Robust Recursive Principal Component Analysis Modeling for Adaptive Monitoring

Hyung Dae Jin,[†] Young-Hak Lee,[‡] Gibaek Lee,[§] and Chonghun Han*,[||]

Wastes Pyrolysis Research Center, Korea Institute of Energy Research, Jang-dong, Yuseong-gu 71-2, DaeJeon 305-343, Korea, Automation Systems and Research Institute and School of Chemical and Biological Engineering, Seoul National University, San 56-1, Shillim-dong, Gwanak-gu, Seoul 151-742, Korea, Department of Chemical Engineering, Chungju National University, Chungju, Chungbuk 380-702, Korea, and Institute of Chemical Processes, School of Chemical and Biological Engineering, Seoul National University, San 56-1, Shillim-dong, Gwanak-gu, Seoul 151-742, Korea

This paper proposes a robust recursive principal component analysis (PCA) modeling procedure that aims to improve the monitoring performance by detecting and identifying process changes, removing disturbances, and updating the model to reflect the operating mode change. The proposed approach was applied to an industrial fired heater. Compared with previous approaches based on conventional PCA or recursive PCA, this new procedure demonstrated improved monitoring performance. The case study shows that both the number of false alarms and the number of model updates were significantly reduced in comparison with previous methods.

Downloaded via DUKE UNIV on July 11, 2025 at 01:42:15 (UTC).
See https://pubs.acs.org/sharingguidelines for options on how to legitimately share published articles.

1. Introduction

Process monitoring has been widely employed to guarantee process safety and product quality and to identify the state of a process.[1] Process monitoring is required both to detect process changes as early as possible and to reduce the number of false alarms.[2] Multivariate statistical process control (MSPC) based on principal component analysis (PCA) has been widely applied to satisfy these key demands. However, in real industrial processes, process changes frequently occur due to variations in the demand for products, fluctuations in raw materials, and changes in utility prices. MSPC is difficult to apply to such a process with nonstationary and time-varying behavior.[1,3,4]

Wold[5] and Gallagher[6] introduced the use of exponentially weighted moving average (EWMA), exponentially weighted moving covariance (EWMC), and exponentially weighted moving PCA (EWM-PCA). However, these approaches do not reflect the process changes well because EWMA, EWMC, and EWM-PCA iteratively apply exponentially decreasing weights without any identification for process changes. As another alternative, model library based methods have been introduced.[4,7,8] Predefined models match their corresponding modes. However, the approach has the limitation that the operating modes are not fixed. Therefore, the model library should still be updated continuously to take into account the generation of new operating modes. Li[3] presented a monitoring strategy that builds a recursive PCA (RPCA) model with a moving window. The moving window approach based on the recursive method carries out *blind updates*, which means that continuous updates are performed without regard to whether a process change has been identified. That is, when the process exceeds the control limits, RPCA updates the current model without discriminating between operating mode changes and disturbances. In fact, in a case in which the process state goes out of the control limits due to disturbances, it does not need to update the model. Therefore, it is doubtful whether the model is satifactorily updated.

In this paper, we propose a robust recursive PCA monitoring methodology for a process that includes frequent operating mode changes. The core idea of the proposed approach is to combine process knowledge for the detection of mode changes with statistical indices to seek an update starting point. This provides a more useful way of searching for optimal update points, due to the fact that decisions made with physical meanings avoid the errors of judgment that arise because of biased parameter estimation. The proposed approach introduces a similarity index to obtain a forgetting factor of recursive update and to find the update termination point by steady-state detection.

In section 2, theoretical background of recursive PCA is introduced. Section 3 proposes efficient model update schemes based on recursive PCA for the processes that are subject to nonstationary and time-varying conditions, such as frequent and sudden changes. Section 4 provides a process description of a fired heater process and the monitoring results of the proposed approach and compares the monitoring performance between the proposed approach and those of previous works. Section 5 describes the conclusions.

