Korean J. Chem. Eng., 32(6), 1029-1036 (2015)
DOI: 10.1007/s11814-014-0301-2

pISSN: 0256-1115
eISSN: 1975-7220

A real-time model based on least squares support vector machines and output bias update for the prediction of NO_x emission from coal-fired power plant

Faisal Ahmed, Hyun Jun Cho, Jin Kuk Kim, Noh Uk Seong, and Yeong Koo Yeo†

Department of Chemical Engineering, Hanyang University, Haengdang-dong, Sungdong-gu, Seoul 139-791, Korea
(*Received 29 June 2014 • accepted 5 October 2014*)

Abstract−The accurate and reliable real-time estimation of NOx emission is indispensable for the implementation of successful control and optimization of NOx emission from a coal-fired power plant. We apply a real-time update scheme to least squares support vector machines (LSSVM) to build a real-time version for real-time prediction of NOx. Incorporation of LSSVM in the update scheme enhances its generalization ability for long-term predictions. The proposed real-time model based on LSSVM (LSSVM-scheme) is applied to NOx emission process data from a coal-fired power plant in Korea to compare the prediction performance of NOx emission with real-time model based on partial least squares (PLS-scheme). Prediction results show that LSSVM-scheme predicts robustly for a long passage of time with higher accuracy in comparison with PLS-scheme. We also present a user friendly and sophisticated graphical user interface to enhance the convenience to approach the features of real-time LSSVM-scheme.

Keywords: NO_x Prediction, Real-time Model, Least Squares Support Vector Machine, Partial Least Squares, Output Bias Update

INTRODUCTION

Most power stations burn fossil fuels such as coal, oil, and natural gas for electricity production. Among these, coal is the most widespread fuel used because of its low cost and availability. Therefore, it is important to develop techniques to address the underlying issues arising from the utilization of pulverized coal in utility boilers to assist the power plant designers and operators to minimize harmful emissions from the stack and run the operation cleanly and efficiently [1]. During the coal combustion process, oxides of nitrogen (NOx) are major pollutants. The indirect enhancement of the greenhouse effect, depletion of stratospheric ozone, photochemical smog and acid rain are some of the adverse effects of NOx. In coal-fired power plants, NOx emission should be measured accurately and reliably and reduced in order to ensure the compliance of plant emissions with stringent emission limits imposed by government environmental bodies, while keeping the operation economically optimized and secure. The majority of coal-fired power plants use the hardware-based continuous emission monitoring system (CEMS) to measure the NOx emissions. Although, the credibility of measurements obtained from CEMS is satisfactory because of its accuracy and reliability. However, its cost, installation and maintenance are expensive. Moreover, a harsh environment at the stack causes the CEMS to be offline frequently for maintenance [2]. Therefore, many researchers have focused on soft sensors, which (1) can replace the CEMS, (2) may be used in parallel with it to provide redundancy to verify if the hardware sensor prediction is deviating, or (3) may be used to improve the control performance of de-NOx device [3].

Standard modeling approaches consist of two main categories: (1) the first principle model, based on physical and chemical laws, which are very difficult or in some cases impossible to obtain due to the complexity of the chemical processes and are time consuming; (2) black-box models [4] are obtained through identification from process variables data history. Although the first principle models provide physical insight into the process, it is very difficult to directly relate mathematically the process variables with the quality variable and an alternative must be sought [5]; soft sensors based on black-box models have been widely employed as a substitute as they require less specific knowledge of the process and use data history of process variables to predict the quality variables of interest [6].

Generally, NOx emission from coal-fired power plants exhibits typical nonlinear behavior. To capture this nonlinearity, many nonlinear techniques have been proposed for the prediction of NOx emissions such as partial least squares (PLS) [7], autoregressive exogenous model (ARX), nonlinear autoregressive exogenous model (NARX), grey box modeling [5] and artificial neural networks [1,8]. NOx emission from coal-fired power plants has also been predicted and reduced by utilizing support vector framework, e.g., support vector regression (SVR) [9], least squares support vector regression (LSSVR) [2], combination of SVR and kernel principal component analysis (kPCA) [10], and least squares support vector machine (LSSVM) [6].

In recent years, SVM and its modification LSSVM, have gained great attention in the field of machine learning as a method for classification and nonlinear function estimation [11,12]. This is due to its better generalization ability and global optimization property over other machine learning methods, such as partial least squares (PLS) and artificial neural network (ANN). A real-time version of

†To whom correspondence should be addressed.
E-mail: ykyeo@hanyang.ac.kr

LSSVM has been applied to many engineering fields [13,14] and comparison of offline LSSVM with offline PLS in different areas of study has been conducted [15-17]. However, not much research effort has been done so far for NOx emission prediction using real-time LSSVM and the comparison of real-time PLS with real-time LSSVM. This motivates the novel development of a model with the incorporation of LSSVM in real-time update scheme to predict NOx emission and compare it with the real-time version of PLS. This paper presents incorporation of LSSVM in a real-time update scheme to develop a novel model to predict the NOx emissions based on the values of process variables history. The proposed real-time scheme called LSSVM-scheme is successfully applied to NOx emissions data of Taean Power Plant, Korea for the real-time prediction of NOx emissions. The proposed real-time LSSVM-scheme is compared with PLS-scheme developed by Ahmed et al. in 2009 [18].

