach works well for some cases but not for all, in particular when the parameters of the structure have an abrupt change. Here, a decremental pruning algorithm is adopted to select the "old" samples based on a sorted support vector (SV)-based criterion, and an incremental algorithm to update the trained LS-SVM whenever a new sample is added to the training set. The incremental updating and decremental pruning algorithms for sequential adaptive tracking technique are described in the following.

3.1. Incremental algorithm

The incremental algorithm updates the trained LS-SVM whenever a new sample, (x_{N+1}, y_{N+1}), is added to the training set $\left\{(x_k, y_k)\right\}_{k=1}^{N}$. In this section, sequential updating of LS-SVR is derived from the incremental updating algorithm.

Let (x_{N+1}, y_{N+1}) be a new training sample added to the first N data pairs $\left\{(x_k, y_k)\right\}_{k=1}^{N}$. From Eq. (7), the incremental relation between the current model (N data pairs) and the next new model (new $N+1$ data pairs) is given by

$$\mathbf{A}_{N+1} \boldsymbol{\alpha}_{N+1} = \mathbf{Y}_{N+1} \tag{11}$$

where

$$\mathbf{A}_{N+1} = \begin{bmatrix} \mathbf{A}_N & \mathbf{a} \\ \mathbf{a}^{\mathrm{T}} & c \end{bmatrix}, \quad \mathbf{Y}_{N+1} = \begin{bmatrix} \mathbf{Y}_N \\ y_{N+1} \end{bmatrix},$$

$$\boldsymbol{\alpha}_{N+1} = \begin{bmatrix} \boldsymbol{\alpha}_N \\ \alpha_{N+1} \end{bmatrix}, \quad \mathbf{A}_N = \begin{bmatrix} 0 & \vec{\mathbf{1}}^{\mathrm{T}} \\ \vec{\mathbf{1}} & \boldsymbol{\Omega} + C^{-1}\mathbf{I} \end{bmatrix},$$

$$\mathbf{Y}_N = \begin{bmatrix} 0 \\ \mathbf{Y} \end{bmatrix}, \quad \mathbf{a} = [1;\ \Phi(x_1, x_{N+1}); \ldots;\ \Phi(x_N, x_{N+1})]$$

$$\text{and} \quad c = C^{-1} + \Phi(x_{N+1}, x_{N+1}).$$

The online incremental training algorithm aims to efficiently update $\mathbf{A}_{N+1}^{-1}$ whenever a new sample is added without explicit computation of the matrix inverse. Following Golub and Van Loan [28] $\mathbf{A}_{N+1}^{-1}$ is obtained as;

$$\mathbf{A}_{N+1}^{-1} = \begin{bmatrix} \mathbf{A}_N^{-1} & \vec{0}^{\mathrm{T}} \\ \vec{0} & 0 \end{bmatrix} + \left[c - \mathbf{a}^{\mathrm{T}}\mathbf{A}_N^{-1}\mathbf{a}\right]^{-1} \begin{bmatrix} \mathbf{A}_N^{-1}\mathbf{a} \\ -1 \end{bmatrix} \times \left[\mathbf{a}^{\mathrm{T}}\mathbf{A}_N^{-1} \quad -1\right]. \tag{12}$$

A detailed derivation description of this incremental algorithm is provided in the Appendix. It is clear that updating $\mathbf{A}_{N+1}^{-1}$ in an incremental algorithm avoids an expensive inversion operation. The corresponding coefficients and bias therefore are obtained by Eq. (11).

3.2. Decremental algorithm

The incremental updating algorithm presented above can achieve a better steady-state performance in a stationary environment. However, the approach is not efficient for tracking non-stationary dynamics, because all the data in this algorithm are weighted equally, and the algorithm has an infinite memory length. Additionally, due to the condition $\alpha_k = Ce_k$ for optimality, the sparseness property in the LS-SVM is lost [15]. This method is inefficient when dealing with online problems with exceedingly large amounts of data because it is costly and there is not enough space to store the many coefficients.

Hence, we proposed a pruning procedure after the incremental updating step which is based on sorted SVs, by omitting a relatively small amount of the least meaningful support values [19,29]. By the adaptive pruning step, the SWLS-SVM is able to adapt to new scenarios, not only by incorporating new data, but also forgetting useless or out-of-date information. The proposed algorithm yields a sparse approximation and online implementation of incremental LS-SVM for non-stationary dynamics.

