

A SOFT-SENSING MODEL FOR FEEDWATER FLOW RATE USING FUZZY SUPPORT VECTOR REGRESSION

MAN GYUN NA*, HEON YOUNG YANG and DONG HYUK LIM

Department of Nuclear Engineering, Chosun University

375 Seosuk-dong, Dong-gu, Gwangju 501-759, Korea

*Corresponding author. E-mail : magyna@chosun.ac.kr

Received November 19, 2007

Accepted for Publication December 29, 2007

Most pressurized water reactors use Venturi flow meters to measure the feedwater flow rate. However, fouling phenomena, which allow corrosion products to accumulate and increase the differential pressure across the Venturi flow meter, can result in an overestimation of the flow rate. In this study, a soft-sensing model based on fuzzy support vector regression was developed to enable accurate on-line prediction of the feedwater flow rate. The available data was divided into two groups by fuzzy c-means clustering in order to reduce the training time. The data for training the soft-sensing model was selected from each data group with the aid of a subtractive clustering scheme because informative data increases the learning effect. The proposed soft-sensing model was confirmed with the real plant data of Yonggwang Nuclear Power Plant Unit 3. The root mean square error and relative maximum error of the model were quite small. Hence, this model can be used to validate and monitor existing hardware feedwater flow meters.

KEYWORDS : Feedwater Flow Rate Monitoring, Fuzzy C-means Clustering, Fuzzy Support Vector Regression, Genetic Algorithm, Soft-sensing, Subtractive Clustering

1. INTRODUCTION

Thermal nuclear reactor power is typically evaluated with secondary system calorimetric calculations based on feedwater flow rate measurements. The feedwater flow rate should therefore be measured accurately. Venturi meters are currently used to measure the feedwater flow rate in most pressurized water reactors (PWRs). The long-term operation of a nuclear power plant causes a buildup of corrosion products near the orifice of the meter. This fouling increases the measured pressure drop across the meter, thereby causing an overestimation of the feedwater flow rate. Whenever calorimetric calculations are conducted during an operating cycle, the thermal reactor power must be decreased to match the false feedwater flow rate overestimated by the Venturi meter. This requirement means that a nuclear power plant must be operated at a lower-than-planned power level.

The fouling phenomenon of Venturi meters is the most significant contributor to PWR derating, which ranges from 0.5% to 3%. The most common practice for resolving this problem at PWRs is to inspect and clean the Venturi meters during every refueling period. However, fouling can reappear in as quickly as a month. With time, the accuracy of the existing hardware sensors becomes degraded due to the

fouling phenomena of the Venturi meter. Many researchers have therefore been interested in resolving the inaccurate measurements of the feedwater flow rate [1-4]. Hence, in this study we developed a soft-sensing model for predicting the feedwater flow rate.

Recently, many researchers have paid considerable attention to soft-sensing, which uses other readily available on-line measurements. This type of soft sensor can either replace existing hardware sensors or be used in parallel to provide redundancy and to verify whether the sensors are drifting [4-10]. The problem can be resolved by using learning and soft computing techniques if the process dynamics for evaluating the process variables is a priori unknown or difficult to model. We therefore developed a fuzzy support vector regression (FSVR) model that can increase the thermal efficiency of a nuclear power plant by making accurate on-line predictions of the feedwater flow rate.

2. A SOFT-SENSING MODEL FOR THE FEEDWATER FLOW RATE

Venturi flow meters measure the flow rate by developing a differential pressure across a physical flow restriction.

The differential pressure is then multiplied by a calibration factor that depends on various flow conditions in order to calculate the feedwater flow rate. The calibration factor is determined by the feedwater temperature and pressure, which are measured with resistance thermometers and pressure transmitters, respectively.

The original thermal power margin needed to evaluate an emergency core cooling system (ECCS) is 2%, regardless of the demonstrated instrument accuracy. A revision to 10CFR50 Appendix K for the ECCS evaluation model was recently modified to allow a margin equal to the actual instrument accuracy. Thus, the thermal power of a nuclear power plant can be increased by 1% or more than its licensed power through the use of advanced and more accurate instruments [11]. The recent revision to 10CFR50 Appendix K encourages the use of advanced feedwater flow instruments in real nuclear power plants.

To predict the feedwater flow rate, we developed a soft-sensing model based on a support vector machine (SVM) equipped with a fuzzy concept. Soft-sensing techniques generally require learning and soft computing -based approaches, such as neural networks (NNs) and SVMs, because they can easily represent complicated processes that are difficult to model with analytical and mechanistic models. These approaches, known as data-based methods, utilize the available data. Even if the methods based on NNs and SVMs yield results that are comparable to the results of the most popular benchmark, the theoretical status of SVMs is an attractive and promising area of research [12].