STATISTICAL METHODS

1. Least Square Support Vector Machine (LSSVM)

Support vector machines (SVM) have been effectively applied in modeling for their high generalization ability and global optimization property [19]. The SVM formulation of the learning problem leads to quadratic programming (QP) with linear constraints. Suykens and Vandewalle proposed a modified version of SVM called least squares support vector machines (LSSVM) in 1999 [12] to reduce the complexity of optimization process of SVM. LSSVM adopts least squares loss function and equality constraints instead of e-insensitive loss function and inequality constraints in SVM. LSSVM can obtain the analytical solution of a set of linear equations rather than by solving a quadratic programming problem. The formulation of LSSVM is introduced as following.

Consider a given training set containing N data points $\{x_k, y_k\}$, (k=1, 2, …, N), with input $x_k \in R^n$ and output $y_k \in R^n$. The following regression model can be obtained by using a nonlinear mapping function, $\varphi(\mathbf{x})$ which maps the input vector into a high dimensional feature space.

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

where b is output bias and **w** is a weight vector of the same dimension as the feature space. As in SVM, it is necessary to minimize a cost function C containing a penalized regression error which forms an optimization problem as follows:

$$\mathrm{Min} J(\mathbf{w}, \mathbf{e}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{1}{2}C\Sigma_{i=1}^{N} e_i^2 \tag{2}$$

subject to the equality constraints

$$y_i = \mathbf{w}^T\varphi(x_i) + b + e_i, \qquad (i=1, 2, \ldots, N) \tag{3}$$

where y_i is the output, x_i is the regression vector, e_i is the error between actual and predictive output at instant i, and C represents punishment on the data set beyond the pre-specified error tolerance. To solve this optimization problem, the Lagrange function is constructed as:

$$L(\mathbf{w}, b, \mathbf{e}; \alpha) = J(\mathbf{w}, e) - \Sigma_{i=1}^{N}\alpha_i\{\mathbf{w}^T\varphi(x_i) + b + e_i - y_i\} \tag{4}$$

where α_i, (i=1, 2, …, N) are Lagrange multipliers. Considering Karush-Kuhn-Tucker (KKT) optimality conditions, we have

$$\frac{\partial L}{\partial w} = 0, \frac{\partial L}{\partial b} = 0, \frac{\partial L}{\partial e_k} = 0, \frac{\partial L}{\partial \alpha_k} = 0 \tag{5}$$

The optimization problem can be written as:

$$\begin{bmatrix} 0 & \mathbf{1} \\ \mathbf{1}^T & \Omega + \frac{1}{C}\mathbf{I} \end{bmatrix} \cdot \begin{bmatrix} b \\ \alpha^T \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{y}^T \end{bmatrix} \tag{6}$$

where $\mathbf{y}=[y_1, \ldots y_N]$, $\mathbf{1}=[1, \ldots 1]$, $\alpha=[\alpha_1, \ldots \alpha_N]$ and Mercer condition is applied as:

$$\Omega_{ij} = K(x_i, x_j) = \varphi(x_i)^T\varphi(x_j), \quad (i, j=1, 2 \ldots, N) \tag{7}$$

Investigations show that RBF kernel function is superior over other kernel functions when there is a lack of prior knowledge [11]. The RBF kernel that satisfies the Mercer conditions is given below:

$$K(x, x_i) = \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right), \quad (i=1, 2 \ldots, N) \tag{8}$$

where σ is the RBF width. The resulting LSSVM model for nonlinear regression can be written as:

$$y(x) = \Sigma_{i=1}^{N}\alpha_i K(x, x_i) + b \tag{9}$$

2. LSSVM Update

Many processes are time-varying, due to which magnitudes of process variables vary with time. It renders addition of new and removal of oldest measurements or data points to the training data set crucial to calibrate the model in a real-time fashion. This change in training data set requires a continuous update of model parameters to deal with the time-varying impact of the processes. The recursive PLS model was developed by Helland et al. [20] by employing recursive update of the training data set simultaneous removal of the oldest data samples [20]. It continuously updates the model with the effects of new events while retaining the process history partially. This motivates us to recursively update the parameters for the LSSVM framework for real-time prediction of NOx for a long passage of time.

Dependence of LSSVM on how the training data set is scaled makes it essential for the process and response variables to be symmetrically transformed to give equal weight to each process variable in a real-time manner. A common approach is mean centering and scaling to unit variance when this relative correlation between process variables is unknown. We used mean centering and unit variance for data scaling, which can be computed as follows:

$$x_{i,ms} = \frac{x_i - m}{s} \tag{10}$$

where $x_{i,ms}$ is the transformed value of x_i, (i=1, 2, …N), and m and s represent the mean and the standard deviation of the corresponding variable respectively. Mean and variance are also updated for the scaling of updated training data set at each arrival of calibration measurements. The updating method is as follows [21]:

$$m_{h+1}=\frac{N-1}{N}m_h+\frac{1}{N}x_{h+1} \tag{11}$$

$$s^2_{h+1}=\frac{N-2}{N-1}s^2_h+\frac{1}{N-1}(x_{h+1}-m_{h+1})^2 \tag{12}$$

where m_h and s_h are the mean and the variance of training data at the h^{th} addition of the calibration measurement respectively, and m_{h+1} and s_{h+1} represent the corresponding values at $(h+1)^{th}$ addition.

3. Output Bias Update

One of the important components of a real-time model is model bias update, as deviation in process environment with time may change some of the process variables. This drift in process variables may deteriorate the reliability of the prediction of the developed model [22]. To model these uncertainties, one of the common adaptive procedures is output bias update, which is incorporated in real-time statistical models to ensure reliability, robust adaptation and sensitivity towards time variant features of the process environment [23]. Output bias at time step (t) is calculated as the difference between predicted value at the previous time step (t−1) and corresponding calibration measurement sampled at previous time step (t−1) as shown in Eq. (13).