2. Recursive PCA (RPCA)

Static PCA has some major shortcomings. One of these weaknesses is not suitable for monitoring nonstationary and time-varying processes like most real industrial processes,[3] because once static PCA is built that means almost settled mean, variance, and covariance exist among variables. Therefore, a new method is required to adapt the monitoring model in a continuous and automatic manner in compliance with present process condition. This can be carried out by using EWMA— or a moving window-based approach including historical data. This method is RPCA. RPCA can handle slow time-varying properties successfully. Its update is executed to on-line monitoring as new data is obtained, allowing changes in the mean, variance, and correlation of the measurements to be followed.[1]

* To whom correspondence should be addressed. Tel.: +82-2-880-1887. Fax: +82-2-888-7295. E-mail: chhan@snu.ac.kr.
[†] Korea Institute of Energy Research.
[‡] Automation Systems and Research Institute and School of Chemical and Biological Engineering, Seoul National University.
[§] Chungju National University.
[||] Institute of Chemical Processes, School of Chemical and Biological Engineering, Seoul National University.

10.1021/ie050850t CCC: $33.50 © 2006 American Chemical Society
Published on Web 12/17/2005

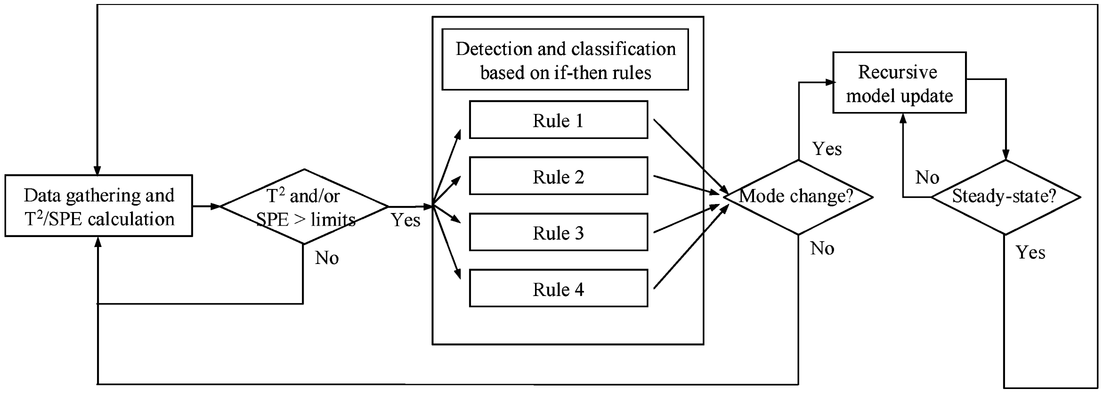

Figure 1. Overall procedure of the proposed robust RPCA modeling.

The normalized matrix, **X**, can be approximated using the loading matrix, **P**, the score matrix, **T**, and the residual matrix, **E**. Two of these matrices, the loading matrix and the score matrix, can be separated into a loading vector p_i and a score vector t_i.

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E} = \sum_{i=1}^{k} t_i p_i^{\mathrm{T}} + \mathbf{E} \tag{1}$$

$$\mathbf{T}_i^{\,2} = x_i \mathbf{P} \Lambda^{-1} \mathbf{P}^{\mathrm{T}} x_i^{\,2} \tag{2}$$

$$Q_i = x_i(\mathbf{I} - \mathbf{P}_k \mathbf{P}_k^{\,-1}) x_i^{\mathrm{T}} \tag{3}$$

where x_i is the ith sample in **X** and the column of $\mathbf{P}_k$ is the first k loadings vector retained in the PCA model. Λ_i is a diagonal matrix containing the eigenvalues associated with the k eigenvectors (principal components), and **I** is an identity matrix.[6]

Though RPCA is worth attempting for monitoring nonstationary and time-varying processes, RPCA can handle successfully not only frequent, large, and sudden process changes but also slow time-varying properties. Besides, if long disturbances are generated, continuous and automatic model updates without identification of process changes lead to misleading results because of the adaptation of disturbances; if a process lies in steady state, it leads to unnecessary loads because of useless updates as well.

3. The Proposed Approach

A new approach must satisfy the following four conditions for enhancing adaptive monitoring: (i) to determine either normal states or disturbances, (ii) to consider either significant or insignificant data, (iii) to detect necessary update points, and (iv) to maintain the model longer than ever before. Figure 1 shows the overview of the proposed approach which consists of (i) the detection of process changes by means of two indices, Hotelling's T^2 and the squared prediction error (SPE), (ii) the identification between mode changes and disturbances using a priori process knowledge, (iii) the removal of disturbance data so as to prevent the model from adapting to disturbances, and (iv) the update of the model using a similarity index and the detection of steady-state conditions in order to stop the model adaptation.

