Proceedings of the Korean Nuclear Society Autumn Meeting
Seoul, Korea, October 2001

A Neuro-Fuzzy Inference System for Sensor Monitoring

Man Gyun Na

Department of Nuclear Engineering, Chosun University
375 Seosuk-dong, Dong-gu, Kwangju 501-759, Korea

Abstract

A neuro-fuzzy inference system combined with the wavelet denoising, PCA (principal component analysis) and SPRT (sequential probability ratio test) methods has been developed to monitor the relevant sensor using the information of other sensors. The parameters of the neuro-fuzzy inference system which estimates the relevant sensor signal are optimized by a genetic algorithm and a least-squares algorithm. The wavelet denoising technique was applied to remove noise components in input signals into the neuro-fuzzy system. By reducing the dimension of an input space into the neuro-fuzzy system without losing a significant amount of information, the PCA was used to reduce the time necessary to train the neuro-fuzzy system, simplify the structure of the neuro-fuzzy inference system and also, make easy the selection of the input signals into the neuro-fuzzy system. By using the residual signals between the estimated signals and the measured signals, the SPRT is applied to detect whether the sensors are degraded or not. The proposed sensor-monitoring algorithm was verified through applications to the pressurizer water level, the pressurizer pressure, and the hot-leg temperature sensors in pressurized water reactors.

1. Introduction

In nuclear power plants, sensor signals from many different measurement locations are used in control and safety critical systems and for plant state identification. Therefore, these signals must be validated to increase the reliability of operator decisions and automatic plant operations. Sensor validation can be done through accurate mathematical modeling and computer coding of a process which are usually very difficult. Recently, a lot of diagnostic techniques of sensors using neural networks and fuzzy inference methods have been developed. After the neuro-fuzzy inference system for sensor monitoring is trained beforehand, the neuro-fuzzy system monitors sensors on-line. By using these on-line condition-monitoring systems, the nuclear power plant safety is increased and also the plant availability can be improved considerably. The unnecessary and unexpected plant shutdown can be prevented and reduced when their failures are detected early. Also, plant outage times for repairing can be minimized and maintenance measures and schedules can be planned optimally while any proceeding of incipient failures is under control of the monitoring system.

The studies on sensor monitoring using artificial intelligence in nuclear power plants was started in the late 80's by Upadhyaya, et. al. [1,2], was also conducted by Singer, et. al. [3], Hines and Uhrig, et. al. [4,5], and Fantoni, et. al [6]. Reifman [7] surveyed artificial intelligence methods for detection and identification of component faults. Through training, the neuro-fuzzy systems have been known to be very good at phenomenal nonlinear function approximation and pattern recognition, especially when expert diagnostic knowledge and the prior relation of fault symptom model are not clear.

In this work, a wavelet denoising and a principal component analysis (PCA) for input signal preprocessing, a neuro-fuzzy system for signal estimation, and a sequential probability ratio test (SPRT) for statistical fault decision are combined for a sensor monitoring and the structure of these combined processes is shown in Fig. 1 [8]. The direct use of transient signals in the time domain to the input of a neuro-fuzzy inference system can be difficult since the subtle differences may occur between different transients. Therefore, it is necessary to preprocess the transient signals. A wavelet denoising technique is applied to remove noise components in the input signals on the neuro-fuzzy inference system. One of characteristics of wavelets is able to analyze a localized area of a larger signal [9]. The dimension of the input signals to a neuro-fuzzy inference system had better be reduced to save the time necessary to train the neuro-fuzzy inference system. Principal component analysis (PCA) [10,11] is used to reduce the dimension of an input space without losing a significant amount of information. This method transforms the input space into an orthogonal space. Also, the PCA method makes easy the selection of the input to the neuro-fuzzy inference system. By using the input signals preprocessed by the wavelet denoising technique and the principal component analysis, a neuro-fuzzy inference system estimates the relevant signals. The neuro-fuzzy system parameters such as the membership functions and the connectives between layers in a neuro-fuzzy inference system are optimized by a genetic algorithm and a least-squares algorithm.

An important problem in sensor monitoring is whether a sensor is decided to be degraded or not after only one abnormal observation. It is sure that several measurements can give a reliable result. At every new sample, a new mean and a new variance may be computed and then, these quantities may be used to check if the sensor is degraded or not. However, this procedure requires too many samples to obtain a meaningful mean and a meaningful variance and also, during the acquisition of the samples, a significant degradation of the process monitored may occur. Therefore, in this work the sequential probability ratio test (SPRT) [12] was used. The method can detect a failure using the degree of degradation and the continuous behavior of the sensor, without having to calculate a new mean and a new variance at each sample. The signal estimated by the neuro-fuzzy inference system is compared with the measured signal, and then the SPRT monitors the sensor using the residuals.

The proposed algorithm was applied to the sensor monitoring of the pressurizer (PRZR) water level and the PRZR pressure, and the hot-leg temperature in pressurized water reactors and was verified for abrupt and gradual bias degradations of these sensors and their noise degradation.

2. Input Selection and Preprocessing

It is difficult to select the number of input signals to be used and also appropriate input signals. In this work, all collected signals excluding an output come through a wavelet denoising block which noise will be removed in and then come through a PCA block which the dimension of an input space into the neuro-fuzzy inference system will be reduced in. The wavelet denoising and PCA methods will be briefly explained below.

2.1 Wavelet Denoising

In wavelet analysis, a signal is considered to consists of shifted and scaled versions of the original wavelet called mother wavelet, while in Fourier analysis, a signal is considered to be composed of sine waves of various frequencies. Since wavelets is able to analyze a localized area of a larger signal, a wavelet analysis is capable of revealing aspects of data that other signal analysis techniques can miss, aspects like trends, breakdown points, discontinuities in higher derivatives, and self-similarity [13]. Let a signal $f(t)$ be expressed as

$$f(t)=\sum_k c_k\varphi_k(t) \quad \text{for any } f(t)\in V_0\,, \tag{1}$$

where V_0 is the subspace of $L^2(R)$ (the space of all functions with a well defined integral of the square of the modulus of the function) spanned by the scaling functions $\varphi_k(t)$ with all integers k from minus infinity to infinity. The size of the subspace can generally be increased by changing the time scale of the scaling functions that is generated from the basic scaling function by scaling and translation expressed as $\varphi_{j,k}(t)=2^{j/2}\varphi(2^j t-k)$ [9].

By introducing the wavelet functions $\psi_{j,k}(t)$ that span the differences between the spaces spanned by the various scales of the scaling function $\varphi_{j,k}(t)$, the important features of a signal can be better described. If $f(t)\in V_{j+1}$ can be expressed at a scale of $j+1$, when it is expressed at one scale lower resolution, wavelets are necessary for the detail not available at a scale of j as follows:

$$f(t)=\sum_k a_{j+1}(k)\varphi_{j+1,k}(t)=\sum_k a_j(k)\varphi_{j,k}(t)+\sum_k d_j(k)\psi_{j,k}(t)\,. \tag{2}$$

Since $\varphi_{j,k}(t)$ and $\psi_{j,k}(t)$ are orthonormal, in Eq. (2) $a_j(k)$ and $d_j(k)$ can be expressed as follows:

$$a_j(k)=\sum_m l(m-2k)a_{j+1}(m)\,, \tag{3}$$

$$d_j(k)=\sum_m h(m-2k)a_{j+1}(m)\,. \tag{4}$$

The low pass and high pass filtering of the input signal is thought as a moving average with the coefficients being the weights like Eqs. (3) and (4), respectively. Wavelet decomposition is to obtain low pass approximations and high pass details. An approximation is a low-resolution representation of the original signal, while a detail is the difference between two successive low-resolution representations of the original signal [14]. Thus, an approximation contains the general trend of the original signal, while a detail contains the high frequency contents of the original signal. The detail and approximation of the original signal are obtained by passing it through a filter bank [9] which consists of low and high pass filters and by downsampling it. Downsampling means throwing away every second data point.