(at $t=t^{th}$ run):

$$\text{bias}(t)=Y_{cal}(t-1)-Y_{pred}(t-1), \tag{13}$$

where Y_{cal} and Y_{pred} are the calibration measurement and the predicted values, respectively, and bias(t) represents the output bias at the t^{th} run. The predicted values are modified by output bias which is added to it to give final prediction of quality variable as shown in following equation:

$$Y_{mod}(t)=Y_{pred}(t)+\text{bias}(t) \tag{14}$$

where Y_{mod} are the modified values of Y_{pred} and bias(0)=0.

4. Chance Correlation

A prediction model developed through various statistical techniques might be a result of a chance correlation. In other words, process variables might be related to the quality variable just by chance, and only random values of process variables and/or quality variable might give a better fit of the model. This point appears to be generally ignored while validating the statistical models. A statistical model can be characterized as significant if at least the model is reliable and robust for future predictions and is able to fit the given data better than chance would do. Y-scrambling or Y-randomization is said to be the most prevailing validation method among all to verify whether the model is reliable, i.e., it does not exhibit correlation just by chance [24].

To validate the model fitting, the relation between quality variable and process variables is intentionally destroyed. The data samples of process and/or quality variable(s) are randomly permuted or replaced at all with random numbers. In this way, various pseudo models are built with pseudo variables. Then R^2 values are computed and compared to check the statistical significance of the original model against variants of pseudo models developed through Y-randomization. For chance correlation four variants were used in this study, the details of which can be found elsewhere [25].

Variant 1: Randomly permuted quality variable vs. original process variables (PY vs. X)

Variant 2: Randomly permuted quality variable vs. random numbers replacing original process variables (PY vs. RX)

Variant 3: Random numbers replacing quality variable vs. original process variables (RY vs. X)

Variant 4: Random numbers replacing quality variable vs. random numbers replacing original process variables (RY vs. RX)

PROPOSED REAL-TIME LSSVM-SCHEME

In several industrial processes, offline measurements of quality variables are available after a certain interval. These values can be used for calibration of the model to make it reliable and consistent for long-term predictions. The proposed scheme builds an initial model through LSSVM using process variables as the data set. This initial LSSVM model predicts every minute until a calibration measurement arrives. As soon as a calibration measurement is available, it is added to the training data set and the oldest measurement is removed. The scheme then activates either of the two update methods (LSSVM update or output bias update) and generates predictions until new calibration measurement is available. The algorithm of this scheme identifies the slow and rapid changes of the output

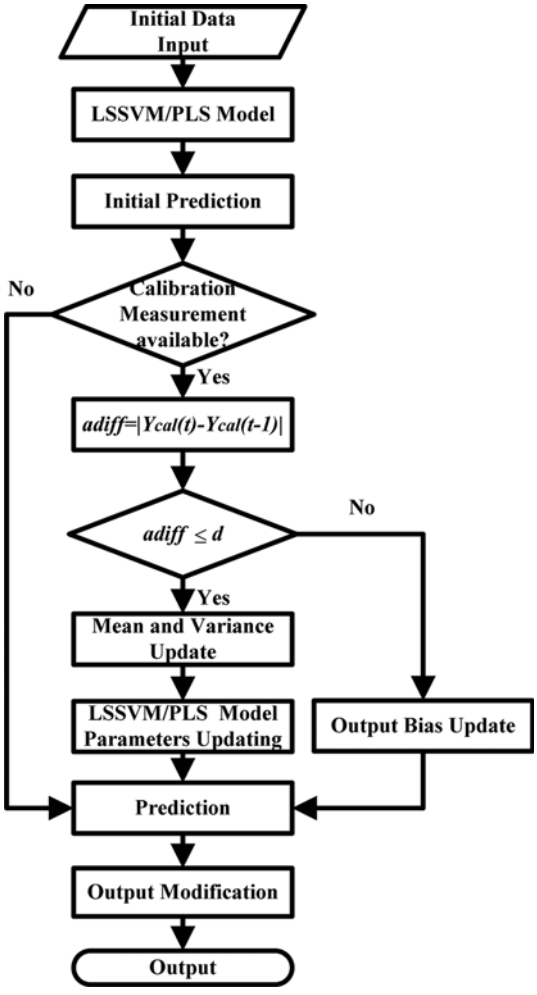

Fig. 1. Flow diagram of the proposed scheme.

variable behavior and switches to LSSVM and output bias update, respectively. The selection of the update method is based on threshold constant d, which is the crucial parameter of the proposed scheme. The threshold constant decides the suitable and efficient updating method for the future predictions at a specific instance. This constant acts as a switch between LSSVM update and output bias update; it chooses the LSSVM update when the absolute difference (*adiff*) of current and previous calibration measurements is less than d as shown in Fig. 1; otherwise, the output bias update is selected for updating the model. Predictions from both update methods are modified by the bias term by using Eq. (14) and finally provide output of the model. The flow of algorithm is demonstrated in Fig. 1.

A graphical user interface (GUI) named StatPredLSSVM, is developed for accessibility and convenience approach to key features of real-time model adaptation based on the LSSVM parameter update and output bias update. The GUI mainly consists of two sections. The first section accepts basic inputs and has features to load and unload the data, while the second section has the features to optimize the parameters, display optimization results, predict and display the prediction results (in this case, NOx prediction). A snapshot with the results of an example execution on StatPredLSSVM is shown in Fig. 2.