A decremental algorithm means that a SV is removed when a pair of training data is removed. Similar to the case of an incremental algorithm, to avoid computing the matrix inverse, $\mathbf{A}_N^{-1}$ must be updated from $\mathbf{A}_{N+1}^{-1}$. Here $\mathbf{A}_N^{-1}$ is the matrix without the k-th row and the k-th column. For the decremental way, when the k-th sample is pruned from the $N+1$ pairs of the data set, the update rule was obtained [30]

$$a_{ij} \leftarrow a_{ij} - a_{kk}^{-1} a_{ik} a_{kj} \tag{13}$$

where $i, j = 1, \ldots, N; i, j \neq k$, a_{ij} stands for the item at the i-th row and j-th column of $\mathbf{A}_{N+1}^{-1}$, and k stands for the k-th SV to be removed. According to Eq. (13), $\mathbf{A}_N^{-1}$ can be efficiently updated from $\mathbf{A}_{N+1}^{-1}$ without explicitly computing the matrix inverse. Then the coefficients of LS-SVM can be updated with Eq. (7).

The incremental and decremental algorithms for updating the LS-SVM, presented above, make online learning for the LS-SVM possible. Furthermore, a sparse LS-SVM solution is obtained by gradual decremental pruning of the sorted support vectors. The outline of the SWLS-SVM algorithm for online parameter estimation is as follows:

Algorithm: sequential weighted LS-SVM

1. Initialize. Set constant C. Set a threshold number of training data N_{thre}, $N = N_{\text{thre}}$.

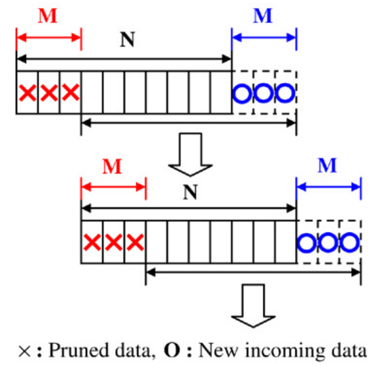


Fig. 1. Updating and pruning data for the on-line SWLS-SVM.

2. Given the training data $\{(x_k, y_k)\}_{k=1}^{N}$. Compute $e_k = \alpha_k / C$ by solving the linear systems (Eq. (7)).
3. Compute $\hat{s}$ from the distribution e_k.
4. Determine the weights υ_k based on $\hat{s}$, e_k.
5. Solve the weighted LS-SVR (Eq. (12)). Store $\mathbf{A}_N^{-1}$.
6. Input new training data (x_{N+1}, y_{N+1}). Compute $\mathbf{a}$ and c.
7. Update $\mathbf{A}_{N+1}^{-1}$. Update coefficients $\boldsymbol{\alpha}_{N+1}$.
8. Process the weighted LS-SVR (steps 2 to 4). Solving the weighted LS-SVM yields $\boldsymbol{\alpha}_{N+1}$. Store $\mathbf{A}_{N+1}^{-1}$.
9. Compute coefficients $\mathbf{w}$ (Eq. (6)).
10. If $N < N_{\text{thre}}$ go to step 6, otherwise go to step 11.
11. Sort the values $|\boldsymbol{\alpha}_{N+1}|$.
12. Process decremental LS-SVM (Eq. (17)). Remove some of the smallest M values in the sorted $|\boldsymbol{\alpha}_{N+1}|$. Retain the N-M points data, and set $N := N - M$ (see Fig. 1).
13. Return to step 6.

4. Numerical examples

Consider an m degree of freedom (DOF) non-linear hysteretic shear-type structure subject to ground excitation $\ddot{x}_g$; the equation of motion is

$$\mathbf{M}\ddot{x} + \mathbf{f}(\dot{x}, x) = -\mathbf{M}\{\mathbf{I}\}\ddot{x}_g \tag{14}$$

where $\mathbf{M}$ is the $m \times m$ mass matrix; x, $\dot{x}$, and $\ddot{x}$ are the relative displacement, velocity, and acceleration vector to the ground; $\{\mathbf{I}\}$ is the identity of the $m \times 1$ column matrix; and $\mathbf{f}$ the restoring force vector expressed by the Bouc–Wen model [31]. In this case, the i-th component of the vector is