3.1. Detection and Identification of Process Changes. Process changes generally include operating mode changes, start-ups or shut-downs, and disturbances. It is easy to detect start-ups or shut-downs; therefore, this study is limited to the detection of mode changes and disturbances.[9,10] Disturbance adaptations lead to biased parameter estimation, as well as incorrect analysis results. SPC and MSPC are powerful problem-solving tools for detecting process variability.[11] MSPC based on PCA is a useful monitoring method for dealing with the detection of process changes in complex chemical processes.[11] Detection points, which are process changes, are considered to be candidate time points for model updates.

Process changes tend to be against the process operating point outside of the control limits, T^2 and SPE, because of different correlation relationships between variables. The control limits of detection criteria for process changes are given by

$$T_{\mathrm{lim}}^{\ \ 2} = \frac{k(N-1)}{(N-k)} F_{k,N-k,\alpha} \tag{4}$$

$$\mathrm{SPE}_{\mathrm{lim}} = \left(\frac{\upsilon}{2m}\right) \chi_{2m^2/\upsilon,\alpha}^{\ \ 2} \tag{5}$$

where $F_{k,N-k,\alpha}$ is the critical value of the F-distribution with $k,N - k$ degrees of freedom at the confidence region, α. The terms m and υ are the sample mean and variance of the model training set, and $\chi_{2m^2/\upsilon,\alpha}^{\ 2}$ is the critical value of the χ-squared variable with $2m^2/\upsilon$ degrees of freedom at the significant level, α. T_i^2 provides a measure of the variation within the PCA model for each observation,[4] while SPE provides a measure of how well the new observation is described by the PCA model.[12]

However, variations of process changes can only be confirmed as being mode changes or disturbances once the process changes are completed. One solution to this problem is offered by a new method which provides for the real-time identification of process changes.[9,10]

Lee, Jin, and Han proposed if−then rules for the detection and classification of process changes based on a priori process knowledge.[9,10] Rule generation steps constitute (i) extraction of possible main factors for operating mode changes, (ii) division of a historical data matrix into subdata matrices with the possible main factors, (iii) identification of main factors and removal of pointless factors, that is, decision about the main factors of operating mode changes. In other words, if there is a large process change and this change is not a disturbance but an operating mode change, that is, checked by an operator and process knowledge through the extracted main factors, then a verification step is carried out.

The form of the if−then rules used for detection is as follows:

$$\begin{aligned}&\text{IF } \{\text{change of factor}_i \text{ and change of effect of causal}\\ &\quad \text{factor}_i \text{ and identification of no disturbance (factor}_i)\},\\ &\qquad \text{THEN operating mode change caused by factor}_i\end{aligned} \tag{6}$$

To identify whether a deviation from nominal operating conditions is an operating mode change or not, the tabular