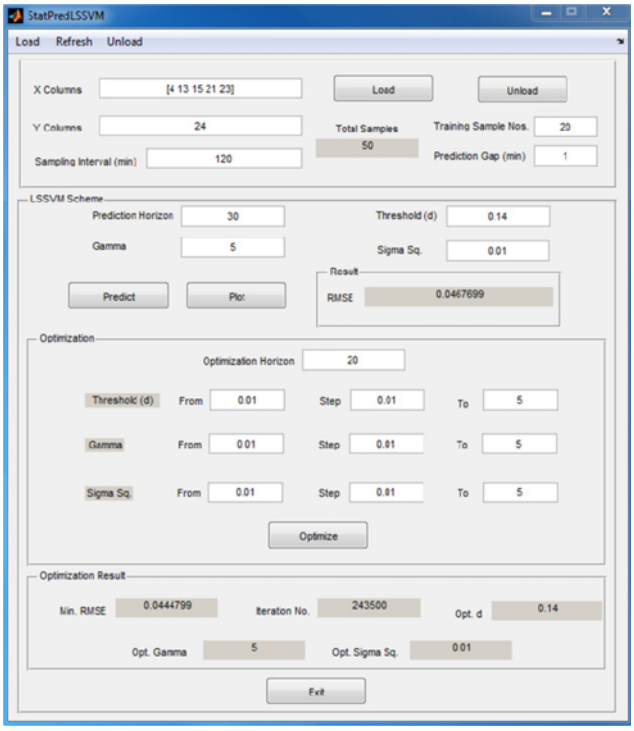

Fig. 2. A graphical user interface for LSSVM-scheme.

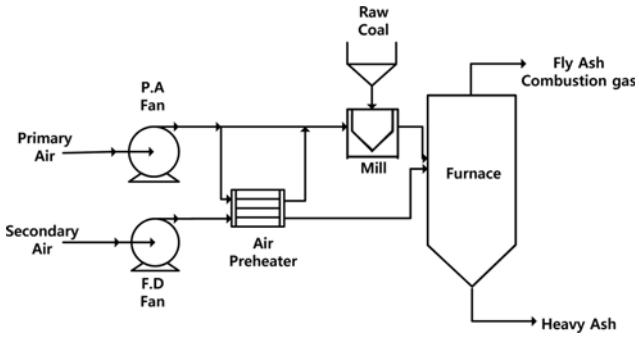

Fig. 3. Simplified process flow diagram of tangentially fired pulverized-coal boiler.

APPLICATION TO NOx EMISSIONS

1. Process Description

We considered a standard 500 MW power plant with tangentially fired pulverized-coal boiler located in Taean, Korea. A simplified process flow diagram is shown in Fig. 3. The boiler consists of six layers and 24 burners. Stacked coal in silos is conveyed through a conveyer belt to the mill where size reduction of coal is carried out. The amount of coal used depends on the boiler load. Primary air is preheated in a preheater and enters the mill where it transports the pulverized coal to the boiler. Secondary air travels through air preheater to the boiler by FD fan and is used in the furnace to burn the coal. Coal combustion produces combustion gas and ash. Heavy ash is collected at the bottom and fly ash at the top, respectively. NOx is then removed by selective catalytic reduction (SCR) and the remaining part is fed to a dust collector where fly ash is captured. Finally, SOx is removed followed by emission of remaining gases through stack.

2. Numerical Simulations

The data consisted of 43 process variables, which included coal feed rate, primary air flow bias, mill bowl differential pressure, total moisture, fuel ratio, etc. A graphical investigation was conducted and some variables were omitted that were found to be almost constant or noise along the whole passage of time. Preliminary investigation left the variable selection process with 23 variables at hand. Among these variables, it was crucial to eliminate the variables which were exhibiting the same varying pattern. To get rid of these redundant variables, the genetic algorithm (GA) in PLS_Toolbox (Eigenvector Research Inc.) was used to select five process variables from the remaining 23 variables. The time lag between input process variables and output response variable was found to be 11 minutes, which was accordingly incorporated in the data. Samples giving erroneous and far from real life values according to the plant were considered outliers and were eliminated by examining the NOx emission, and the input data were obtained corresponding to instances of the quality variable (NOx).

Eventually, 160 data samples were collected with a sampling time of 2 hours. Thirty samples were assigned for initial LSSVM model, and 30 samples were used for LSSVM parameter optimization of the scheme from which one data sample was used for the prediction to initialize optimization. The remaining 100 samples were used for testing and validating the model.

The initial LSSVM model was built with 30 data samples and prediction was carried out for one data sample to provide a base model. The model parameters gamma, sigma and threshold d for the scheme were then optimized with 29 data samples.

After the LSSVM parameters were optimized, the LSSVM-scheme was then initiated to predict NOx emission. LSSVM-scheme pre-

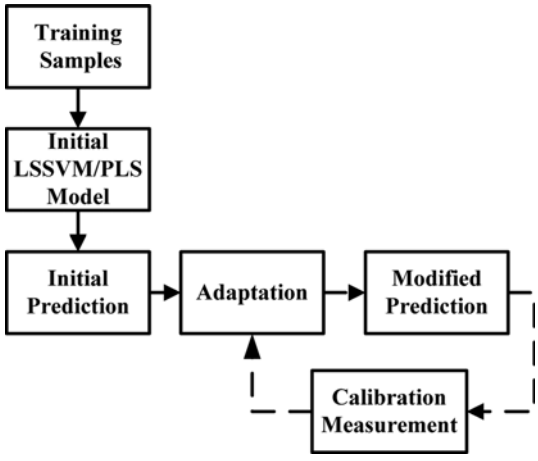

Fig. 4. Initialization and flow of sequential arrangement of model adaptation and predictions.

dicts for 2 hours on per minute basis until new calibration measurement is supplied to the training data set. In this manner, the model is updated either with LSSVM update or output bias update, depending on the optimized value of d and the current calibration measurement.