$$\dot{f}_i = c_i \ddot{u}_i + k_i \dot{u}_i - \alpha_i |\dot{u}_i| |f_i|^{n_i - 1} f_i - \beta_i \dot{u}_i |f_i|^{n_i} \qquad i = 1, \ldots, m \tag{15}$$

where $\dot{u}_i = \dot{x}_i - \dot{x}_{i-1}$ and $\ddot{u}_i = \ddot{x}_i - \ddot{x}_{i-1}$ are the relative velocity and acceleration between the $(i-1)$-th and i-th mass point; and $c_i, k_i, \alpha_i, \beta_i$ and n_i are the damping, stiffness and the non-linear parameters of the i-th mass point.

In this study, it is assumed that only the ground and structural accelerations are available for identification. Structural displacements and velocities are obtained by integration of the corresponding accelerations by means of an integration scheme.

For notational simplicity, terms are labeled subscript 'k' to indicate the time step. Subscript 'i', which presents the i-th DOF, is omitted hereafter; e.g., f_k presents the value of the i-th DOF's restoring force at time k. The unknown parametric vector, $\mathbf{w}$, is defined as $\mathbf{w} = [c, k, \alpha, \beta]^{\mathrm{T}}$. To identify unknown parameters, the hysteretic equations of motion in Eqs. (14) and (15) must be discretized. Based on the third-order Predictor–Corrector method of Lin et al. [8], the incremental component of the restoring force relative to the i-th DOF at time k is

$$f_k = f_{k-1} + (\Delta t/12)(5\dot{f}_k + 8\dot{f}_{k-1} - \dot{f}_{k-2}) \quad (16)$$

where Δt represents the sampling time.

Following Eqs. (14)–(16), it is then possible to define the measurement y_k;

$$y_k = f_k - f_{k-1} \quad (17)$$

and the observation matrix $\mathbf{H}_k$;

$$\mathbf{H}_k = \Delta t \mathbf{h}_c \begin{bmatrix} \ddot{u}_k & \dot{u}_k & \dot{u}_k |f_k|^{n_i-1} f_k & \dot{u}_k |f_k|^{n_i} \\ \ddot{u}_{k-1} & \dot{u}_{k-1} & \dot{u}_{k-1} |f_{k-1}|^{n_i-1} f_{k-1} & \dot{u}_{k-1} |f_{k-1}|^{n_i} \\ \ddot{u}_{k-2} & \dot{u}_{k-2} & \dot{u}_{k-2} |f_{k-2}|^{n_i-1} f_{k-2} & \dot{u}_{k-2} |f_{k-2}|^{n_i} \end{bmatrix} \quad (18)$$

where $\mathbf{h}_c = \frac{1}{12}[5, 8, -1]$.

The system equation of the i-th DOF for the identification therefore is given by

$$y_k = \mathbf{w}^{\mathrm{T}} \mathbf{H}_k^{\mathrm{T}} + e_k \quad (19)$$

where e_k is the measurement noise.

Without loss of generality, consider the case of the i-th DOF for identification. The mass is assumed to be known, given a training set of data point pairs, $\{y_k, \mathbf{H}_k^{\mathrm{T}}\}_{k=1}^{N}$, where $y_k \in R$ is the output data and $\mathbf{H}_k^{\mathrm{T}} \in R^4$ the input data. The support vector method aims at constructing a function (Eq. (9)) to simulate the system function (Eq. (19)).

For simplicity, Eq. (19) is expressed in vector form;

$$\mathbf{H}_k^{\mathrm{T}} = [h_k^1, h_k^2, h_k^3, h_k^4]^{\mathrm{T}}. \quad (20)$$

The feature map selected for the weighted LS-SVM is

$$\boldsymbol{\varphi}(x_k) = [h_k^1, h_k^2, h_k^3, h_k^4]^{\mathrm{T}}. \quad (21)$$

Mercer's condition is applied to the matrix $\boldsymbol{\Omega}$, with $\Omega_{kl} = \boldsymbol{\varphi}(x_k)^{\mathrm{T}} \boldsymbol{\varphi}(x_l)$.

Hence, the function (Eq. (9)) is found by solving the linear set of Eqs. (7) and (21). Based on the KKT conditions,

$$\mathbf{w} = \sum_{k=1}^{N} \alpha_k \boldsymbol{\varphi}(x_k) \quad (22)$$

and from the support values α_k, one obtains the parameters of the structural system.