Using wavelets to remove noise from a signal requires identifying which components contain the noise and then reconstructing the signal without these components. Thus, successive approximations become less and less noisy as more and more high-frequency information is filtered out of the signal.

2.2 Principal Component Analysis (PCA)

PCA is a method of preprocessing data to extract uncorrelated features from data. A PCA method involves linearly transforming the input space into an orthogonal space that can be chosen to be of lower dimension with minimal loss of information and is used to reduce the dimension of an input space into the neuro-fuzzy inference system. A lower dimensional input space will reduce the time necessary to train a neuro-fuzzy inference system.

Given a signal vector $\mathbf{x}$ of p dimensions, $\mathbf{x} = [x_1 \quad x_2 \quad \cdots \quad x_p]^T$, its true mean and covariance matrix are replaced with the sample mean $\mathbf{m}$ and the sample covariance matrix $\mathbf{S}$ because they are seldom known. The eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_p$, and the corresponding orthonormal eigenvectors $\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_p$ of the covariance matrix $\mathbf{S}$ are calculated, and then the eigenvalues are arranged according to their magnitude, $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_p$. The respective eigenvectors $\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_p$ are called the principal components. The eigenvalues are proportional to the amount of variance (information) represented by the corresponding principal component. The transformation to the principal component space can be written as:

$$\mathbf{z} = \mathbf{x}^T \mathbf{P}, \tag{5}$$

where $\mathbf{P} = [\mathbf{p}_1 \ \mathbf{p}_2 \ \cdots \ \mathbf{p}_p]$.

The feature vector $\mathbf{z}$ can be transformed back into the original data vector $\mathbf{x}$ without a loss of information as long as the number of features, m, is equal to the dimension of the original space, p. For $m < p$, some information is usually lost. The objective is to choose a small m that does not lose much information. In this work, the feature vector which is calculated by Eq. (5) and has smaller dimension than the dimension of an original data space is used as inputs to the neuro-fuzzy inference system.

3. Neuro-Fuzzy Inference System for Sensor Signal Estimation

It is required that sensor signals should be estimated first to monitor sensors because the residuals between the estimated signals and the measured signals are used to decide whether sensors are degraded or not. In this work, sensor signals are estimated by a neuro-fuzzy inference system. A neuro-fuzzy inference system or an adaptive network-based fuzzy inference system (ANFIS) [15] consists of a fuzzy inference system and its neuronal training. Therefore, a fuzzy inference system and its training method will be briefly explained below.

3.1 Fuzzy Inference System

In a fuzzy inference system, the arbitrary i-th rule can be described using the first-order Sugeno-Takagi type [16] as follows:

$$\text{If } x_1 \text{ is } A_{i1} \textit{ AND } \cdots \textit{ AND } x_m \text{ is } A_{im}, \text{ then } y_i \text{ is } f_i(x_1, \cdots, x_m), \tag{6}$$

where

$x_1, \cdots, x_m$ = input variables to the neuro-fuzzy inference system (m = number of input variables),

$A_{i1}, \cdots, A_{im}$ = antecedent membership functions of each input variable for the i-th rule (i = 1, 2, ..., n),

y_i = output of the i-th rule,

$$f_i(x_1, \cdots, x_m) = \sum_{j=1}^{m} q_{ij} x_j + r_i, \tag{7}$$

q_{ij} = weighting value of the j-th input onto the i-th rule output,

r_i = bias of the i-th output,

n = number of rules.

In this work, the Gaussian and sigmoid membership functions are used for each input variable and these membership functions are shown in Fig. 2. The sigmoid membership function is used for the maximum and minimum center values, c_{ij}, in each input variable, x_j, while the Gaussian membership function is used for other center values (refer to Fig. 2). It is shown in Fig. 2 as if five membership functions always exist for each input signal. However, the number of membership functions for each input is the same as the number of rules. If the number of rules is three, two sigmoid membership functions are used for the maximum and minimum center values and one Gaussian membership function is used for a middle center value. The output of an arbitrary i-th rule, f_i, consists of the first-order polynomial of inputs as given in Eq. (7). The output of a fuzzy inference system with n rules is obtained by weighting the real values of consequent parts for all rules with the corresponding membership grade. The estimated output for sensor signal estimation is described as follows:

$$\hat{y} = \sum_{i=1}^{n} \overline{w}_i f_i , \tag{8}$$

where

$$\overline{w}_i = \frac{w_i}{\sum_{i=1}^{n} w_i} , \tag{9}$$

$$w_i = \prod_{j=1}^{m} A_{ij}(x_j) . \tag{10}$$

The neuro-fuzzy inference system for signal estimation is described in Fig. 3. x_1, x_2 and x_m are the input values to the fuzzy inference system and A_{ij} means the membership function of the j-th input for the i-th rule. The membership value for rule i, w_i, means a compatibility grade between antecedent parts through multiplicative weight. If the Fig. 3 is explained from left to right, the signs Π and N mean multiplication and normalization which are expressed as Eqs. (10) and (9), respectively. The second sign Π and the sign Σ are expressed as Eq. (8). The sign Σ means the summation of the input values.

In the next subsection, this fuzzy inference system will be trained, which means that its rules and membership functions are automatically generated and tuned.

3.2 Training of the Fuzzy Inference System

The neuro-fuzzy inference system is optimized by adapting the antecedent parameters (membership function parameters) and consequent parameters (the polynomial coefficients of the consequent part) so that a specified objective function is minimized. The adaptation methods of most fuzzy inference systems rely on the back-propagation algorithm which is generally used to recursively solve for parameter optimization. Since this conventional optimization algorithm is susceptible to getting stuck at local optima, the genetic algorithm is used in this work. However, the genetic algorithm requires much time if there are many parameters to be optimized. Therefore, the least-squares method that is a one-pass optimization method is combined for optimizing a part of the parameters. The genetic algorithm is used to optimize the antecedent parameters c_{ij} and s_{ij} (refer to Fig. 2), and the least-squares algorithm is used to solve the consequent parameters q_{ij} and r_i in Eq. (7) [17].