PLS-scheme is simulated by following the same algorithm as shown in Fig. 1. The threshold d for PLS-scheme was optimized followed by its application to predict NOx emission. The flow of sequential arrangement of model update and predictions through both versions of scheme is shown in Fig. 4.

The difference between the actual values of the inputs of the model and the predicted output is referred to as deviation or error, which can be measured in different ways. Here, relative root mean squared error (rRSME), mean absolute error (MAE), mean relative error (MRE) and corrected sample standard deviation of error (STD) are adopted for the comparison purpose between LSSVM-scheme and PLS-scheme. The term RMSE will be used in next sections to represent the rRMSE.

$$\text{relative RMSE} = \sqrt{\frac{1}{N_t}\sum_{i=1}^{N_t}\left(\frac{Y_{i,\,actual} - Y_{i,\,mod}}{Y_{i,\,actual}}\right)^2} \tag{15}$$

$$\text{MAE} = \frac{1}{N_t}\sum_{i=1}^{N_t}\left|Y_{i,\,actual} - Y_{i,\,mod}\right| \tag{16}$$

$$\text{MRE} = \frac{1}{N_t}\sum_{i=1}^{N_t}\left|\frac{Y_{i,\,actual} - Y_{i,\,mod}}{Y_{i,\,actual}}\right| \tag{17}$$

$$\text{STD} = \sqrt{\frac{1}{N_t - 1}\sum_{i=1}^{N_t}(e_i - \bar{e})^2} \tag{18}$$

where Y_{actual} and Y_{mod} are the actual and modified values (predicted values are modified by the Eq. (14)) of NOx respectively, N_t is the number of observations in test or validation data set, $e_i = Y_{i,actual} - Y_{i,mod}$ and $\bar{e}$ is the mean of e_i.

3. Optimization Approach

For the optimization of the threshold constant d, the following equation applies to both LSSVM-scheme and PLS-scheme. The optimization problem takes the following form:

Table 1. Optimized parameters

Scheme	Optimized parameters	Iteration number
PLS-scheme	d=0.3	48
LSSVM-scheme	d=0.1, gamma=0.2 sig. sq.=0.1	448

$$\text{Minimize: RMSE(d)=LSSVM-scheme or PLS-scheme} \quad 0 \le d \le \text{RCDV} \tag{19}$$

where RCDV is defined as the noisiness, i.e., unsmoothed behavior of output. It can be computed as the mean of the difference between NOx values at a particular time instance (t) and the previous time instance (t−1). In this application the RCDV is calculated to be 7.1623 as the upper limit of the range of the threshold constant. The threshold value that gives the lowest RMSE is used as the updating method selector d.

4. Schemes Formulation

4-1. LSSVM-scheme

Table 1 shows the optimized parameters. Threshold d is optimized using optimization data as 0.1 in this application along with gamma=0.2 and sigma square=0.1.

Fig. 5 represents the optimization result corresponding to minimized RMSE, gamma, sigma square and number of LSSVM run (NLR). The optimization iterations are plotted against RMSE, gamma, sigma square and NLR with varying d form RCDV to zero. With decreasing values of d, NLR follows a stepwise decreasing trend. The number of iterations during optimization is truncated to 50 to depict the results clearly.

4-2. PLS-scheme

Non-linear iterative partial least squares (NIPALS) algorithm is used for PLS framework. Usually the number of latent factors for

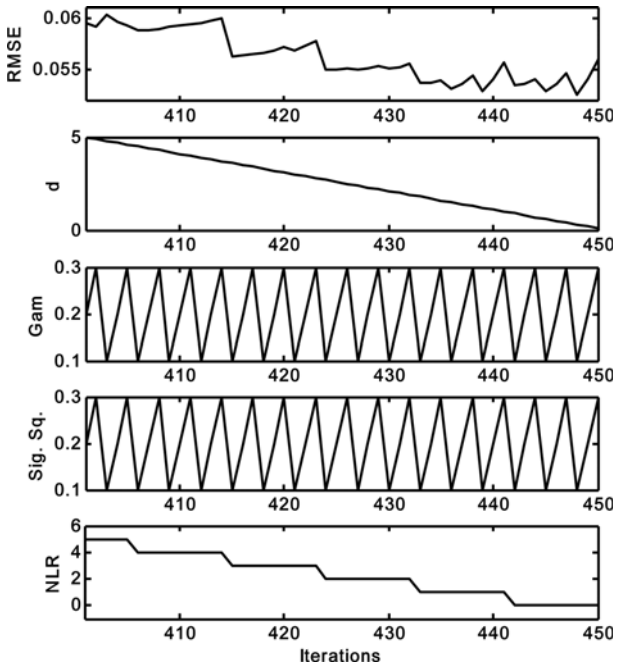

Fig. 5. Optimization of threshold (d) for minimum RMSE and corresponding gamma, sigma square and NLR.