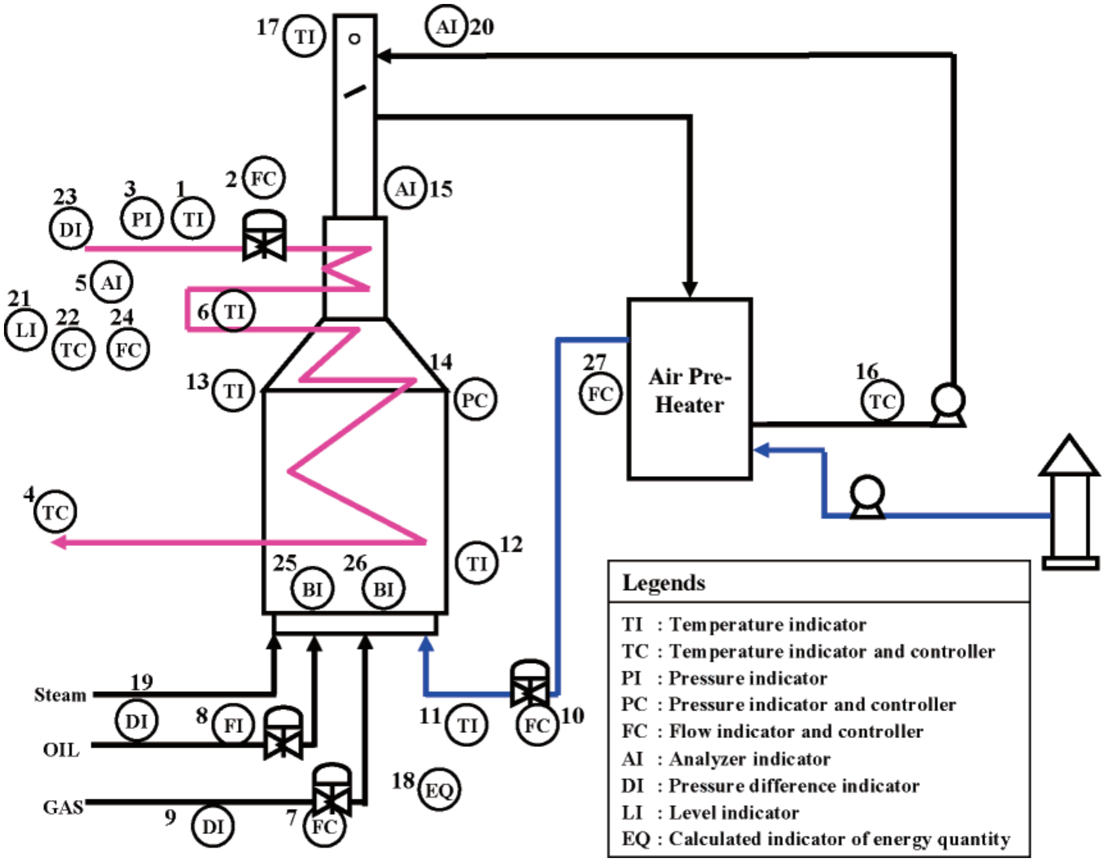

Figure 2. Schematic diagram of the fired heater process.

CUSUM is used, since this control chart is widely used to detect process changes quickly and allows trivial changes and noise to be to filtered out.[13] If eq 6 is accepted, the switch from one state to another state can be concluded to be a mode change.

In conclusion, if a process change is detected by two indices, T^2 and SPE, the if−then rules begin to identify process changes, whether the current change is a mode change or not. If a mode change occurs, variables of the main factors that represent the root causes in the if−then rules should exceed the control limits.

3.2. Model Update. The steps required for a model update consist of (i) isolating disturbances, (ii) updating the model using the similarity index, and (iii) stopping adaptation through the detection of steady-state conditions.

3.2.1. Disturbance Isolation. Disturbances are deviations that lie significantly outside of the normal region and can be caused by start-ups or shut-downs, sensor malfunctions, process disturbances, instrument degradation, and human-related errors.[13] The proposed approach is based on the PCA model of MSPC. Two major monitoring indices are calculated, namely, T^2 and SPE. An abnormal situation will cause at least one of these two values, T^2 and SPE, to deviate from the control limits.

If the process changes are disturbances, at least, T^2 or SPE will deviate from the limits and the if−then rules will not be accepted. On the other hand, if the process changes are operating mode changes, investigation and corrective action are required to find and eliminate the causes of variability. Therefore, data representing disturbance should be isolated, since it constitutes irrelevant information.

The ultimate goal of MSPC is the elimination of variability in the process. It may not be possible to completely eliminate variability, but the use of control limits provides an effective method of reducing variability as much as possible.[10] Therefore, this helps to keep the model from adapting to insignificant information, thereby reducing the number of updates and false alarms.

3.2.2. Start Update. Update according to each quantitative magnitude of process change. For example, if sudden and frequent process changes start from the ith time interval, the history data for the $(i-1)$th and $(i-2)$th time intervals would have relatively little pertinent information, as compared with the case in which the changes are less sudden and frequent. To cope with this problem, metrics were introduced.