To use a genetic algorithm, a solution to a given problem must be represented as a chromosome which can be thought of as a point in the search space of candidate solutions and the chromosome contains the antecedent parameters c_{ij} and s_{ij} which describe the fuzzy membership functions. The genetic algorithm then creates a population of solutions (chromosomes) and applies genetic operators such as selection, crossover and mutation to evolve the solutions in order to find the best one. The genetic algorithms require a fitness function that assigns a score to each chromosome in the current population. The fitness of a chromosome (individual) depends on how well that chromosome solves the problem at hand [18,19]. In this work, a fitness function that evaluates the extent to which each individual is suitable for the given objectives such as small maximum error together with small total squared error and small number of inputs, was suggested as follows:

$$F = \exp\left(-\mu_1 E_1 - \mu_2 E_2 - \mu_3 E_3\right), \tag{11}$$

where μ_1, μ_2 and μ_3 are the weighting coefficients, and E_1, E_2 and E_3 are overall sum of squared errors, maximum absolute error, and the number of inputs defined as follows:

$$E_1 = \sum_{k=1}^{N} \left(y(k) - \hat{y}(k)\right)^2 , \tag{12}$$

$$E_2 = \max_k \left\{ \left| y(k) - \hat{y}(k) \right| \right\} , \tag{13}$$

$$E_3 = N_{input} . \tag{14}$$

$y(k)$ and $\hat{y}(k)$ denote the measured signal and the estimated signal, respectively.

In this work, to increase the efficiency of the conventional genetic algorithm, three schemes are applied to accomplish the following good performance of the genetic algorithm: (a) initial coarse tuning and final fine tuning by changing the bit number of chromosomes versus generation; (b) prevention of an initial premature convergence without reaching optimal solutions and the acceleration of a final convergence by using two different selection methods of the crossover site that is randomly selected anywhere in a chromosome or randomly selected between only parameters in a chromosome; (c) prevention of final drifting without convergence by a certain part of chromosomes with higher fitness surviving a subsequent generation (refer to [20] for details).

If some parameters of the fuzzy inference system are fixed by the genetic algorithm, the resulting fuzzy inference system is equivalent to a series expansion of some basis functions. This basis function expansion is linear in its adjustable parameters. Therefore, we can use the least-squares method to determine the remaining parameters (consequent parameters). When a total of N input-output pattern data for training are given, from Eq. (8) the consequent parameters are chosen such that the pattern data satisfy the following equation:

$$\mathbf{y} = \mathbf{W}\mathbf{q} , \tag{15}$$

where $\mathbf{y}$ is the measured output data, $\mathbf{q}$ is the parameter vector consisting of the consequent parameters, and the matrix $\mathbf{W}$ includes the input data defined as, respectively

$$\mathbf{y} = \left[y^1 \; y^2 \; \cdots \; y^N \right]^T ,$$

$$\mathbf{q} = \left[q_{11} \cdots q_{n1} \cdots\cdots q_{1m} \cdots q_{nm} \; r_1 \cdots r_n \right]^T ,$$

$$\mathbf{W} = \left[\mathbf{w}^1 \; \mathbf{w}^2 \; \cdots \; \mathbf{w}^N \right]^T ,$$

$$\mathbf{w}^k = \left[\overline{w}_1 x_1^k \cdots \overline{w}_n x_1^k \cdots\cdots \overline{w}_1 x_m^k \cdots \overline{w}_n x_m^k \; \overline{w}_1 \cdots \overline{w}_n \right]^T , \quad k = 1, 2, \cdots, N.$$

The neuro-fuzzy inference outputs are represented by the $N \times (m+1)n$-dimensional matrix $\mathbf{W}$ and $(m+1)n$-dimensional parameter vector $\mathbf{q}$. In order to solve the parameter vector $\mathbf{q}$ in Eq. (15), the matrix $\mathbf{W}$ should be invertible but is not usually a square matrix. Therefore, we solve the vector using the pseudo-inverse as follows:

$$\mathbf{q} = \left(\mathbf{W}^T \mathbf{W} \right)^{-1} \mathbf{W}^T \mathbf{y} . \tag{16}$$

The least-squares method is a one-pass regression procedure and is consequently much faster than the back-propagation algorithm and the genetic algorithm.

4. Sensor Degradation Detection Using SPRT

In order to monitor sensors, at every new sample, a new mean and a new variance may be computed to check if the sensor is degraded or not. However, this procedure requires too many samples to obtain a meaningful mean and a meaningful variance. During the acquisition of the samples, a significant degradation of the process monitored may occur. So a method is required to detect a failure using the degree of degradation and the continuous behavior of the sensor without having to calculate a new mean and a new variance at each sample. The SPRT (Sequential Probability Ratio Test) which is a statistical model developed by Wald in 1945 [12] satisfies these requirements.

The objective of sensor degradation detection is to detect the degradation as soon as possible with a very small probability of making a wrong decision. In the application of sensor monitoring, the SPRT uses the residual (difference between the sensor measurement and the sensor estimate). Normally the residual signals are randomly distributed, so they are nearly uncorrelated and have a normal distribution $P_i(\varepsilon_k, m_i, \sigma_i)$, where ε_k is the residual signal at time k, and m_i and σ_i are the mean and the standard deviation under hypothesis i, respectively. The sensor degradation can be stated in terms of a change in the mean m or a change in the variance σ^2. The basis for the SPRT lies in the likelihood ratio, which is given by

$$\gamma_k = \frac{P_1(\varepsilon_k \mid H_1)}{P_0(\varepsilon_k \mid H_0)}, \tag{17}$$

where H_1 represents a hypothesis that the sensor is degraded and H_0 represents a hypothesis that the sensor is normal. The ratio is updated at every sampling step. If a set of samples x_i, $i = 1, 2, \cdots, n$, is collected with a density function P describing each sample in the set, an overall likelihood ratio is given by

$$\gamma_n = \frac{P_1(\varepsilon_1 \mid H_1) \cdot P_1(\varepsilon_2 \mid H_1) \cdot P_1(\varepsilon_3 \mid H_1) \cdots P_1(\varepsilon_n \mid H_1)}{P_0(\varepsilon_1 \mid H_0) \cdot P_0(\varepsilon_2 \mid H_0) \cdot P_0(\varepsilon_3 \mid H_0) \cdots P_0(\varepsilon_n \mid H_0)}. \tag{18}$$

By taking the logarithm of the foregoing equation and replacing the probability density functions in terms of residuals, means and variances, the log likelihood ratio (LLR, λ_n) can be written as the following recurrent form:

$$\lambda_n = \lambda_{n-1} + \ln\left(\frac{\sigma_0}{\sigma_1}\right) + \frac{(\varepsilon_n - m_0)^2}{2{\sigma_0}^2} - \frac{(\varepsilon_n - m_1)^2}{2{\sigma_1}^2}. \tag{19}$$

This form is used for deriving the sensor drift detection algorithm. By using the foregoing equation, we can identify two kinds of sensor degradations, bias and noise degradations. If only the bias degradation is checked, Eq. (19) can be converted into the following equation by substituting $\sigma_1^2 = \sigma_0^2$ and $m_0 = 0$ since normal residual signals usually have almost zero mean values:

$$\lambda_n = \lambda_{n-1} + \frac{m_1}{\sigma_0^2}\left(\varepsilon_n - \frac{m_1}{2}\right). \tag{20}$$

Also, if only the noise degradation is checked, Eq. (19) can be converted into the following equation by substituting $m_0 = m_1 = 0$ since mean values do not change in this case:

$$\lambda_n = \lambda_{n-1} + \ln\left(\frac{\sigma_0}{\sigma_1}\right) + \frac{\varepsilon_n^2}{2}\left(\frac{1}{{\sigma_0}^2} - \frac{1}{{\sigma_1}^2}\right). \tag{21}$$

For a normal sensor, the likelihood ratio would decrease and eventually reach a specified bound A, a smaller value than zero. When the ratio reaches this bound, the decision is made that the sensor is normal, and the ratio is initialized by setting it equal to zero. For a degraded sensor the ratio would increase and eventually reach a specified bound B, a larger value than zero. When the ratio is equal to B, the decision is made that the sensor is degraded. The decision boundaries A and B are chosen by a false alarm probability α and a missed alarm probability β; $A = \ln\left(\frac{\beta}{1-\alpha}\right)$ and $B = \ln\left(\frac{1-\beta}{\alpha}\right)$.