2.1 Soft-Sensing by FSVR

We used an FSVR method for the soft-sensing measurements of the feedwater flow rate of a PWR. A regression problem approximates an unknown function that can be expressed as a linear expansion of basis functions. The regression problem is transformed to determine the coefficients of the basis function of linear expansion. The support vector regression (SVR) nonlinearly maps the original input data, \mathbf{x} , into a higher dimensional feature space. Thus, in the given set of data $\{(\mathbf{x}_k, y_k)\}_{k=1}^N \in R^m \times R$, where \mathbf{x}_k is the input vector of an SVM, y_k is the actual output value, and N is the total number of data patterns, the SVR is based on the following regression function:

$$y = f(\mathbf{x}) = \sum_{k=1}^N w_k \phi_k(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}) + b, \quad (1)$$

where

$$\mathbf{w} = [w_1 \ w_2 \ \dots \ w_N]^T, \\ \boldsymbol{\varphi} = [\phi_1 \ \phi_2 \ \dots \ \phi_N]^T.$$

The function $\phi_k(\mathbf{x})$ is called the feature, and the parameters \mathbf{w} and b are the support vector weight and the bias. We can calculate these parameters by minimizing the regularized risk function as follows:

$$R(\mathbf{w}) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \lambda \sum_{k=1}^N |y_k - f(\mathbf{x})|_\epsilon, \quad (2)$$

where

$$|y_k - f(\mathbf{x})|_\epsilon = \begin{cases} 0, & \text{if } |y_k - f(\mathbf{x})| < \epsilon \\ |y_k - f(\mathbf{x})| - \epsilon, & \text{otherwise} \end{cases} \quad (3)$$

The first term of Eq. (2) characterizes the complexity of the SVR models. The parameters λ and ϵ are user-defined parameters, and $|y_k - f(\mathbf{x})|_\epsilon$ is known as the ϵ -insensitive loss function [13]. The loss equals zero if the estimated value falls within an ϵ -tube. For all other estimated points outside the error level, ϵ , the loss is equal to the magnitude of the difference between the estimated value and ϵ (as shown in Fig. 1).

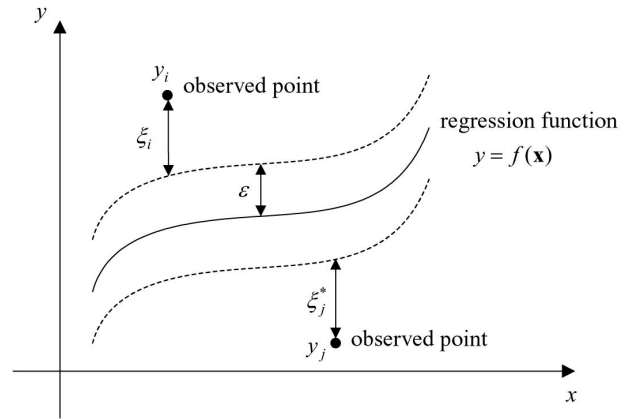


Fig. 1. The Parameters for the FSVR Models

The FSVR is a form of SVR equipped with a fuzzy concept. The proposed FSVR enhances the SVR by reducing the effect of outliers and noise. Through the application of a fuzzy membership function to each data point of the SVR model, the regularized risk function can be reformulated in such a way that different input data points can make different contributions to the learning of a regression function as follows:

$$R(\mathbf{w}) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \lambda \sum_{k=1}^N \mu_k |y_k - f(\mathbf{x})|_\epsilon, \quad (4)$$

where μ_k is a fuzzy membership grade. Commonly used SVR methods apply an equal weighting to all data points. However, FSVR uses different weightings according to their importance, which is specified by the fuzzy membership grade. Minimizing the regularized risk function is equivalent to minimizing the following

constrained risk function:

$$R(\mathbf{w}, \xi, \xi^*) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + \lambda \sum_{k=1}^N \mu_k (\xi_k + \xi_k^*), \quad (5)$$

subject to the constraints

$$\begin{cases} y_k - \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}) - b \leq \varepsilon + \xi_k, & k = 1, 2, \dots, N \\ \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}) + b - y_k \leq \varepsilon + \xi_k^*, & k = 1, 2, \dots, N \\ \xi_k, \xi_k^* \geq 0, & k = 1, 2, \dots, N \end{cases} \quad (6)$$

where the constant λ determines the trade-off between the complexity of $f(\mathbf{x})$ and the amount up to which deviations greater than ε are tolerated. The parameters $\xi = [\xi_1 \ \xi_2 \ \dots \ \xi_N]^T$ and $\xi^* = [\xi_1^* \ \xi_2^* \ \dots \ \xi_N^*]^T$ are the (positive) slack variables that represent the upper and lower constraints on the outputs of the system.