In this study, a similarity index, D, is used for process changes to reflect metrics. The similarity index is calculated from the value of D.[2] The value of D is given by

$$R_i = \frac{1}{N_i - 1}\mathbf{X}_i^{\mathrm{T}}\mathbf{X}_i \tag{7}$$

$$R = \frac{1}{N-1}[\mathbf{X}_1\ \mathbf{X}_2]^{\mathrm{T}}[\mathbf{X}_1\ \mathbf{X}_2] \tag{8}$$

$$\mathbf{Y}_i = \sqrt{\frac{N_i - 1}{N - 1}}\mathbf{X}_i\mathbf{P}_{\mathrm{o}}\Lambda^{-(1/2)} \tag{9}$$

$$\mathbf{S}_i = \frac{1}{N_i - 1}\mathbf{Y}_i^{\mathrm{T}}\mathbf{Y}_i \tag{10}$$

$$D_i = \frac{4}{k}\sum_{j=1}^{k}(\lambda_j - 0.5)^2 \tag{11}$$

where N_i is the number of ith data set samples, $\mathbf{P}_{\mathrm{o}}$ is an orthogonal matrix of R, Λ is a diagonal matrix whose diagonal elements are eigenvalues of R, λ consists of the eigenvalues of the covariance matrix of $\mathbf{S}$, and D_i is used to quantitatively

Table 1. Variables Used for PCA Modeling and Rule Checking

variable no.	description
1	feed inlet temperature
2	feed flowrate
3	feed pressure
4	feed out temperature
5	deviation of feed passflow
6	feed outlet temperature
7	fuel gas flowrate
8	fuel oil flowrate
9	density of gas
10	air flowrate
11	air temperature
12	radiation in heatbox
13	convection in heatbox
14	pressure in heatbox
15	excess O_2
16	APH out temperature
17	heatbox out temperature
18	total energy consumed
19	pressure difference between oil and steam
20	NO*x*
21	level of feed tank
22	feed out temperature (set point)
23	feed density
24	feed flowrate (set point)
25	number of gas burners used
26	number of oil burners used
27	air flowrate (set point)

evaluate the difference in the covariance between the $(i - 1)$th data set and the ith data set.

The model is updated using eqs 12 and 13,[11] whenever the process change is identified as a mode change by means of the if−then rule and two major indices. That is, D is applied in the form of a variable, rather than a constant, whose value changes according to the quantitative magnitude of the process change.

$$T_{i+1}^{\ 2} = D_i T_i^{\ 2} + (1 - D_i) T_{i-1}^{\ 2} \tag{12}$$

$$\mathrm{SPE}_{i+1} = D_i \mathrm{SPE}_i + (1 - D_i)\mathrm{SPE}_{i-1} \tag{13}$$

where D_i is used as a forgetting factor.

3.2.3. Update Termination by Steady-State Detection. Define the model adaptation stopping point. Generally, if the process is assumed to be in control, no action is required at all. As in the case of blind updates, updating the model is unnecessary at steady state. Besides, performing a blind update would incur certain risks, such as adapting to disturbances. One solution to this problem is to continue monitoring without performing a model update until the next mode change occurs. A steady state is a situation in which no process changes occur, and generated process data would be stationary and time-invariant. To define the criterion to stop model adaptation, D is used as a metric of steady-state conditions.

To detect changes in the distribution of process data, reference distributions representing normal operating conditions are defined, and the values of the dissimilarity index among the reference distributions are obtained. The maximum limit of those values can be set as a threshold for steady-state detection. Steady-state data of ith window size are denoted as $\mathbf{M}_i$. The dissimilarity index between $\mathbf{M}_i$ and $\mathbf{M}_{i+1}$, $\hat{D}_i$, is defined as

$$\hat{D}_i = \mathrm{DISSIM}(\mathbf{M}_i, \mathbf{M}_{i+1}) \tag{14}$$

where the function DISSIM is a symbol that compresses the operations from eqs 7−11. The threshold for steady-state detection is equal to