First, we consider a single DOF (SDOF) non-linear hysteretic structure subjected to ground excitation. The following parametric values are used in a simulation study: $m = 12.5$ kg, $c = 7$ kN s/m, $k = 25$ kN/m, and $\alpha = 3$, $\beta = 2$, $n = 2$. The Niigata, Japan (NS, 2004) earthquake record with the modified maximum amplitude of 25 cm/s^2 is the input excitation. The structural responses sampling interval is 0.01 s.

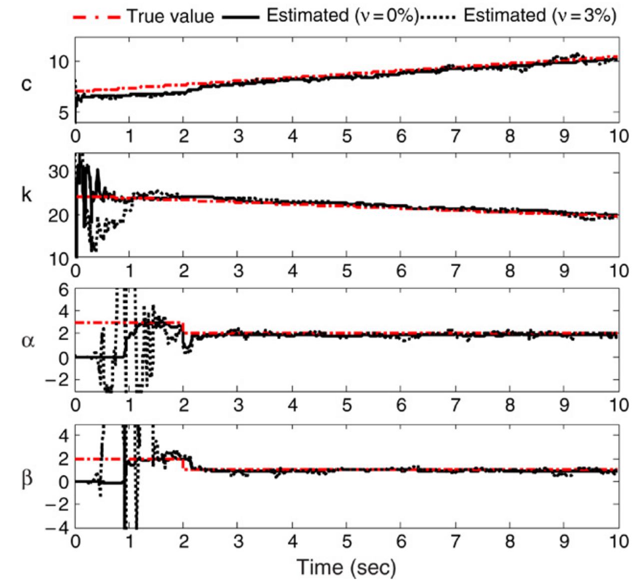


Fig. 2. Identified parameters c, k, α and β for an SDOF hysteretic structure with linearly changed stiffness and damping, and abruptly reduced α and β; k in kN/m and c in kN s/m.

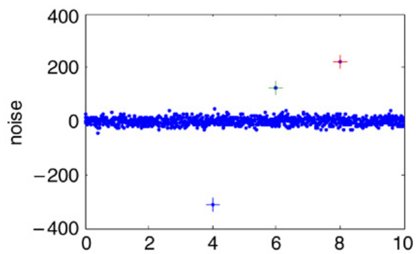
In this study, the effectiveness of the noise injection training is also investigated; noises are artificially added to structural responses in generating the training data. The noise level is defined as the value of the standard deviation. The Gaussian noise level is defined as

$$\upsilon = \frac{\sigma_{\text{noise}}}{\sigma_{\text{obser}}} \times 100\% \quad (23)$$

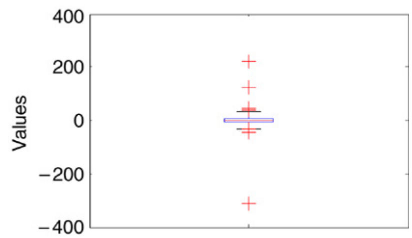
where σ_{noise} and σ_{obser} are the standard deviations of the added noise and the observation.

Throughout, the best parameter values (for C, ε) were found by using a cross-validation procedure in which we looked for the minimum average root-mean-squared error (RMSE) over a range of parameter values. The best values for each parameter were found for a single learning scenario in which the training set consisted of 1000 training samples corrupted by Gaussian noise with a level of 1%. This set of parameters was then used throughout the examples. Here, the best parameter values are $C = 15$ and $\varepsilon = 0.03$. Usually, it is convenient to use two samples for the starting data set. The end training set of data points given in this paper is $N = 200$, and the pruning parameters are $N_{\text{thre}} = 100$, $M = 10$. All examples (coded in Matlab) were run on a 256-MB, 2.4-GHz Pentium 4 Windows workstation.

To verify the time-varying tracking ability of the proposed technique, suppose stuffiness k and damping c are linearly changed with time. Namely, the k reduces from 25 kN/m to 20 kN/m, and the c increases from 7 kN s/m to 12 kN s/m. Abrupt changes also considered in this case are that both the α and β reduce abruptly from $\alpha = 3$ and the $\beta = 2$ to $\alpha = 2$ and $\beta = 1$ at time $t = 2$ s, respectively. Based on the proposed tracking technique, the identified results are shown in Fig. 1. Also shown in Fig. 2 as dot curves are estimation results with 3% noise level for comparison. It is observed from Fig. 2 that



(a) Time (sec).