The constrained optimization problem can be solved by applying the Lagrange multiplier technique to Eqs. (5) and (6). The technique can be expressed by the following Lagrange functional:

$$\begin{aligned} \Phi(\mathbf{w}, b, \xi, \xi^*, \alpha_k, \alpha_k^*, \beta_k, \beta_k^*) &= \frac{1}{2} \mathbf{w}^T \mathbf{w} + \lambda \sum_{k=1}^N \mu_k (\xi_k + \xi_k^*) \\ &- \sum_{k=1}^N \alpha_k [\mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_k) + b - y_k + \varepsilon + \xi_k] - \sum_{k=1}^N \alpha_k^* [y_k - \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}_k) - b + \varepsilon + \xi_k^*] \\ &- \sum_{k=1}^N (\beta_k \xi_k + \beta_k^* \xi_k^*). \end{aligned} \quad (7)$$

The minimization of Eq. (7) with respect to the primal variables \mathbf{w} , b , ξ_k , and ξ_k^* gives the following conditions:

$$\begin{aligned} \mathbf{w} &= \sum_{i=1}^N (\alpha_i - \alpha_i^*) \boldsymbol{\varphi}(\mathbf{x}_i), \quad \sum_{k=1}^N (\alpha_k - \alpha_k^*) = 0, \\ \mu_k \lambda - \alpha_k - \beta_k &= 0, \quad \mu_k \lambda - \alpha_k^* - \beta_k^* = 0, \quad k = 1, 2, \dots, N. \end{aligned}$$

We can then rewrite the Lagrange functional [14] by using the above minimum conditions as follows:

$$\begin{aligned} \Phi(\alpha_k, \alpha_k^*) &= \sum_{k=1}^N y_k (\alpha_k - \alpha_k^*) - \varepsilon \sum_{k=1}^N (\alpha_k + \alpha_k^*) - \\ &\frac{1}{2} \sum_{k=1}^N \sum_{i=1}^N (\alpha_k - \alpha_k^*) (\alpha_i - \alpha_i^*) \boldsymbol{\varphi}^T(\mathbf{x}_k) \boldsymbol{\varphi}(\mathbf{x}_i) \end{aligned} \quad (8)$$

subject to the constraints

$$\begin{cases} \sum_{k=1}^N (\alpha_k - \alpha_k^*) = 0 \\ 0 \leq \alpha_k \leq \mu_k \lambda, & k = 1, 2, \dots, N. \\ 0 \leq \alpha_k^* \leq \mu_k \lambda, & k = 1, 2, \dots, N \end{cases}$$

The Lagrange functional of Eq. (8) can be solved if we use a quadratic programming technique to determine the values of α_k and α_k^* . Finally, the regression function of Eq. (1) is expressed as follows:

$$y = f(\mathbf{x}) = \sum_{k=1}^N (\alpha_k - \alpha_k^*) \boldsymbol{\varphi}^T(\mathbf{x}_k) \boldsymbol{\varphi}(\mathbf{x}) + b = \sum_{k=1}^N (\alpha_k - \alpha_k^*) K(\mathbf{x}, \mathbf{x}_k) + b, \quad (9)$$

where $K(\mathbf{x}, \mathbf{x}_k) = \boldsymbol{\varphi}^T(\mathbf{x}_k) \boldsymbol{\varphi}(\mathbf{x})$ is known as the kernel function. A number of coefficients, $(\alpha_k - \alpha_k^*)$, are nonzero values, and the corresponding training data points, which are known as support vectors (SVs), have an approximation error greater than or equal to ε . We used the following radial basis function as the kernel function:

$$K(\mathbf{x}, \mathbf{x}_k) = \exp \left(-\frac{(\mathbf{x} - \mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k)}{2\sigma^2} \right). \quad (10)$$

The bias, b , is calculated as follows:

$$b = -\frac{1}{2} \sum_{k=1}^N (\alpha_k - \alpha_k^*) (K(\mathbf{x}_r, \mathbf{x}_k) + K(\mathbf{x}_s, \mathbf{x}_k)),$$

where \mathbf{x}_r and \mathbf{x}_s are called SVs and they are data points positioned at the boundary of the ε -insensitivity zone (as shown in Fig. 1) [13].

The four most relevant design parameters for the FSVR model are the insensitivity zone, ε , the regularization parameter, λ , the kernel function parameter, σ , and the fuzzy membership grade, μ . An increase in the insensitivity zone, ε , reduces the accuracy requirements of the approximation and allows a decrease in the number of SVs, thereby facilitating the data compression; such an increase also has smoothing effects on the modeling of highly noisy data. In addition, an increase in the regularization parameter, λ , reduces larger errors, thereby minimizing the approximation error. The minimization of the approximation error can also be achieved by increasing the weight vector norm. However, an increase in the weight vector norm decreases the good generalization capability of the FSVR model. The kernel function parameter, σ , determines the sharpness of the radial basis kernel function. In Section 2.3, we explain the fuzzy membership grade.