$$\hat{D}_c = \max\{\hat{D}_i | i = 1,2,3,...\} \tag{15}$$

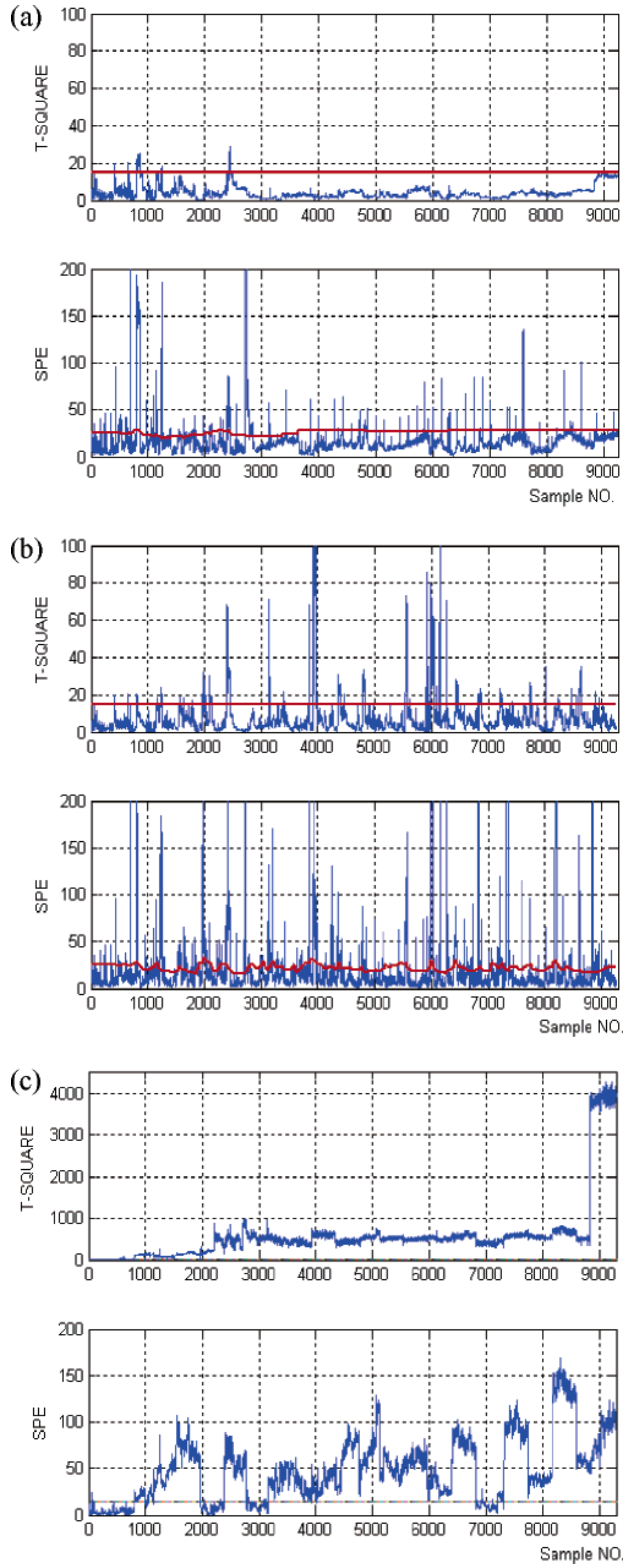

Figure 3. Monitoring results including various disturbances and mode changes: (a) robust RPCA; (b) recursive PCA; (c) conventional PCA.

where the function max means extracting a maximum value from the elements in a set.

The main steps required for model updating with the proposed approach are summarized, as follows: (i) Execute the