(b) Boxplot.

Fig. 3. Zero mean Gaussian noise and 3 outliers (denoted by '+').

the proposed method tracks the structural parameters and their change online very well. As it can also be observed, when a parameter varies, the predicted results for all parameters don't exhibit significant oscillations.

We now compare the SWLS-SVM algorithm with the other various algorithms. One of them is the unweighted sequential LS-SVM (SLS-SVM), the second is QP-SVM and the third is the sequential minimal optimal optimization (SMO) algorithm [32]. We have used the proposed algorithm and we compare the results with those found by the other various algorithms for the above SDOF structure online estimation with different noise cases. To further verify the robustness of the weighted LS-SVM algorithm, a non-Gaussian noise case (Gaussian noise plus 3 outliers) is also considered. Given are 1000 training data points corrupted by zero mean Gaussian noise and 3 outliers (denoted by '+') (Fig. 3). The long, lower tail and plus signs in the box-plot of Fig. 3(b) denote the strong outliers in the sample values. Results have been collected in Table 1.

It can be observed in Table 1 that QP-SVM and SMO achieve the best estimation error (EE) on the system estimation. SWLS-SVM yields EE larger than QP-SVM, but with a computational cost dramatically smaller than QP-SVM. The results in Table 1 also show that both SWLS-SVM and SLS-SVM algorithms perform similarly at the Gaussian noise case. However, the SWLS-SVM yields to a slight improvement in training EE compared with the SLS-SVM when the injection noise is a non-Gaussian case.

To further explore the effectiveness of the SWLS-SVM algorithm, we consider a 3-story shear-beam building ($m = 3$) subject to the Niigata earthquake excitation identical to that given in the above SDOF example. In this building, all story units are the Bouc–Wen model in Eq. (34) is used for all storys. The properties of each story unit are: $m_1 = 18$ kg, $k_1 = 25$ kN/m, $c_1 = 7$ kN s/m, $\alpha_1 = 3$, $\beta_1 = 2$, $n_1 = 2$,

Table 1
Results of SDOF estimation

	QP-SVM	SMO	SLS-SVM	SWLS-SVM
CPU time (s)	675.37	283.62	9.48	13.85
EE (%) $\upsilon = 0\%$	0.32	0.44	1.2	1.2
EE (%) $\upsilon = 1\%$	0.75	0.97	3.2	3.1
EE (%) $\upsilon = 2\%$	1.33	1.34	4.7	4.5
EE (%) $\upsilon = 3\%$	1.72	2.15	5.6	5.2
EE (%) $\upsilon = 3\% +$ outliers	2.31	2.79	8.3	6.5

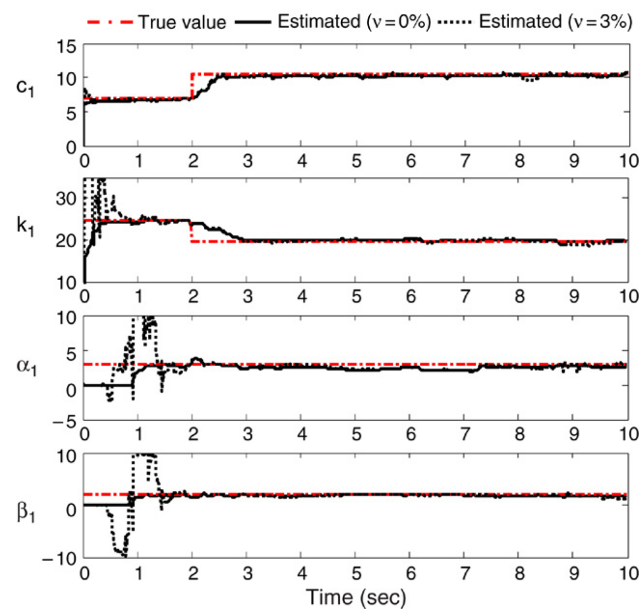


Fig. 4. Identified parameters c, k, α and β for a 3-story non-linear hysteretic structure (1st story); k in kN/m and c in kN s/m.