2.2 Genetic Optimization

The FSVR model was optimized by a process of learning from available data. As mentioned, the performance of the FSVR model depends strongly on the four major design parameters of the insensitivity zone, ε , the regularization parameter, λ , the kernel function parameter, σ , and the fuzzy membership grade, μ . These parameters (except for the fuzzy membership grade) must therefore be optimized in order to maximize the performance of the FSVR model.

A genetic algorithm is the most useful method for solving optimization problems with multiple objectives [15-16]. Hence, we used a genetic algorithm to optimize the insensitivity zone, ϵ , the regularization parameter, λ , the sharpness (σ) of the radial basis kernel function, and two additional parameters (r_α and r_β) of a subtractive clustering (SC) method (which is explained in next section) for sampling the training data from the available data.

To optimize these five parameters, we encoded them as a bit string in each chromosome of the genetic algorithm. Our specified multiple objectives were to minimize the root mean squared (RMS) error and the maximum error, and we achieved these objectives by maximizing the following fitness function:

$$F = \exp(-w_1 E_1 - w_2 E_2 - w_3 E_3 - w_4 E_4), \quad (11)$$

where w_1 , w_2 , w_3 and w_4 are the weighting coefficients and E_1 , E_2 , E_3 and E_4 are defined as follows:

$$E_1 = \sqrt{\frac{1}{N_t} \sum_{k=1}^{N_t} (y_k^t - \hat{y}_k^t)^2}, \quad (12)$$

$$E_2 = \sqrt{\frac{1}{N_o} \sum_{k=1}^{N_o} (y_k^o - \hat{y}_k^o)^2}, \quad (13)$$

$$E_3 = \max_k \{y_k^t - \hat{y}_k^t\}, \quad (14)$$

$$E_4 = \max_k \{y_k^o - \hat{y}_k^o\}. \quad (15)$$

The variable y_k denotes the measured (target) output and the variable \hat{y}_k denotes the output predicted from the FSVR model. The superscripts, t and o , indicate the training and optimization data, respectively, and N_t and N_o represent the number of training and optimization data points. The design parameters of the genetic algorithm that optimizes the FSVR model are the crossover probability, the mutation probability, and the population size.

2.3 Data Grouping and Training Data Selection

The computing time for training the FSVR model increases exponentially as the number of training data points increases. To reduce the training time, we divided the available input data into subsets (groups) and then developed an individual FSVR model for each group.

A fuzzy c-means (FCM) clustering method was used to group the available input data [17-18]. FCM clustering is a clustering method that classifies one set of data into

two or more groups. The FCM clustering uses fuzzy partitioning in which a data point can belong to all groups with different membership grades. The aim of the FCM method is to minimize the following objective function [18]:

$$J(u_{kg}, \mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_G) = \sum_{k=1}^N \sum_{g=1}^G u_{kg}^q \|\mathbf{x}_k - \mathbf{c}_g\|^2, \quad (16)$$

where u_{kg} is a membership grade where a data point \mathbf{x}_k belongs to group g , q is any real number greater than 1, \mathbf{c}_g is a cluster center, and G is the number of groups. The parameter q determines the fuzziness of the clusters. The number of groups is determined on the basis of the data distribution. We used only two groups.

The FCM clustering was carried out with iterative optimization of the objective function, as in Eq. (16), and the membership matrix was randomly initialized to satisfy the following condition:

$$\sum_{g=1}^G u_{kg} = 1, k = 1, \dots, N. \quad (17)$$

The membership, u_{kg} , and each cluster center, \mathbf{c}_g , were updated as follows:

$$\mathbf{c}_g = \frac{\sum_{k=1}^N u_{kg}^q \mathbf{x}_k}{\sum_{k=1}^N u_{kg}^q}, \quad (18)$$

$$u_{kg} = \frac{1}{\sum_{j=1}^G (\|\mathbf{x}_k - \mathbf{c}_g\| / \|\mathbf{x}_k - \mathbf{c}_j\|)^{2/(q-1)}}. \quad (19)$$

The iteration procedure stops when $\max_{k,g} \{u_{kg}^{l+1} - u_{kg}^l\} < \epsilon$,

where ϵ is a termination criterion and l represents an iteration step. Each data point \mathbf{x}_k is classified into group n if $\max_g u_{kg} = u_{kn}$.

In addition, each set of data was divided into three types of data sets: a training data set, an optimization data set, and a test data set. The training data was used to solve the coefficients, α_k - α_k^* , and the bias, b , in Eq. (9). The optimization data was used in conjunction with a genetic algorithm to optimize the major design parameters (the insensitivity zone, ϵ , the regularization parameter, λ , and the kernel function parameter, σ) of the FSVR models and to determine the additional parameters (cluster radii, r_α and r_β) for the sampling of the training data. The test data was used to confirm the developed FSVR models.