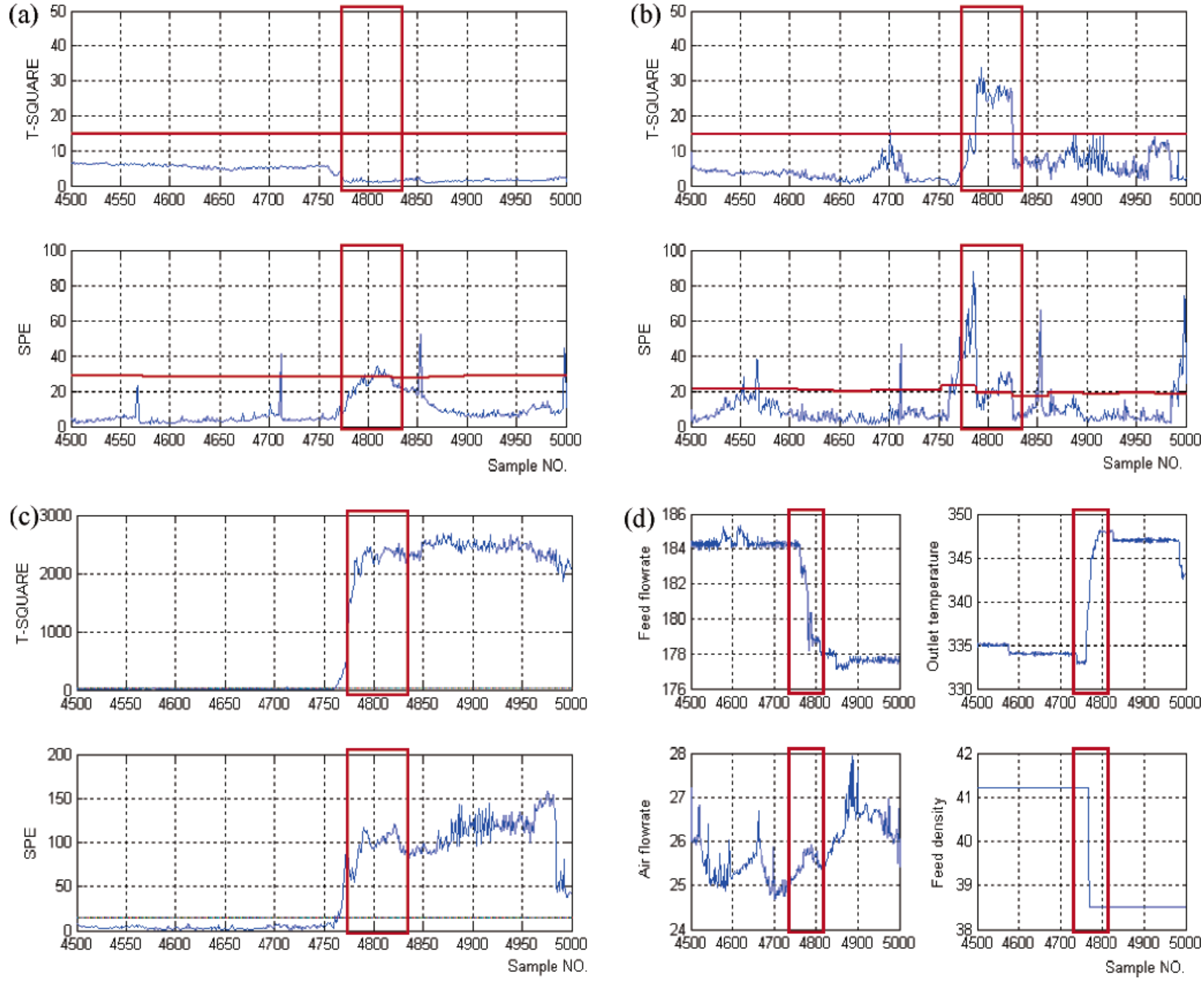

Figure 4. Monitoring results including a normal mode change: (a) robust RPCA; (b) recursive PCA; (c) conventional PCA; (d) variables used for the detection of mode change.

initial PCA with window-size data. (ii) Collect block-size data and detect process changes from eqs 4 and 5. (iii) Identify mode changes using the rules formed by eq 6. If this change is not due to a mode change, ignore these data and return to step 2. (iv) Update the PCA model based on the new significant data until the value of D is smaller than $\hat{D}_c$. Then, go to step 2.

4. Results and Discussion

4.1. Process Description and Rule Definition. An industrial fired heater is used to demonstrate the validity of the proposed approach. The industrial fired heater is shown in Figure 2. Burners generate the heat through the combustion of fuel. Oil and gas are consumed as fuel to heat the oil being fed. The feed absorbs the heat by radiant and convective heat transfer from the flue gases. The flue gases are vented to the ambient air through the stack. To increase the heat efficiency, an air preheater is installed downstream of the convection section. The test data were gathered over a period of 2 months. The window-size used is 3 days, and the block-size is 6 h. As shown in Table 1, 27 variables are measured, including the process variables and set points. The variables from 1 to 21 are associated with the data used for PCA modeling; the remaining 6 variables are associated with the data used for rule checking.

The model update criteria include the if−then rules formed by eq 6. First, a change in the set point of the outlet temperature indicates a transition from one mode to another. To detect a change, we devised the following rule:

$$\text{Rule 1.}\quad \text{if } \{|\text{median}(\mathbf{T}_{\text{Out},t}{}^{\text{sp}}) - T_{\text{Out},0}{}^{\text{sp}}| > 0\} \qquad (16)$$

where $\mathbf{T}_{\text{Out},t}{}^{\text{sp}}$ is a data vector of window-size for the feed outlet temperature set point at time t and $T_{\text{Out},0}{}^{\text{sp}}$ is the feed outlet temperature set point value in the original operating mode.