5. Applications

The proposed algorithm was applied to the pressurizer (PRZR) water level, the PRZR pressure and the hot-leg temperature. To verify the proposed algorithm, the input-output data were obtained for the load-decrease transients from the simulation of the MARS code [21] which is a unified version of COBRA/TF and RELAP5/MOD3. The input-output data consist of a total of 11 different signals from the primary and secondary sides of nuclear power plants (refer to Table 2) and also are standardized. In the three application cases (the PRZR water level, the PRZR pressure and the hot-leg temperature), one corresponding signal is used as an output and the remaining ten signals are used as inputs. Noise is added to all input and output data to model the real data of the nuclear power plant. The noise magnitude is proportional to the maximum variation $\sigma_{\max}$ of each signal and the noise has a normal distribution $N(0, 0.005\sigma_{\max})$. In all computer simulations that a denoising method is used, the wavelet denoising technique which uses a Daubechies wavelet function is applied to all measurement signals [22]. Each signal consists of a total of 700 discrete time points where its sampling period is 1 sec. The neuro-fuzzy inference system was trained using one fifth of all the given data in the training stage and was verified using the remaining data in the verification stage. The false alarm probability α and the missed alarm probability β are chosen as 0.0001 and 0.1, respectively. Therefore, the specific bounds, A and B, used to determine whether a sensor is degraded or not, are -2.3025 and 9.1050, respectively. And sensor tolerances are

5 percent mean value deviation between the maximum and minimum values of data (for degraded mean m_1) and 3 times standard deviation of the previous undegraded residuals (for degraded standard deviation σ_1).

In these three application cases, whether the denoising technique is used or not and whether the PCA method is used or not will be compared. Also, various types of output degradations (gradual bias degradation, noise degradation, abrupt bias degradation) will be tested and the cases one of only inputs (not output) is gradually degraded will be tested, too. Since the estimation errors are very small in spite of the small number of rules, the small number of rules (only two rules) is used, which can prevent an overfitting problem.

Table 1 shows the information amount of each feature component. The first four feature components for all the application cases contain almost all information which the input signals have. Therefore these four feature components were used as inputs to the proposed neuro-fuzzy inference system. Table 2 shows the collected signals and their correlation coefficient matrix which informs the relationship of the signals. Table 3 shows the signal estimation results of each application case. The error level is important to decide whether sensors are degraded or not. If the principal component analysis is not applied, the input signals was chosen to have a total of 5 signals so that the neuro-fuzzy inference system should maintain similar error levels (fitness values) irrespective of the PCA application. Note that four input signals were used if the PCA is applied. The application without PCA has one more input than that with PCA. In case that the principal component analysis is not applied, the chosen inputs are shown in Table 3 for the three application cases. These inputs were chosen through a correlation analysis and many computer simulations among the signals that have a close relationship with each output but have small dependence between chosen inputs so that the fitness value would be minimized. There is the small number of parameters to be optimized due to the relatively small numbers of inputs (4 or 5 inputs) and rules (2 rules). That means that there usually exists sufficient information in collected input-output data which can identify such a small number of the parameters. Therefore, there is no necessity for worrying about an overfitting problem.

The maximum error and the standard deviation for the three application cases are almost the same for the training data set and the verification data set, which means that once the neuro-fuzzy inference system is trained for a data set, the neuro-fuzzy system can be successfully applied to other data sets. If an output is estimated without removing noise, the output error is about three times larger (refer to Table 3). Therefore, it is necessary to remove noise before the inputs are applied to the neuro-fuzzy inference system. Of course, the most part of this larger error is induced by noise itself but this large error delays the degradation detection. Figures 4 through 6 show that the output signals are well estimated irrespective of the PCA and the denoising applications. Of course, when noise is not removed, the errors are relatively larger (refer to Table 3).

In all the three application cases, the output measurement signals was continuously degraded on purpose in a degree of 3.0×10^{-4} of the measured values each time step from $200\,\text{sec}$ to verify the sensor monitoring algorithm. The failure flag '1' in each figure represents that the sensor is decided to be degraded. Figures 7 through 9 show degradation detection times in case each output sensor is gradually degraded. If we cannot distinguish the detection time of each application case from these figures, we can find out the accurate detection times from Table 4. The detection times in Table 4 represent the times after the beginning of the gradual degradation. The detection times are almost the same (30~45 sec) irrespective of the PCA application (If the neuro-fuzzy inference system has four inputs in case the PCA is not applied, the detection time is expected to be a little longer). But if noise is not removed, the detection times are about two times longer (70~95 sec). This is because that the standard deviation of the estimation errors is used to determine whether it is degraded or not. As shown in Table 3, the standard deviation with the denoising application is smaller than that without the denoising application. Therefore, if the standard deviation is smaller, as can be known in Eq. (20), its log likelihood ratio (LLR) increases faster and reaches the specific bound B faster, which makes the detection time fast.

To verify the noise degradation detection of the proposed sensor monitoring algorithm, ten times larger noise is added from 200 sec. Figures 10 through 12 show degradation detection times for the three application cases. If noise is removed, the sudden large noise is detected simultaneously with the appearance of the new large noise. If the denoising technique is not applied, the detection times are 50 sec for the PRZR water level irrespective of the PCA application (refer to Table 4). And for the PRZR pressure, the detection times are 32 sec with the PCA application and 49 sec without the PCA application. For the hot-leg temperature, the detection times are 28 sec with the PCA applied and 49 sec with the PCA not applied. The detection times for noise degradation are faster if the PCA is applied.

To verify the abrupt bias degradation detection, the output signals were suddenly biased from 200 sec to a degree of one tenth of the maximum variation $\sigma_{\max}$ of each output signal (refer to Figs. 13-15). Irrespective of the PCA application and the denoising application, the sudden bias degradation is detected at almost the same time that the outputs are biased (refer to Table 4).

It is meaningful to verify the robustness of the sensor-monitoring algorithm in case that one of the input signals is gradually degraded and an output is not degraded. Table 5 shows the results of this test case. The outputs are decided to be degraded some time later for the degradation of some input signals and not decided to be degraded for the degradation of some other input signals. Since the output is not actually degraded, the outputs should not be decided to be degraded. However, if the degraded input is assumed to be an output and the neuro-fuzzy inference system and the SPRT are applied for the output signal, the proposed sensor monitoring algorithm detects faster the degradation of the output (actually the degraded input), which means that we can identify the degraded inputs and isolate them. There is one exception which is a case consisting of a PRZR pressure output and a degraded PRZR temperature input (PT) (refer to Table 5.b). This is a special case that an output signal is very closely related with a degraded input (almost one-to-one relationship) (refer to Table 2).