The selection of appropriate training data is important because it can affect the performance of the FSVR model.

The input and output training data are expected to have many clusters in each group, and the data at these cluster centers are more informative than the neighboring data. An FSVR model for each data set can be well trained with informative data. The cluster centers are located with the aid of an SC scheme and used as the training data set.

We assumed that N_g input/output training data $\mathbf{z}_k=(\mathbf{x}_k, y_k)$, $k=1, 2, \dots, N_g$ in a group were available and that the data points were normalized in each dimension. The SC scheme begins by generating a number of clusters in an $m \times N$ dimensional input space. The SC scheme uses the following equation to measure the potential of each data point, which is a function of the Euclidean distances to all other input data points [19]:

$$P_i(k) = \sum_{j=1}^{N_g} e^{-4\|\mathbf{x}_k - \mathbf{x}_j\|^2 / r_a^2}, \quad k = 1, 2, \dots, N_g, \quad (20)$$

where r_a is the radius that defines a particular neighborhood. Note that the potential of a data point is high when the data point is surrounded by an abundance of neighboring data. After the potential of each data point is calculated, the data point with the highest potential is selected as the first cluster center.

In general, after the i -th cluster center, \mathbf{c}_i and its potential value, P_i^c , have been determined, the potential of each data point can be revised with the following equation:

$$P_{i+1}(k) = P_i(k) - P_i^c e^{-4\|\mathbf{x}_k - \mathbf{c}_i\|^2 / r_b^2}, \quad k = 1, 2, \dots, N_g, \quad (21)$$

where r_b is usually greater than r_a in order to limit the number of clusters generated. Equation (21) indicates that an amount of potential is subtracted from each data point as a function of its distance from the cluster center. The data points near the cluster center have greatly reduced potential and are unlikely to be selected as the next cluster center. When the potentials of all the data points have been revised according to Eq. (21), the data point with the highest potential is selected as the $(i+1)^{\text{th}}$ cluster center. These calculations stop if the inequality $P_i^c < \epsilon P_1^c$ is true; otherwise the calculations are repeated. If the calculations are eventually stopped at an iterative step, N_c , then each data group has N_c cluster centers. The input/output data (training data) positioned in the cluster centers of the data group are selected to train the FSVR model for each group. In addition, at every fifth time-step, the test data is selected from the remaining sequential data where the training data have already been eliminated. Hence, the optimization data and the test data comprise 80% and 20% of the remaining sequential data, respectively.

For the training of the FSVR models, the data points with a high potential, as calculated by Eq. (20), are more important and weighted more highly than the other neighboring data points. Thus, the potential of the cluster

centers calculated by Eq. (20) is used as a fuzzy membership grade in Eq. (4) as follows:

$$\mu_k = 1 - \frac{1}{P_i(k)}, \quad k = 1, \dots, N_c. \quad (22)$$

3. APPLICATION OF THE SOFT-SENSING MODEL TO THE FEEDWATER FLOW PREDICTION

The proposed algorithm was confirmed with the real plant startup data of the Yonggwang Nuclear Power Plant Unit 3 (YGN3). These data are the values measured from the primary and secondary systems of the nuclear power plant, with particular focus on the steam generator (SG). The data is derived from the following 16 types of measured signals: the SG feedwater flow rate, the SG steam flow rate, the SG pressure, the SG temperature, the SG wide-range level, the SG narrow-range level, the hot-leg temperature, the cold-leg temperature, the pressurizer pressure, the pressurizer temperature, the pressurizer water level, the feedwater temperature, the reactor power (the ex-core neutron detector signal), the feedwater pump suction pressure, the feedwater pump discharge pressure, and the steam header pressure.

The SG feedwater flow rate was the target output signal of the FSVR model, and all other signals were candidate input signals for the FSVR model. The time differential values of the feedwater flow rate are also candidate input signals. We used the FCM method to divide the available data into two groups and then developed an FSVR model for each group. In addition, we used the SC method to separate each group into three data sets: namely, the training data, the optimization data, and the test data.

The degree of the relation between the candidate input signals and the feedwater flow rate (the target output signal) can be derived from a correlation matrix of the candidate input/output signals. For candidate input signals with a very close relation to the target output, we inputted the current and past delayed values into the FSVR model. However, we did not use the candidate input signals with a very low relation to the output as input for the FSVR model. Moreover, for the other candidate input signals, we used only the current values as input for the FSVR model.