$m_2 = 12.5$, $k_2 = 2k_1/3$, $c_2 = 5$ kN s/m, $\alpha_2 = 3$, $\beta_2 = 2$, $n_2 = 2$, $m_3 = 12.5$ kg, $k_3 = 1k_1/3$, $c_3 = 4$ kN s/m, $\alpha_3 = 3$, $\beta_3 = 2$, $n_3 = 2$. In a similar manner as the SDOF cases, the hysteretic equations of motion can be established and the data matrix can be constructed. Unknown parametric vectors will consist of c_i, k_i, α_i, and β_i ($i = 1; 2; 3$). Suppose a damage just occurs in the 1st story unit at $t = 2$ s, at which time the stiffness in the first story unit k_1 reduces abruptly from 25 to 20 kN/m, and the damping c_1 increases abruptly from 7 to 10 kN s/m.

Based on the proposed tracking technique, the identified results with different noise levels for comparison are presented in Figs. 4–6. It is observed from Figs. 4–6 that the proposed method tracks the structural parameters and their variations very well. Also shown in these figures, when a parameter varies, the estimated results for the other parameters don't exhibit significant oscillations It should be mentioned that the results based on the constant-forgetting factor and other approaches are not satisfactory.

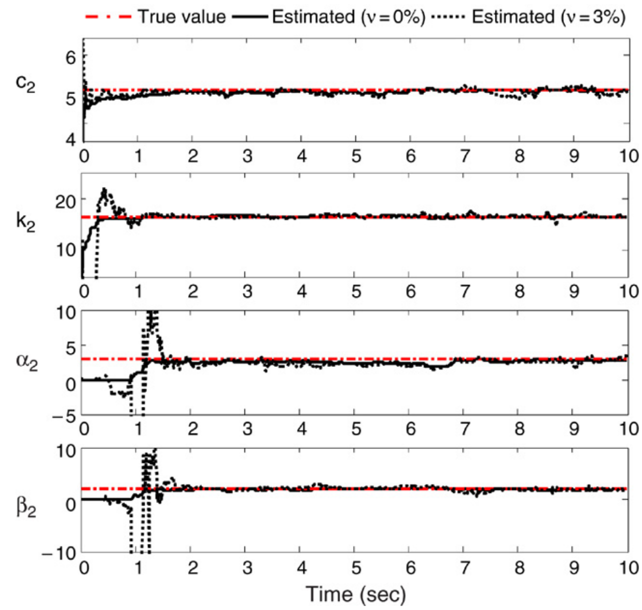


Fig. 5. Identified parameters c, k, α and β for a 3-story non-linear hysteretic structure (2nd story); k in kN/m and c in kN s/m.

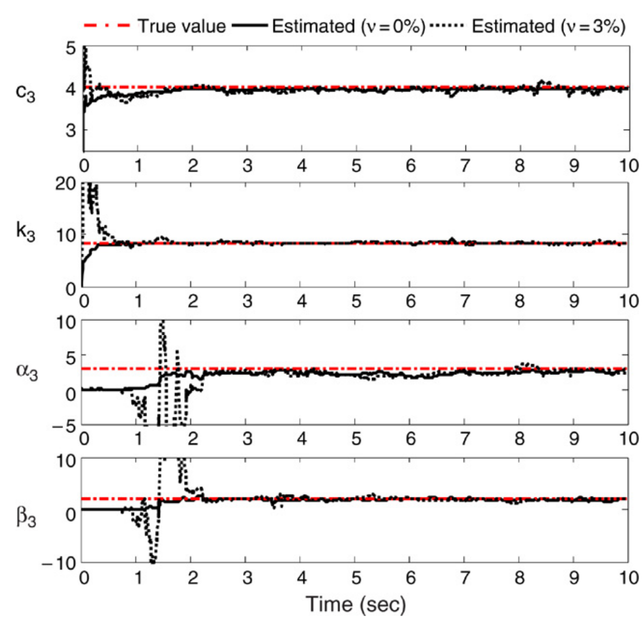


Fig. 6. Identified parameters c, k, α and β for a 3-story non-linear hysteretic structure (3rd story); k in kN/m and c in kN s/m.

5. Conclusions

Based on the incremental updating and decremental pruning algorithms, a sequential least squares support vector machine regression tracking technique has been proposed to identify online the structural parameters and their variations due to damage for non-linear hysteretic structures. A more robust estimate is obtained when the weighted LS-SVM is used. The effectiveness of the proposed technique has been demonstrated using the simulation results for SDOF and multi-DOF non-linear hysteretic structures. Numerical results indicate that the proposed approach is particularly suitable for tracking the abrupt or slow time changes of system parameters from which the structural damage can be determined. Results also show the low computation cost of this method.