6. Conclusions

In this work, a neuro-fuzzy inference system combining the wavelet denoising, the PCA and the SPRT has been developed to monitor sensor degradations. The input signals into the neuro-fuzzy inference system are preprocessed by the wavelet denoising and the number of inputs is reduced by the PCA analysis. The first four feature components of a total of ten feature components are used as its input signals. The number of inputs may change according to the information contents of the ten feature components. In this work, the number of the feature components was chosen to have at least 97 percent of all information content which was known from a correlation analysis. The input signals into the neuro-fuzzy inference system can easily be selected with the PCA applied. The neuro-fuzzy inference system actually estimates the relevant output signal using other input signals. The SPRT decides whether a sensor is degraded or not by using the residuals between the measured signal and the estimated signal. The applications of the PCA and the denoising provide better performance by and large in detecting the gradual and abrupt bias degradations and the noise degradation. Also, to study the effect of input degradation on the sensor-monitoring algorithm, it was assumed that one of the input signals was gradually degraded and an output was not degraded. It was known that we could identify the degraded inputs and isolate them except for almost one-to-one relationship between an output and an input before the output would be decided to be degraded.

Acknowledgment

This work has been conducted under a Korean Nuclear Energy Research Initiative (K-NERI) project supported by the Korea Institute of Science & Technology Evaluation and Planning (KISTEP) which is funded by Korea Ministry of Science and Technology (Korea MOST).

References

1. B. R. UPADHYAYA, et. al., Development and Testing of an Integrated Signal Validation System for Nuclear Power Plants, DOE/NE/37959-35 (Sept. 1989).
2. A. ERBAY and B. R. UPADHYAYA, "A Personal Computer-Based On-Line Signal Validation System for Nuclear Power Plants," *Nucl. Technol.*, **119**, 63 (1997).
3. R. M. SINGER, K. C. GROSS, J. P. HERZOG, R. W. KING, and S. W. WEGERICH, "Model-Based Nuclear Plant Monitoring and Fault Detection: Theoretical Foundations," *Proc. 9th Intl. Conf. on Intelligence Systems Applications to Power Systems*, Seoul, Korea (1997).
4. J. W. HINES, D. J. WREST, and R. E. UHRIG, "Signal Validation Using an Adaptive Neural Fuzzy Inference System," *Nucl. Technol.*, **119**, 181 (1997).
5. J. W. HEINS, A. V. GRIBOK, I. ATTIEH, and R. E. UHRIG, "Regularization Method for Inferential Sensing in Nuclear Power Plants," in Da Ruan (ed) *Fuzzy Systems and Soft Computing in Nuclear Engineering*, Springer-Verlag, Berlin Heidelberg New York, pp.285-314 (1999).
6. P. FANTONI, S. FIGEDY, and A. RACZ, "A Neuro-Fuzzy Model Applied Full Range Signal Validation of PWR Nuclear Power Plant Data," *FLINS-98,* Antwerpen, Belgium (1998).
7. J. Reifman, “Survey of Artificial Intelligence Methods for Detection and Identification of Component Faults in Nuclear Power Plants,” *Nucl. Technol.*, **119**, 119 (1997).

8. M. G. NA, "Neuro-Fuzzy Inference System Combining Wavelet Denoising, PCA And SPRT for Sensor Monitoring," *NPIC & HMIT 2000*, Washington, D.C., Nov. 13-17 (2000).

9. C. S. BURRUS, R. A. GOPINATH, and H. GUO, *Introduction to Wavelets and Wavelet Transforms*, Prentice-Hall, London (1998).

10. X. Z. WANG and R. F. LI, "Combining Conceptual Clustering and Principal Component Analysis for State Space Based Process Montoring," *Ind. Eng. Chem. Res*., **38**, 4345 (1999).

11. J. CHEN and J. LIU, "Mixture Principal Component Analysis Models for Process Monitoring," *Ind. Eng. Chem. Res*., **38**, 1478 (1999).

12. A. WALD, *Sequential Analysis*, John Wiley & Sons, New York (1947).

13. M. MISITI, Y. MISITI, G. OPPENHEIM, and J.-M. POGGI, *Wavelet Toolbox User's Guide*, MathWorks, Natick, MA (1996).

14. P. KANG, D. BIRTWHISTLE, and K. KHOUZAM, "Transient Signal Analysis and Classification for Condition Monitoring of Power Switching Equipment Using Wavelet Transform and Artificial Neutral Network," *1998 2nd Intl. Conf. Knowledge-Based Intelligent Electronic Systems*, Apr. 21-23, 1998, Adelaide, Australia (1998).

15. J.-S. R. JANG, "ANFIS: Adaptive-Network-Based Fuzzy Inference Systems," *IEEE Transactions on Systems, Man, and Cybernetics*, **23**, 665 (1993).

16. T. TAKAGI and M. SUGENO, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," *IEEE Trans. System, Man, Cybern.*, **1**, 116 (1985).

17. M. G. NA, "A Fuzzy Neural Network Combining Wavelet Denoising and PCA for Sensor Signal Estimation," *J. Korean Nucl. Soc.*, **32**, 485 (2000).

18. D. E. GOLDBERG, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, Reading, Massachusetts (1989).

19. M. MITCHELL, *An Introduction to Genetic Algorithms*, The MIT Press, Cambridge, Massachusetts (1996).

20. M. G. Na, Neuro-Fuzzy Control Applications in Pressurized Water Reactors: in Da Ruan (ed) *Fuzzy Systems and Soft Computing in Nuclear Engineering*, Springer-Verlag, Berlin Heidelberg New York, pp.172-207 (1999).

21. W.-J. LEE, B.-D. CHUNG, J.-J. JEONG, K.-S. HA, and M.-K. HWANG, Improved Features of MARS 1.4 and Verification, Korea Atomic Energy Research Institute, KAERI/TR-1386-99 (1999).

22. I. DAUBECHIES, *Ten Lectures on Wavelets*, SIAM (1992).

Table 1. Relative information of each feature component.

Feature component	Pressurizer water level	Pressurizer pressure	Hot-leg temperature
1st	5.3426e+001	6.0332e+001	5.2266e+001
2nd	2.5443e+001	2.1864e+001	2.5851e+001
3rd	1.6673e+001	1.3833e+001	1.6873e+001
4th	4.3458e+000	3.8386e+000	4.8907e+000
5th	8.7525e-002	1.0310e-001	9.3261e-002
6th	1.4765e-002	1.7005e-002	1.6033e-002
7th	4.8999e-003	6.6709e-003	6.1960e-003
8th	3.6737e-003	3.6858e-003	3.6812e-003
9th	3.7611e-004	8.6696e-004	3.7699e-004
10th	2.9762e-004	2.8128e-004	3.0072e-004

Table 2. Correlation coefficient matrix for collected signals.