To optimize the proposed FSVR model with the genetic algorithm, we set the parametric values of the genetic algorithm as 1 for the crossover probability, 0.05 for the mutation probability, and 20 for the population size. The optimized parameters are as follows:

$$\begin{aligned} \epsilon &= 1.4974 \times 10^{-4}, \lambda = 2.7847 \times 10^3, \sigma = 4.7720 \\ &\text{for the first group,} \\ \epsilon &= 8.1229 \times 10^{-4}, \lambda = 6.8332 \times 10^2, \sigma = 4.5975 \\ &\text{for the second group.} \end{aligned}$$

Figure 2 shows the data points selected by the SC

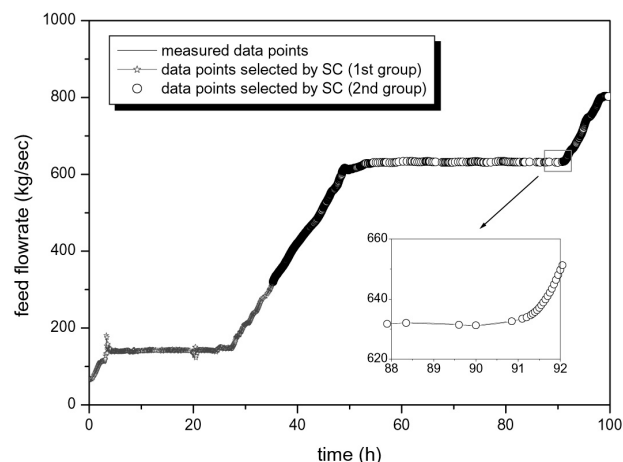


Fig. 2. Data Points Selected for Optimizing the FSVR Models

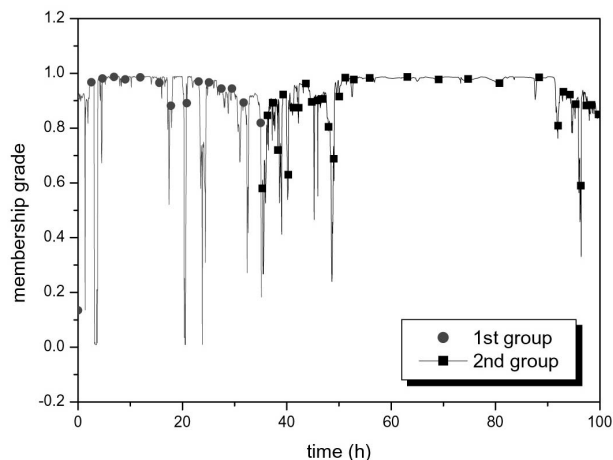
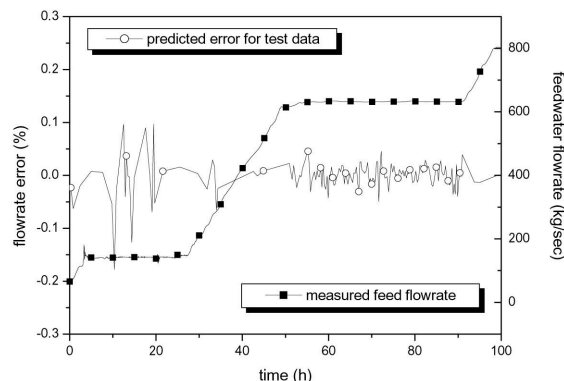


Fig. 3. Fuzzy Membership Grade for the Input Data Points

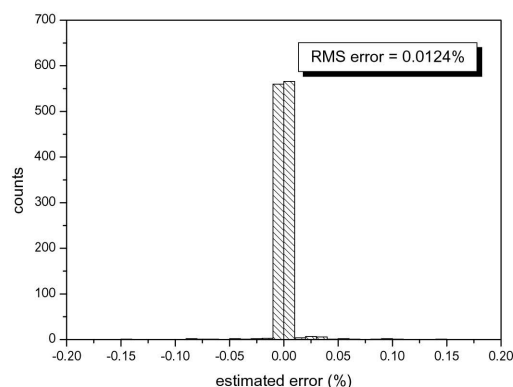
scheme for training the proposed FSVR model. The selected data points were concentrated for the first 50 h and distributed quite sparsely for the period from 55 h to 90 h. This result means that the informative data is concentrated in the first 50 h. In addition, the data are selected relatively sparsely in the period from 55 h to 90 h because the power plant system is close to a steady state for that period.

Figure 3 shows the fuzzy membership grade for the training data. The fuzzy membership grade is calculated by Eq. (22). The data points with a high potential are more important and weighted more highly than the other neighboring data points.