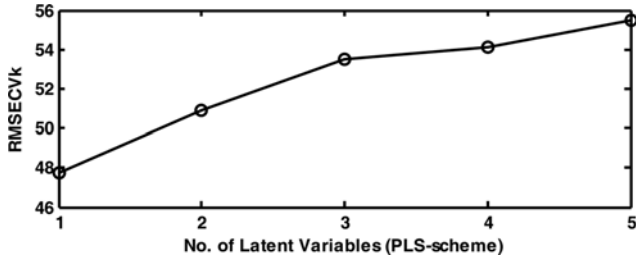

Fig. 6. Selection of number of latent factors (variables) for PLS model in PLS-scheme.

PLS model is determined by cross validation [26]. In cross validation, the data is divided into k subsets or folds, a model is built using all subsets except one subset at a time. The model is then simulated and tested on the left-out subset and root mean squared error of k-fold cross validation ($RMSECV_k$) is calculated. This procedure is repeated until every subset is left out and tested by the model based on other remaining subsets. The number of latent factors giving the lowest $RMSECV_k$ is selected for PLS model [26]. In this scheme 5-fold cross validation is used to compute the number of latent factors. $RMSECV_k$ is calculated as follows:

$$RMSECV_k = \sqrt{\frac{\Sigma_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{n}} \quad (20)$$

where $\hat{Y}$ are the predicted values and is the number of observations included in model building subsets. One latent factor for PLS-scheme (Table 1) is selected based on the lowest $RMSECV_k$, which is 47.7134 as shown in Fig. 6. The threshold value d for PLS-scheme is optimized to be 0.3 as shown in Table 1.

Fig. 7 demonstrates the optimization of d corresponding to minimum RMSE and number of PLS run (NPR) for PLS-scheme. It represents optimization iterations against RMSE and NPR changing with varying d from RCDV to zero. With decreasing values of d, NPR is observed to follow a stepwise decreasing manner. The number of iterations during optimization is truncated to 450 to show the results clearly.

4-3. Model Validation through Chance Correlation

To formulate the variants of variables for chance correlation test, MATLAB function randsample( ) was used to perform random permutation, and random numbers were generated between 0 and 32767 with uniform distribution by the MATLAB function ran-

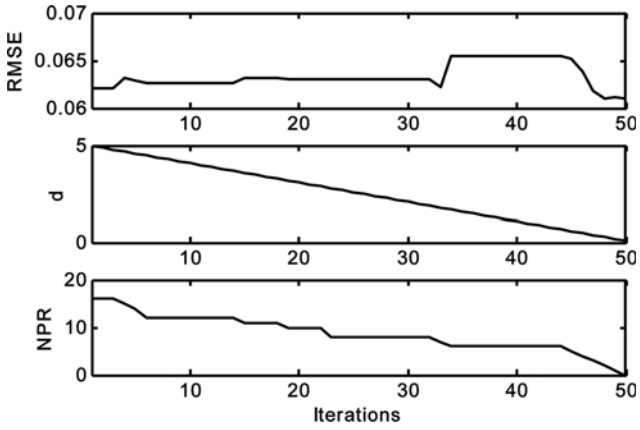

Fig. 7. Optimization of threshold (d) for minimum RMSE and corresponding NPR.

dom( ). Using these variants, four different models were developed and the coefficients of determination were determined for each variant model. The results from these variant models and original model are shown in Table 2. The positive R^2 values for original model (Y vs. X) approaching 1 indicate higher correlation and predictive ability of the model's good data fitting, while the negative R^2 values describe lower correlation and poor data fitting ability for the models developed from the variants for both versions of schemes. Hence the results validate the original models and confirm that LSSVM-scheme and PLS-scheme are not a consequence of chance correlation. Note that for nonlinear cases, R^2 values outside the range 0 to 1 may occur where it is used to measure the agreement between observed and modeled values [27].

RESULTS AND DISCUSSION

The optimization was for PLS-scheme and RMSE of optimization data is calculated as 0.0610 (Table 3). The trend of optimization data against actual values of NOx is shown in Fig. 8(a). For validation purpose, the model was applied to the test data in real-time manner: calibration values were added to the model after every two hours and the RMSE of validation data was computed to be 0.0763 as shown in Table 3. PLS-scheme predicts the NOx emission in a good manner and follows the actual NOx emission values accordingly as the emission of NOx varies with the passage of

Table 2. R^2 values calculated for variants of chance correlation

Scheme	Y vs. X	PY vs. X	RY vs. X	PY vs. RX	RY vs. RX
PLS-scheme	0.6668	−0.9688	−0.9912	−2.9618	−1.3079
LSSVM-scheme	0.7115	−1.0296	−1.0038	−0.9808	−0.9876

Table 3. Performance comparison of LSSVM and PLS schemes

Scheme	Optimization data				Validation data			
	RMSE	MAE	MRE (%)	STD	RMSE	MAE	MRE (%)	STD
PLS-scheme	0.0610	8.3973	4.6	11.1011	0.0763	10.2795	5.88	13.5524
LSSVM-scheme	0.0536	7.7851	4.24	9.8947	0.0696	9.4111	5.34	12.6129