	SF	FF	SP	ST	NL	WL	HT	CT	PP	PL	PT
steam flowrate(SF)	1.0000	0.9894	-0.9571	-0.9569	-0.0015	-0.0007	0.9850	-0.4840	-0.1421	0.9542	-0.1422
feed flowrate(FF)	0.9894	1.0000	-0.9353	-0.9347	-0.0880	-0.0875	0.9837	-0.4352	-0.0906	0.9595	-0.0909
steam pres. (SP)	-0.9571	-0.9353	1.0000	1.0000	-0.0249	-0.0257	-0.9088	0.7139	0.2935	-0.8327	0.2938
steam temp.(ST)	-0.9569	-0.9347	1.0000	1.0000	-0.0265	-0.0273	-0.9084	0.7147	0.2940	-0.8321	0.2944
S/G water level(NL)	-0.0015	-0.0880	-0.0249	-0.0265	1.0000	0.9996	-0.0325	-0.1094	-0.1965	-0.0436	-0.1957
S/G wide-range level(WL)	-0.0007	-0.0875	-0.0257	-0.0273	0.9996	1.0000	-0.0319	-0.1101	-0.1980	-0.0431	-0.1972
hot-leg temp.(HT)	0.9850	0.9837	-0.9088	-0.9084	-0.0325	-0.0319	1.0000	-0.3647	0.0176	0.9849	0.0175
cold-leg temp.(CT)	-0.4840	-0.4352	0.7139	0.7147	-0.1094	-0.1101	-0.3647	1.0000	0.5769	-0.2089	0.5777
PRZR pressure(PP)	-0.1421	-0.0906	0.2935	0.2940	-0.1965	-0.1980	0.0176	0.5769	1.0000	0.0713	1.0000
PRZR water level(PL)	0.9542	0.9595	-0.8327	-0.8321	-0.0436	-0.0431	0.9849	-0.2089	0.0713	1.0000	0.0714
PRZR temp. (PT)	-0.1422	-0.0909	0.2938	0.2944	-0.1957	-0.1972	0.0175	0.5777	1.0000	0.0714	1.0000

Table 3. Estimation results of the neuro-fuzzy inference system for each sensor signal.

(a) Pressurizer water level

Denoising application		Yes	Yes	No	No
PCA application		Yes	No	Yes	No
Training data	Maximum error [%]	1.8492e-002	1.5167e-002	6.5160e-002	6.0784e-002
	Standard deviation of residuals	8.6789e-003	7.9163e-003	2.9507e-002	2.5684e-002
	Fitness	8.7238e-001	8.6383e-001	7.5001e-001	7.4708e-001
Verification data	Maximum error [%]	1.9034e-002	1.5358e-002	1.1175e-001	1.0037e-001
	Standard deviation of residuals	8.6456e-003	7.8484e-003	3.0270e-002	2.8221e-002
Number of neuro-fuzzy system inputs		4	5	4	5
Used signals		4 PC's	HT,CT,PP FF,WL	4 PC's	HT,CT,PP FF,WL

(b) Pressurizer pressure

Denoising application		Yes	Yes	No	No
PCA application		Yes	No	Yes	No
Training data	Maximum error [%]	1.9353e-002	1.5018e-002	7.4203e-002	7.5069e-002
	Standard deviation of residuals	7.3319e-003	6.7849e-003	2.9633e-002	2.9889e-002
	Fitness	8.7039e-001	8.6442e-001	7.2986e-001	7.1340e-001
Verification data	Maximum error [%]	1.9955e-002	1.7090e-002	9.5220e-002	7.1340e-001
	Standard deviation of residuals	7.2636e-003	6.7775e-003	9.5220e-002	2.9494e-002
Number of neuro-fuzzy system inputs		4	5	4	5
Used signals		4 PC's	PT,PL,CT ST,NL	4 PC's	PT,PL,CT ST,NL

(c) Hot-leg temperature

Denoising application		Yes	Yes	No	No
PCA application		Yes	No	Yes	No
Training data	Maximum error [%]	1.9788e-002	1.4271e-002	4.6199e-002	4.8380e-002
	Standard deviation of residuals	7.8318e-003	5.9689e-003	2.1602e-002	2.1242e-002
	Fitness	8.6917e-001	8.6648e-001	7.9841e-001	7.7767e-001
Verification data	Maximum error [%]	2.0310e-002	1.5030e-002	6.1645e-002	6.8607e-002
	Standard deviation of residuals	7.8381e-003	6.0023e-003	2.2931e-002	2.3732e-002
Number of neuro-fuzzy system inputs		4	5	4	5
Used signals		4 PC's	CT,PP, PL,ST,WL	4 PC's	CT,PP, PL,ST,WL

Table 4. Failure detection times for various degradation types in each sensor application.

		Detection times [sec]			
Denoising application		Yes	Yes	No	No
PCA application		Yes	No	Yes	No
Pressurizer water level	Gradual bias degradation	42	41	95	74
	Noise degradation	0	0	50	50
	Abrupt degradation	0	0	6	4
Pressurizer pressure	Gradual bias degradation	45	44	82	83
	Noise degradation	0	0	32	49
	Abrupt degradation	0	0	2	2
Hot-leg temperature	Gradual bias degradation	35	28	78	70
	Noise degradation	0	0	28	49
	Abrupt degradation	0	0	4	4

Table 5. Determining whether the output is degraded or not when one of only input signals (not output) is gradually degraded.

(a) Pressurizer water level

Degraded input	SF	FF	SP	ST	NL	WL	HT	CT	PP	PT
Output failure detection time	88	77	170	171	121	121	82	72	not found[1]	not found
Output failure detection time in case the degraded input is assumed to be an output	27	30	45	31	25	36	35	16	45	58

1. The output sensor is not decided to be degraded until final simulation time (500 sec after the beginning of the gradual degradation) is reached.

(b) Pressurizer pressure

Degraded input	SF	FF	SP	ST	NL	WL	HT	CT	PL	PT
Output failure detection time	478	not found	not found	not found	not found	not found	194	358	not found	42
Output failure detection time in case the degraded input is assumed to be an output	27	30	45	31	25	36	35	16	42	58

(c) Hot-leg temperature

Degraded input	SF	FF	SP	ST	NL	WL	CT	PP	PL	PT
Output failure detection time	113	91	111	122	not found	not found	258	490	79	492
Output failure detection time in case the degraded input is assumed to be an output	27	30	45	31	25	36	16	45	42	58

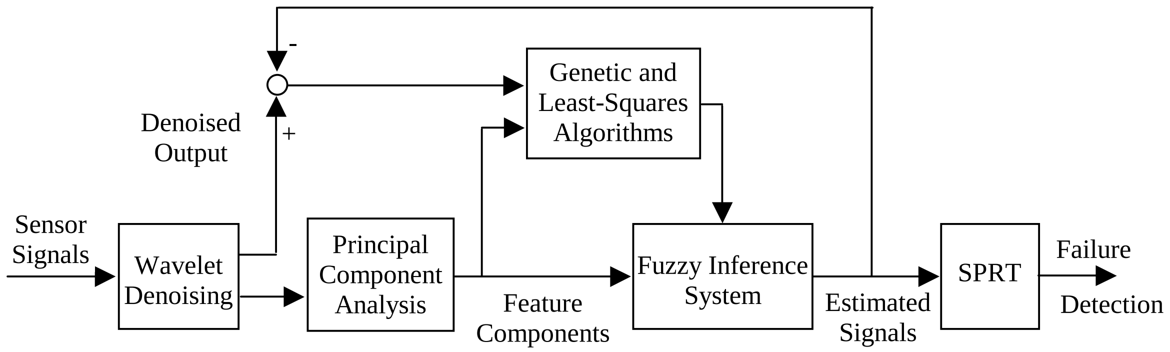

Fig. 1. Schematic diagram of the proposed sensor-monitoring algorithm.