Appendix. Derivation of the incremental updating

After sub-matrices computations [28], data matrix $\mathbf{A}$ is partitioned into the following sub-matrices;

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}. \tag{A1}$$

If $\mathbf{A}_{11}^{-1}$ and $\mathbf{A}_{22}^{-1}$ exist, then the matrix $\mathbf{A}$ inverse is

$$\mathbf{A}^{-1} = \begin{bmatrix} \left[\mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21}\right]^{-1} & \mathbf{A}_{11}^{-1}\mathbf{A}_{12}\left[\mathbf{A}_{21}\mathbf{A}_{22}^{-1}\mathbf{A}_{12} - \mathbf{A}_{22}\right]^{-1} \\ \left[\mathbf{A}_{22}^{-1}\mathbf{A}_{12} - \mathbf{A}_{22}\right]^{-1}\mathbf{A}_{21}\mathbf{A}_{21}^{-1} & \left[\mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\right]^{-1} \end{bmatrix}. \tag{A2}$$

From Eqs. (A1) and (A2), matrix $\mathbf{A}_{N+1} = \begin{bmatrix} \mathbf{A_N} & \mathbf{a} \\ \mathbf{a}^{\mathrm{T}} & c \end{bmatrix}$ inverse leads to

$$\mathbf{A}_{N+1}^{-1} = \begin{bmatrix} \left[\mathbf{A}_N - c^{-1}\mathbf{a}\mathbf{a}^{\mathrm{T}}\right]^{-1} & \mathbf{A}_N^{-1}\mathbf{a}\left[\mathbf{a}^{\mathrm{T}}\mathbf{A}_N^{-1}\mathbf{a} - c\right]^{-1} \\ \left[\mathbf{a}^{\mathrm{T}}\mathbf{A}_N^{-1}\mathbf{a} - c\right]^{-1}\mathbf{a}^{\mathrm{T}}\mathbf{A}_N^{-1} & \left[c - \mathbf{a}^{\mathrm{T}}\mathbf{A}_N^{-1}\mathbf{a}\right]^{-1} \end{bmatrix}^{-1}. \tag{A3}$$

A matrix inverse lemma $(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{C} + \mathbf{DA}^{-1}\mathbf{B})^{-1}\mathbf{DA}^{-1}$, gives

$$\left[\mathbf{A}_N - c^{-1}\mathbf{a}\mathbf{a}^{\mathrm{T}}\right]^{-1} = \mathbf{A}_N^{-1} - \mathbf{A}_N^{-1}\mathbf{a}\left[-c + \mathbf{a}^{\mathrm{T}}\mathbf{A}_N^{-1}\mathbf{a}\right]^{-1}\mathbf{a}^{\mathrm{T}}\mathbf{A}_N^{-1}. \tag{A4}$$

From Eqs. (A3) and (A4) Eq. (12) is obtained.

References

[1] Shinozuka M, Yun C, Imai H. Identification of linear structural dynamic systems. ASCE J Eng Mech 1982;108(6):1371–90.

[2] Hoshiya M, Saito E. Structural identification by extended Kalman filter. ASCE J Eng Mech 1984;110(12):1757–71.

[3] Shinozuka M, Ghanem R. Structural system identification. II: Experimental verification. ASCE J Eng Mech 1995;121(2):265–73.

[4] Maruyama O, Hoshiya M. System identification of an experimental model by extended Kalman filter. In: Proceedings on structural safety and reliability. Lisse: Swets & Zeitinger; 2002. 8 pages, http://www.balkema.nl; 2001. CD-ROM.

[5] Loh CH, Tou IC. A system identification approach to the detection of changes in both linear and nonlinear structural parameters. J Earthq Eng Struct Dyn 1995;24(1):85–97.

[6] Loh CH, Lin CY, Huang CC. Time domain identification of frames under earthquake loadings. ASCE J Eng Mech 2000;126(7):693–703.

[7] Lus H, Betti R, Longman RW. Identification of linear structural systems using earthquake-induced vibration data earthquake. J Earthq Eng Struct Dyn 1999;28:1449–67.