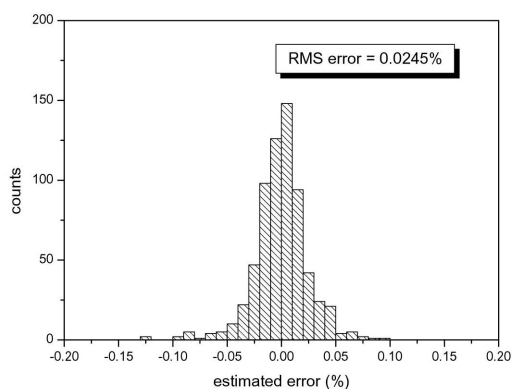
The measured feedwater flow rate was assumed to degrade rapidly. Even if the time period of 100 h is short in terms of an overall operating fuel cycle and the buildup time of corrosion products, it does not have influence on



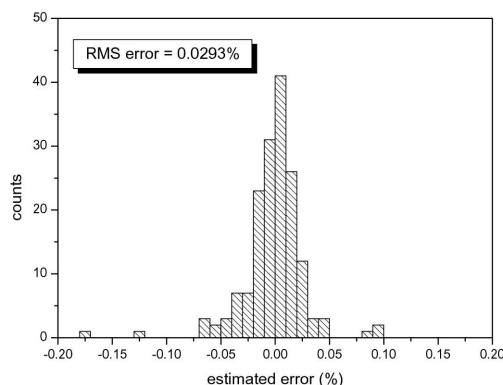
(a) Feedwater flow rate error



(b) Prediction error histogram (training data)



(c) Prediction error histogram (optimization data)



(d) Prediction error histogram (test data)

Fig. 4. Feedwater Flow Rate Error and Histogram

investigating the performance of the proposed algorithm. In this type of acceleration test, the proposed algorithm can be used to verify the overestimation of the feedwater flow rate for the entire operating fuel cycle.

Figure 4 shows the performance of the proposed algorithm. Figure 4(a) shows the measured feedwater flow rate and the prediction errors obtained with the FSVR model in conjunction with the FCM clustering method and the SC scheme. Figures 4(b), 4(c) and 4(d) show histograms of the predicted errors for the training, optimization, and test data obtained with the proposed FSVR model, respectively. In these figures, the relative RMS errors compared with the rated value (801.34 kg/sec) were 0.0124% for the training data, 0.0245% for the optimization data, and 0.0293% for the test data. These errors are so small that the estimated feedwater flow rate can be used to monitor the measured flow rate.

Figure 5 shows the simulation results in the case where the feedwater flow rate begins to be artificially degraded after the first 20 h. The predicted feedwater flow rate is almost the same as the actual feedwater flow rate even though the measured feedwater flow rate had degraded.

Table 1 summarizes the performance of the proposed soft-sensing method for the feedwater flow rate. The results of the second data group are better than those of the first data group. This outcome, as shown in Fig. 2, appears to be due to some unreliable data measured at

around 3 h and 20 h. Table 2 compares the proposed method with a previous result [10]. As shown in this table, it was possible to improve the performance by training the FSVR model with the training data, which was divided by means of the FCM method and selected with the aid of the SC scheme. The errors in the test data

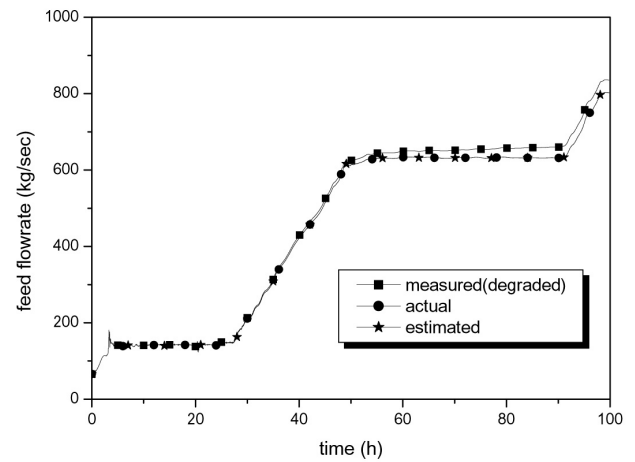


Fig. 5. Soft-Sensing of the Feedwater Flow Rate in Cases of Artificial Degradation

Table 1. Performance of the FSVR for the Feedwater Flow Rate

Data group	Data type	RMS error (%)	Relative maximum error (%)	Number of data	Number of support vectors
1st group	Training data	1.5525×10^{-2}	1.4930×10^{-1}	564	562
	Optimization data	4.5276×10^{-2}	1.2453×10^{-1}	112	-
	Test data	5.9990×10^{-2}	1.7778×10^{-1}	28	-
2nd group	Training data	8.5819×10^{-3}	1.0659×10^{-2}	601	440
	Optimization data	1.7500×10^{-2}	5.2297×10^{-2}	552	-
	Test data	1.7348×10^{-2}	4.9209×10^{-2}	138	-

Table 2. Comparison of the Soft-Sensing Methods for the Feedwater Flow Rate

Methods	Data type	RMS error (%)	Relative maximum error (%)	Number of data	Number of support vectors
Proposed method	Training data	1.2437×10^{-2}	1.4930×10^{-1}	1165	1002
	Optimization data	2.4503×10^{-2}	1.2453×10^{-1}	664	-
	Testdata	2.9278×10^{-2}	1.7778×10^{-1}	166	-
Previous work [10]	Training data	2.8873×10^{-2}	1.5247×10^{-1}	779	631
	Optimization data	7.3029×10^{-2}	4.6816×10^{-1}	977	-
	Test data	9.2343×10^{-2}	6.0066×10^{-1}	245	-

are similar to those in the optimization data. Thus, we can use the FSVR model to predict the feedwater flow rate if the FSVR model is optimally identified in the initial stage with the training and optimization data.