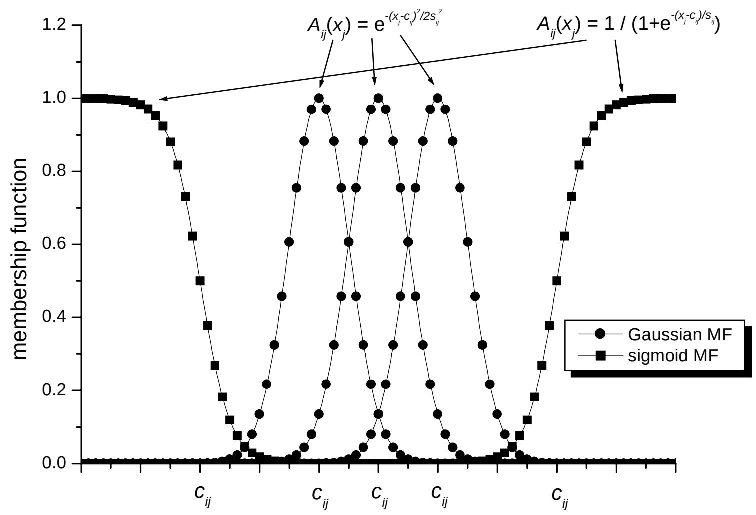

Fig. 2 Membership functions.

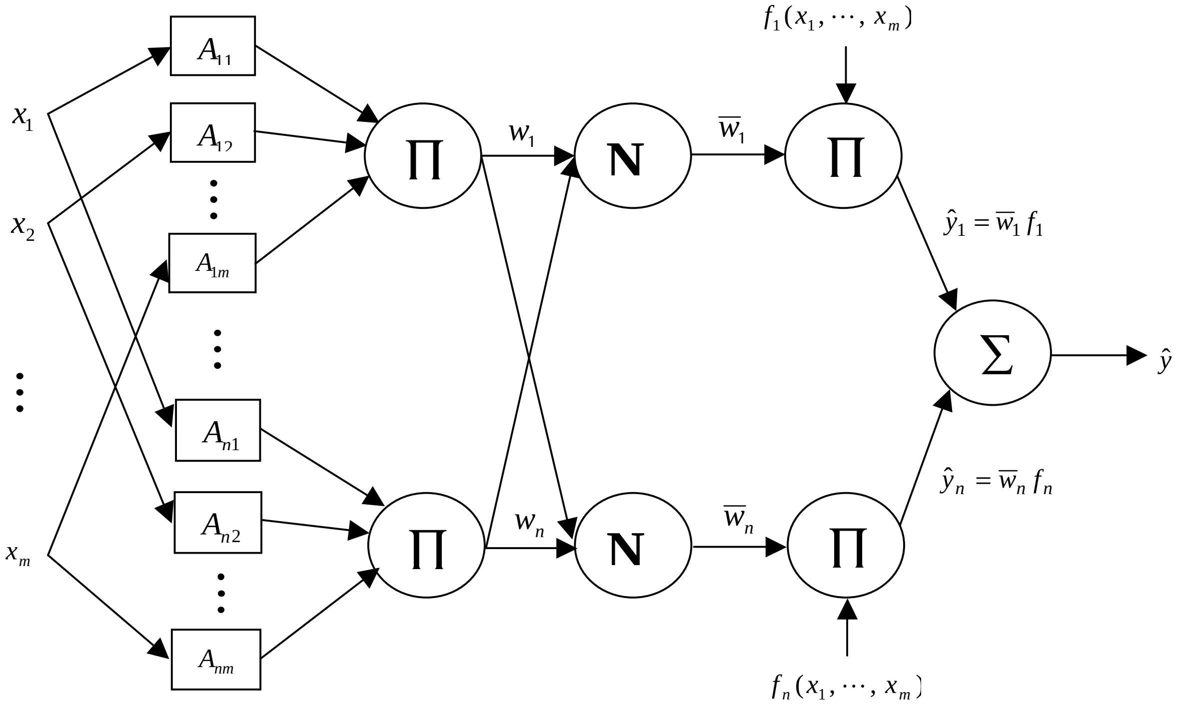

Fig. 3. A fuzzy inference system.

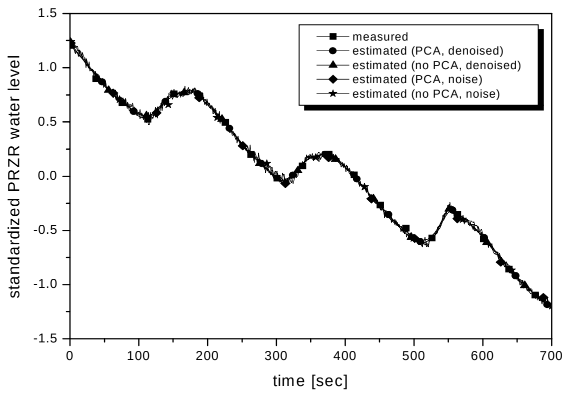

Fig. 4. Estimation of the pressurizer water level (using the verification data that were not used in the training stage).

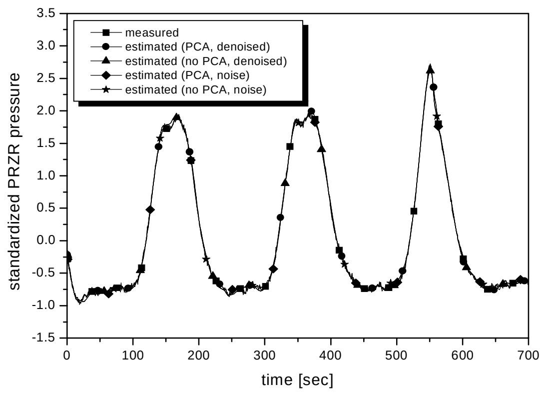

Fig. 5. Estimation of the pressurizer pressure (using the verification data that were not used in the training stage).

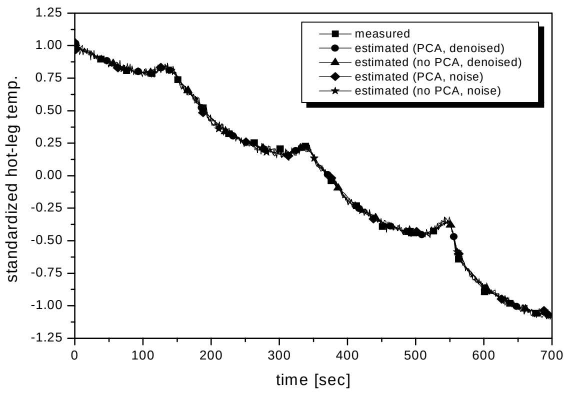

Fig. 6. Estimation of the hot-leg temperature (using the verification data that were not used in the training stage).