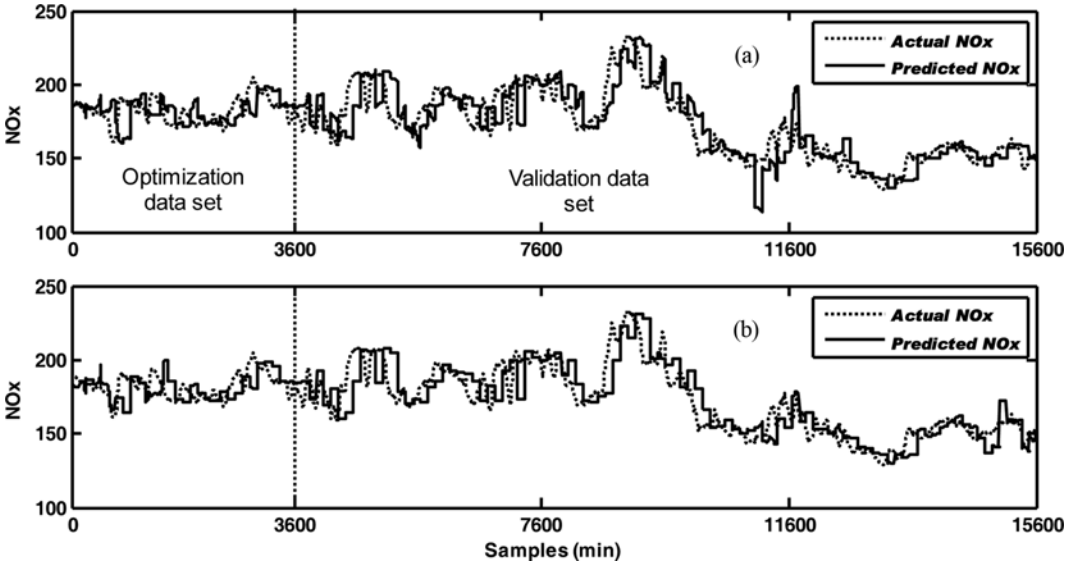

Fig. 8. Comparison of NOx predictions: (a) PLS-scheme, RMSE (validation)=0.0763 (b) LSSVM-scheme, RMSE (validation)=0.0696.

time which spans over several days (Fig. 8(a)). Initially, it achieved high accuracy by following the nonlinear behavior of NOx emission in real-time manner. Since the update scheme is characterized by its inherent property of the dependence of threshold d on the output variable behavior (NOx emission behavior) and also the dependence of the update selection method on threshold d, it effectively predicts NOx and keeps the prediction error under acceptable range. However, after several days in our application, prediction from PLS-scheme deviated a bit and remained diverged for several hours. Later on, real-time updates of PLS and output bias through calibration measurements minimized these deviations.

For LSSVM version of the scheme, RMSE of optimization data and validation data were calculated as 0.0536 and 0.0696, respectively (Table 3) and the trend of optimization data and validation data is shown in Fig. 8(b). Lower values of RMSEs for proposed LSSVM version of scheme than that of PLS version for both calibration and validation data show that it outperforms the PLS version of the scheme. It exhibits better performance over the whole period of time without any deviation (Fig. 8(b)). It initially predicts NOx exceptionally while suppressing the deviations which were quite apparent in the PLS-scheme. The reason for the LSSVM-scheme keeping a good track of NOx emission and also absorbing the disturbing effect of abrupt change in some process variable is the better generalization ability of LSSVM over PLS. This ability of LSSVM over PLS allows LSSVM-scheme to predict reliably and remain stable even after several days of uninterrupted predictions and causes 12% decrease in RMSE with respect to PLS-scheme applied to this application.

The prediction performance of models can also be visualized by a very simple graph, which plots the measured result against the predicted ones. A well-fitted model results in predicted values close to the actual data values. In the graph of a well fitted model, all points fall near a diagonal line and a small amount of noise give deviations from this line. Fig. 9(a) and 9(b) give the graph of the actual NOx vs. the predicted NOx for PLS-scheme and LSSVM-scheme, respectively. 12000 data points of actual NOx and predicted NOx were excessive to plot them clearly; therefore, the data points were sampled with a sampling time of 100 minutes and plotted to enhance the visibility of the figure. R^2 is interpreted as the proportion of total variance explained by the model and measure of extent of fitness of model; its values for both schemes are tabulated in Table 2. It can be seen from Fig. 9(a) that the data points in the PLS-scheme graph with a model fitness criteria R^2=0.6668 are a bit more sparse than that of LSSVM-scheme graph with R^2=0.7115 (Fig. 9(b)). It indicates the better fitted model and superior predictive ability by LSSVM-scheme over PLS-scheme.

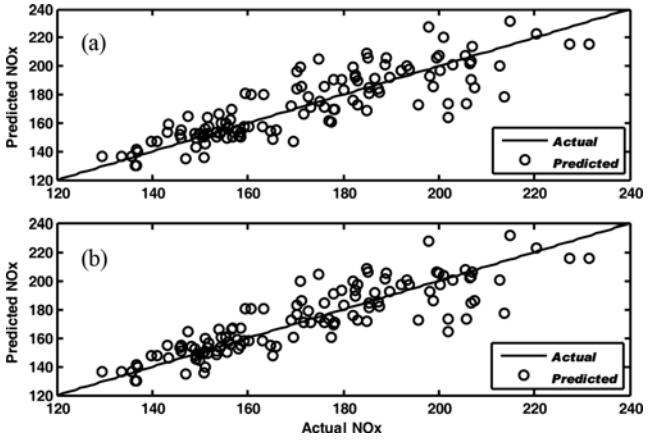

Fig. 9. Actual vs. predicted NOx: (a) PLS-scheme, R^2=0.6668 (b) LSSVM-scheme, R^2=0.7115.

To illustrate the universality of the LSSVM-scheme, a detailed comparison of the optimization and validation data set is presented in Table 3. The performance shown by values of MAE, MRE and STD is consistent with the above test results.