[8] Lin JW, Raimondo B, Smyth AW, Longman RW. Online identification of nonlinear hysteretic structural systems using a variable trace approach. J Earthq Eng Struct Dyn 2001;30:1279–303.

[9] Yang JN, Lin S. On-line identification of non-linear hysteretic structures using an adaptive tracking technique. Int J Non-Linear Mech 2004;39(11): 1481–91.

[10] Sato T, Qi K. Adaptive H_∞ filter: its application to structural identification. ASCE J Eng Mech 1998;124(11):1233–40.

[11] Tang H, Sato T. SVD-based derivative-free Kalman filter for nonlinear

hysteretic structural system identification. In: 9th international conference on structural safety and reliability. 2005.
[12] Tang H, Sato T. Auxiliary particle filtering for structural system identification. In: SPIE's 12th symposium on smart structures and materials. 2005.
[13] Rojo-Álvarez JL, Martínez-Ramón M, de Prado-Cumplido M, Artés-Rodríguez A, Figueiras-Vidal AR. Support vector method for robust ARMA system identification. IEEE Trans Signal Process 2004;52(1): 155–64.
[14] Suykens JAK, Vandewalle J, De Moor B. Optimal control by least squares support vector machines. Neural Netw 2001;14(1):23–35.
[15] Suykens JAK, De Brabanter J, Lukas L, Vandewalle J. Weighted least squares support vector machines: robustness and sparse approximation. Neurocomputing 2002;48(1–4):85–105.
[16] Suykens JAK. Support vector machines: a nonlinear modelling and control perspective. In: Fundamental issues in control. Eur J Control 2001; 7(2–3):311–27 [special issue].
[17] Vapnik V. The nature of statistical learning theory. New York: Springer-Verlag; 1995.
[18] Vapnik V. Statistical learning theory. New York: John Wiley & Sons; 1998.
[19] Suykens JAK, Van Gestel T, De Brabanter J, De Moor B, Vandewalle J. Least squares support vector machines. Singapore: World Scientific; 2002.
[20] Ralaivola L, d'Alche-Buc F. Incremental support vector machine learning: A local approach. In: Dorffner G, Bischof H, Hornik K, editors. Artificial neural networks—ICANN 2001. Berlin: Springer-Verlag; 2001. p. 322–30.
[21] Vijayakumar S, Wu S. A gradient based technique for generating sparse representation in function approximation. In: 6th international conference on neural information processing. 1999.
[22] Navia-Vázquez A, Pérez-Cruz F, Artés-Rodríguez A, Figueiras-Vidal AR. Weighted least squares training of support vector classifiers leading to compact and adaptive schemes. IEEE Trans Neural Netw 2001;12(5): 1047–59.
[23] Engel Y, Mannor S. The kernel recursive least-squares algorithm. IEEE Trans Signal Process 2004;52(8):2275–85.
[24] Kuh A. Adaptive Kernel Methods for CDMA Systems. In: Proc. of the international joint conference on neural networks. 2001. p. 1404–9.
[25] David HA. Early sample measures of variability. Statist Sci 1998;13(4): 368–77.
[26] Rousseeuw PJ, Leroy A. Robust regression and outlier detection. New York: Wiley; 1987.
[27] Astrom KJ. Self-tunning regulators-design principles and applications. In: Narendra KS, Monopoli RV, editors. Applications of adaptive control. New York: Academic Press; 1980.
[28] Golub GH, Van Loan CF. Matrix computations. Baltimore (MD): Johns Hopkins University Press; 1989.
[29] Suykens JAK, Lukas L, Vandewalle J. Sparse approximation using least squares support vector machines. In: Proc. of the IEEE international symposium on circuits and systems. 2000. p. 757–60.
[30] Cauwenberghs G, Poggio T. Incremental and decremental support vector machine learning. In: Leen TK, Dietterich TG, Tresp V, editors. Advances in neural information processing systems, vol. 13. Cambridge (MA): MIT Press; 2001. p. 409–15.
[31] Wen YK. Method for random vibration of inelastic structures. App Mech Rev 1989;42(2):39–52.
[32] Platt J. Fast training of support vector machines using sequential minimal optimization. In: Scholkopf B, Burges CJC, Smola AJ, editors. Advances in kernel methods: support vector learning. Cambridge (MA): MIT Press; 1999. p. 185–208.