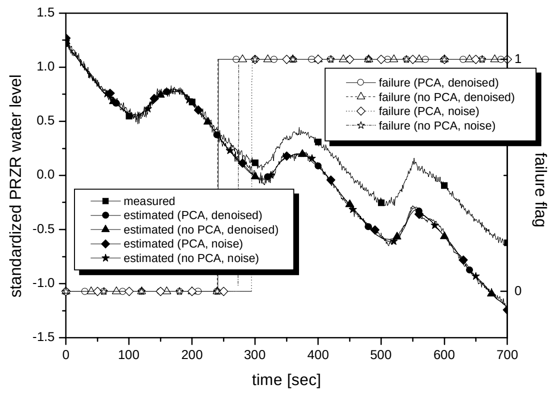

Fig. 7. Gradual bias degradation detection of the pressurizer water level sensor.

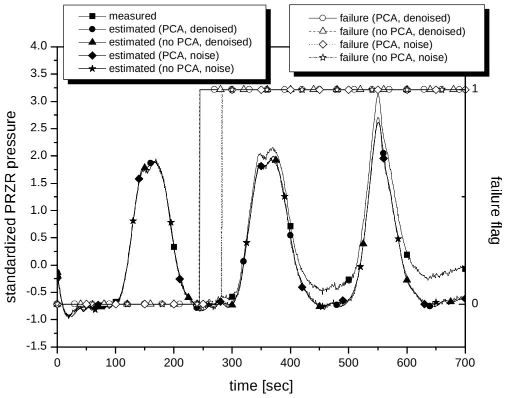

Fig. 8. Gradual bias degradation detection of the pressurizer pressure sensor.

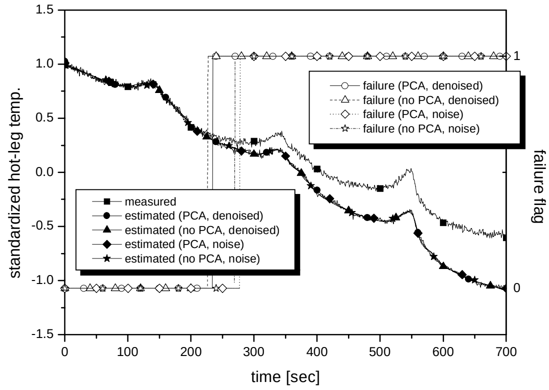

Fig. 9. Gradual bias degradation detection of the hot-leg temperature sensor.

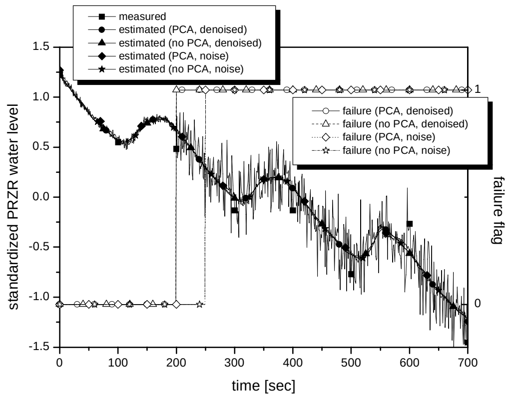

Fig. 10. Noise degradation detection of the pressurizer water level sensor.

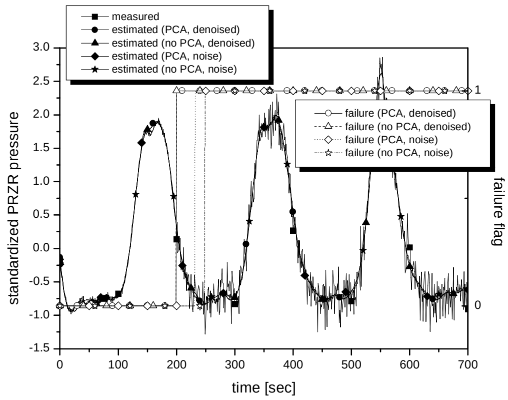

Fig. 11. Noise degradation detection of the pressurizer pressure sensor.

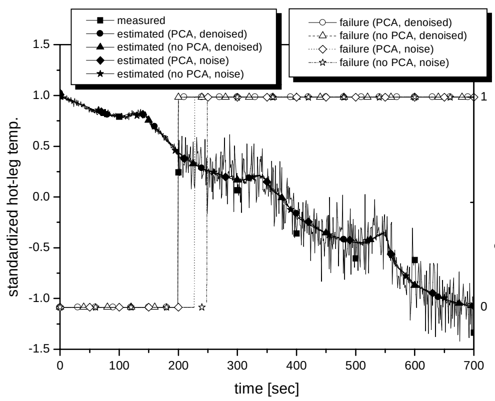

Fig. 12. Noise degradation detection of the hot-leg temperature sensor.

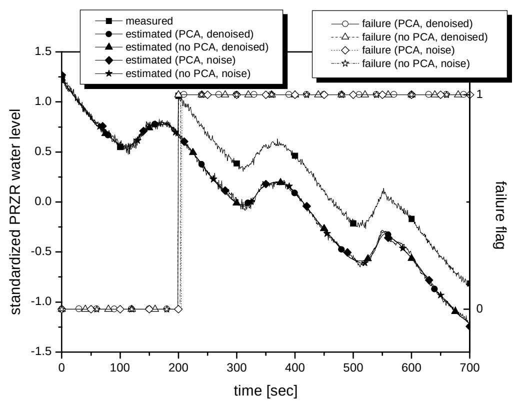

Fig. 13. Abrupt degradation detection of the pressurizer water level sensor.

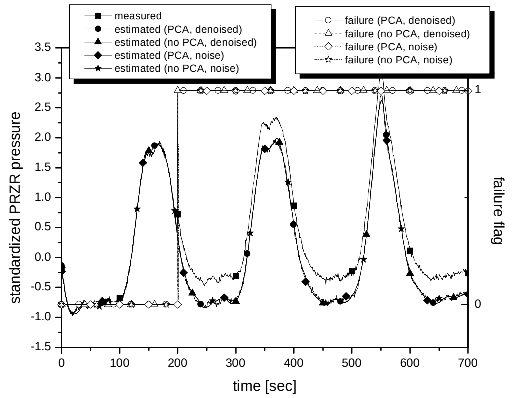

Fig. 14. Abrupt degradation detection of the pressurizer pressure sensor.

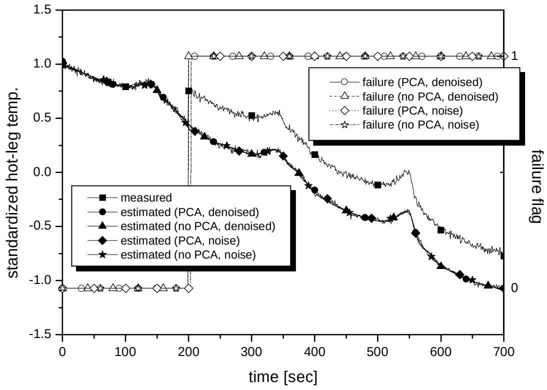

Fig. 15. Abrupt degradation detection of the hot-leg temperature sensor.