The optimized value of threshold d mainly depends on the NOx emission's behavior. The algorithm of real-time scheme is formu-

lated as to trigger output bias update to handle and overcome sudden and sharp deviations. The more un-smoothed response behavior, the smaller the optimized value of d. The values of d near lower bound trigger output bias update in greater number of times than that of LSSVM/PLS update. From this, it is observed that if the output response behavior is varying in steps, schemes are prone to select the output bias update in greater number of times than LSSVM/PLS update to minimize the RMSE. Number of PLS run (NPR) and number of LSSVM run (NLR) are compared with the number of bias run (NBR). In PLS-scheme optimization converged with the objective function of minimum RMSE at iteration number 48 (Table 1) at which NPR runs five times and NBR runs 95 times. While in LSSVM-scheme, minimum RMSE was found at 448^{th} iteration (Table 1) at which NLR and NBR are triggered three times and 97 times, respectively. This is important to note that NOx emission's behavior of this power plant activates the output bias update run in greater number of times than that of LSSVM/PLS update run. This feature of proposed scheme renders it faster, which saves computing time by employing LSSVM/PLS update only when it is needed.

CONCLUSIONS

A new approach is proposed to use LSSVM and output bias in online fashion to successfully predict the NOx emission from a tangentially fired pulverized coal boiler in Taean Power Plant, Korea. Two real-time update models were developed based on LSSVM and PLS titled LSSVM-scheme and PLS-scheme, respectively. A comparison is made between proposed LSSVM-scheme and PLS-scheme by predicting NOx emissions. The proposed LSSVM-scheme outperforms in terms of prediction performance because of better generalization ability of LSSVM than that of PLS over a wide range of data and long-term predictions without deviation. Therefore, it is concluded that the combination of smart update method selection capability of real-time scheme and better generalization ability of LSSVM creates a real-time model superior than that of developed by the combination of real-time scheme and PLS in previous studies. Moreover, consistent results indicate that the LSSVM-scheme provides prediction reliability and accuracy and is supposed to have promising potential for practical use of its real-time version.

ACKNOWLEDGEMENT

We would like to thank Taean Power Plant, Taean, South Korea for their technical support.

REFERENCES

1. H. Zhou, K. Cen and J. Fan, *Energy*, **29**, 167 (2004).
2. Z. Ligang, J. Hailin, Y. Minggao and Y. Minggao, *Bioinformatics and Biomedical Engineering (iCBBE), 2010 4th International Conference on*, 1 (2010).
3. S. Matsumura, T. Iwahara, K. Ogata, S. Fujii and M. Suzuki, *Control Eng. Pract.*, **6**, 1267 (1998).
4. J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P. Glorennec, H. Hjalmarsson and A. Juditsky, *Automatica*, **12**, 1691 (1995).
5. K. Li, S. Thompson and J. Peng, *Control Eng. Prac.*, **12**, 707 (2004).
6. Y. Lv, J. Liu and T. Yang, *Ind. Eng. Chem. Res.*, **51**(49), 16092 (2012).
7. Y. K. Lee, M. Kim and C. Han, *J. Environ. Eng.*, **131**, 961 (2005).
8. T. Adali, B. Bakal, M. K. SÖNmez and R. Fakory, *Integr Comput-Aid E*, **6**, 27 (1999).
9. F. Wu, H. Zhou, T. Ren, L. Zheng and K. Cen, *Fuel*, **88**, 1864 (2009).
10. S. Hui, *Power and Energy Engineering Conference (APPEEC), 2012 Asia-Pacific*, 1 (2012).
11. V. Vapnik, *The nature of statistical learning theory*, Springer Verlag, New York (1995).
12. J. A. K. Suykens and J. Vandewalle, *Neural Processing Letters*, **9**, 293 (1999).
13. L. Sheng, and Y. Zhen, *Industrial Engineering and Engineering Management (IE&EM), 2011 IEEE 18th International Conference on*, 1732 (2011).
14. S. Tatinati, Y. Wang, G. Shafiq and K. C. Veluvolu, *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, 6043 (2013).
15. A. Niazi, S. Sharifi and E. Amjadi, *J. Electroanal. Chem.*, **623**, 86 (2008).
16. A. Niazi, M. Goodarzi and A. Yazdanipour, *J. Brazilian Chemical Society*, **19**, 536 (2008).
17. R. M. Balabin and E. I. Lomakina, *Analyst*, **136**, 1703 (2011).
18. F. Ahmed, S. Nazir and Y. K. Yeo, *Korean J. Chem. Eng.*, **26**(1), 14 (2009).
19. C. Cortes and V. Vapnik, *Mach Learn*, **20**, 273 (1995).
20. K. Helland, H. E. Berntsen, O. S. Borgen and H. Martens, *Chemom. Intell. Lab. Syst.*, **14**, 129 (1992).
21. S. Mu, Y. Zeng, R. Liu, P. Wu, H. Su and J. Chu, *J. Process Contr.*, **16**, 557 (2006).
22. H. Kaneko and K. funatsu, *J. Chem. Eng. Japan*, **46**, 219 (2013).
23. R. Sharmin, U. Sundararaj, S. Shah, L. V. Griend and Y. Sun, *Chem. Eng. Sci.*, **61**, 6372 (2006).
24. H. Kubinyi, "Qsar in drug design," *Handbook of Chemoinformatics*, Wiley-VCH Verlag GmbH, 1532 (2008).
25. C. Rücker, G. Rücker and M. Meringer, *J. Chem. Inf. Model.*, **47**, 2345 (2007).
26. P. Geladi and B. R. Kowalski, *Anal. Chim. Acta*, **185**, 1 (1986).
27. P. Kennedy, *A guide to econometrics*, MIT Press (2003).