5. CONCLUSION

A soft-sensing algorithm was developed in conjunction with an FSVR model to validate and monitor the existing feedwater flow rate. In order to reduce the training time, we used the FCM method to divide the available data into two groups and we subsequently developed two independent FSVR models. In addition, to train the FSVR model with more informative data, we selected the training data from the available data of each group by applying an SC scheme; we then used the training data in conjunction with a genetic algorithm to optimize the soft-sensing algorithm. The developed soft-sensing algorithm actually predicts the feedwater flow rate signal and removes any false over-measurement effect that occurs as a result of fouling degradation of the Venturi meters.

The proposed soft-sensing algorithm was applied to the acquired real plant startup data of YGN3. In the simulations, the RMS error and the relative maximum error for the test data were only 0.0293% and 0.1778%, respectively. The developed soft-sensing algorithm can therefore be used to validate and monitor existing feedwater flow meters.

REFERENCES

- [1] K. Kavaklioglu and B. R. Upadhyaya, "Monitoring feedwater flow rate and component thermal performance of pressurized water reactors by means of artificial neural networks," *Nuclear Technology*, vol. 107, p. 112-123, July 1994.
- [2] G. Heo, S. S. Choi, and S. H. Chang, "Feedwater flowrate estimation based on the two-step de-noising using the wavelet analysis and an autoassociative neural network," *J. Korean Nucl. Soc.*, vol. 31, no. 2, pp. 192-201, April 1999.
- [3] A. V. Gribok, I. Attieh, J. W. Hines, and R. E. Uhrig, "Regularization of feedwater flow rate evaluation for venturi meter fouling problem in nuclear power plants," NURETH-9, San Francisco, California, Oct. 3-8, 1999.
- [4] M. G. Na, S. H. Shin, and D. W. Jung, "Design of a software sensor for feedwater flow measurement using a fuzzy inference system," *Nuclear Technology*, vol. 150, no. 3, pp. 293-302, June 2005.
- [5] D.-J. Choi and H. Park, "A hybrid artificial neural network as a software sensor for optimal control of a wastewater treatment process," *Water Research*, vol. 35, no. 16, pp. 3959-3967, 2001.
- [6] N. Régner, G. Defaye, L. Caralp, and C. Vidal, "Software sensor based control of exothermic batch reactors," *Chemical Engineering Science*, vol. 51, no. 23, pp. 5125-5136, 1996.
- [7] S. Linko, J. Luopa, and Y.-H. Zhu, "Neural networks as 'software sensors' in enzyme production," *Journal of Biotechnology*, vol. 52, no. 3, pp. 257-266, 1997.
- [8] A. Chérut, "Software sensors in bioprocess engineering," *Journal of Biotechnology*, vol. 52, no. 3, pp. 193-199, 1997.
- [9] M. H. Masson, S. Canu, Y. Grandvalet, and A. Lynggaard-Jensen, "Software sensor design based on empirical data," *Ecological Modeling*, vol. 120, nos. 2-3, pp. 131-139, 1999.
- [10] Man Gyun Na, In Joon Hwang, and Yoon Joon Lee, "Inferential Sensing and Monitoring for Feedwater Flowrate in Pressurized Water Reactors," *IEEE Trans. Nucl. Sci.*, Vol. 53, No. 4, pp. 2335-2342, Aug. 2006.
- [11] J. Regan and H. Estrada, "The elements of uncertainty in feedwater flow measurements with three types of instruments," NPIC&HMIT 2000, Washington, DC, Nov. 2000.
- [12] V. Kecman, *Learning and Soft Computing*. Cambridge, Massachusetts: MIT Press, 2001.
- [13] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer, 1995.
- [14] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*, Academic Press, New York, 1982.
- [15] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, Massachusetts: Addison Wesley, 1989.
- [16] M. Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, Massachusetts: MIT Press, 1996.
- [17] J. C. Dunn, "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact, Well-Separated Clusters," *J. Cybernetics*, vol. 3, no. 3, pp. 32-57, 1973.
- [18] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.
- [19] S. L. Chiu, "Fuzzy model identification based on cluster estimation," *J. Intell. Fuzzy Systems*, vol. 2, pp. 267-278, 1994.