Second, a change in the feed density was detected using the CUSUM algorithm. The air flowrate was selected as an effect variable. An increase of the density induced an increase in the air flowrate. This is why the heat duty in the heater was heavier. The rule for this procedure is as follows:

$$\text{Rule 2.}\quad \text{if } \{|\text{CUSUM of median}(\mathbf{D}_{\text{F},t})| > \text{control limits and } |\text{CUSUM of median}(\mathbf{F}_{\text{A},t})| > \text{control limits}\} \qquad (17)$$

where $\mathbf{D}_{F,t}$ is a data vector of window-size for the feed density at time t and $\mathbf{F}_{A,t}$ is a data vector of window-size for the air flow rate at time t.

Third, the CUSUM algorithm detects a mode change for the feed and air flowrates. The corresponding rule is defined as follows:

$$\text{Rule 3.}\quad \text{if } \{|\text{CUSUM of median}(\mathbf{F}_{F,t})| > \text{control limits and } |\text{CUSUM of median}(\mathbf{F}_{A,t})| > \text{control limits}\} \quad (18)$$

$\mathbf{F}_{F,t}$ is a data vector of window-size for the feed flow rate at time t.

Finally, the fuel gas density could not be measured in this case. To detect a change in that factor, the total number of burners firing fuel oil and gas was used. The number of burners consuming fuel gas decreases if the quality of the fuel gas entering the fired heater increases while the number of burners for fuel oil increases. Therefore, the number of burners consuming fuel gas or fuel oil is changed, which results in a mode change. The logic for this is as follows:

$$\text{Rule 4.}\quad \text{if } \{|\text{median}(\mathbf{B}_{O,t}) - (B_{O,0})| > 0 \text{ or } |\text{median}(\mathbf{B}_{G,t}) - (B_{G,0})| > 0\} \quad (19)$$

where $\mathbf{B}_{O,t}$ and $\mathbf{B}_{G,t}$ are data vectors of window-size for the number of burners firing fuel oil and gas, at time t and $B_{O,0}$ and $B_{G,0}$ is the number of burners firing fuel oil and gas in the original operating mode, respectively.[9,10]

4.2. Monitoring with the Proposed Approach and Comparison with the Previous Methods. Figure 3 shows the results of the monitoring performed during the 2 month period. In Figure 3a, there are 39 mode changes and a model update is performed 52 times as a result of the model update criteria being met.

D is calculated to reflect the process change well until a steady state is reached, when a mode is detected. From eq 15, the criterion to be met is that *D* should be smaller than 0.4579. Figure 3 parts b and c show the result of the recursive approach and the conventional PCA. In terms of model accuracy, it can be seen in Figure 3b that recursive PCA monitoring gives rise to more false alarms although a model update is performed 245 times, which is more than the proposed approach, for which the results are shown in Figure 3a. As shown in Figure 4, the proposed approach detects mode changes rapidly when mode changes occur. A mode change occurs at a sample number of approximately 4750. The equivalent regions of Figure 4 parts a and b show that the proposed approach can provide earlier detection of mode changes than the recursive PCA performing blind updates for values ranging from 4760 to 4800. Therefore, the early detection offered by the proposed approach considerably decreases the number of type I errors. The improvement in the model accuracy becomes clear when a mode change occurs, because there is evidence that four variables exceed the control limits in terms of their correlation and causality and that the conventional PCA exceeds the control limits; then, statistical indices are maintained. Therefore, Figure 4 parts c and d are proof that a mode change occurs. Figure 5 shows that, during the disturbance, the fuel oil flow rate oscillates from 4455 to 4465. Performing a model update after the isolation of the disturbance can almost completely rule out the possibility of disturbance adaptation. The equivalent regions of Figure 5 parts a and b show that recursive PCA is highly susceptible to adapting the model to disturbance, because the disturbances for values in the range 4450−4465 resulting from the if−then rule of the proposed approach are isolated. The proposed approach allows the number of type II errors to be reduced.

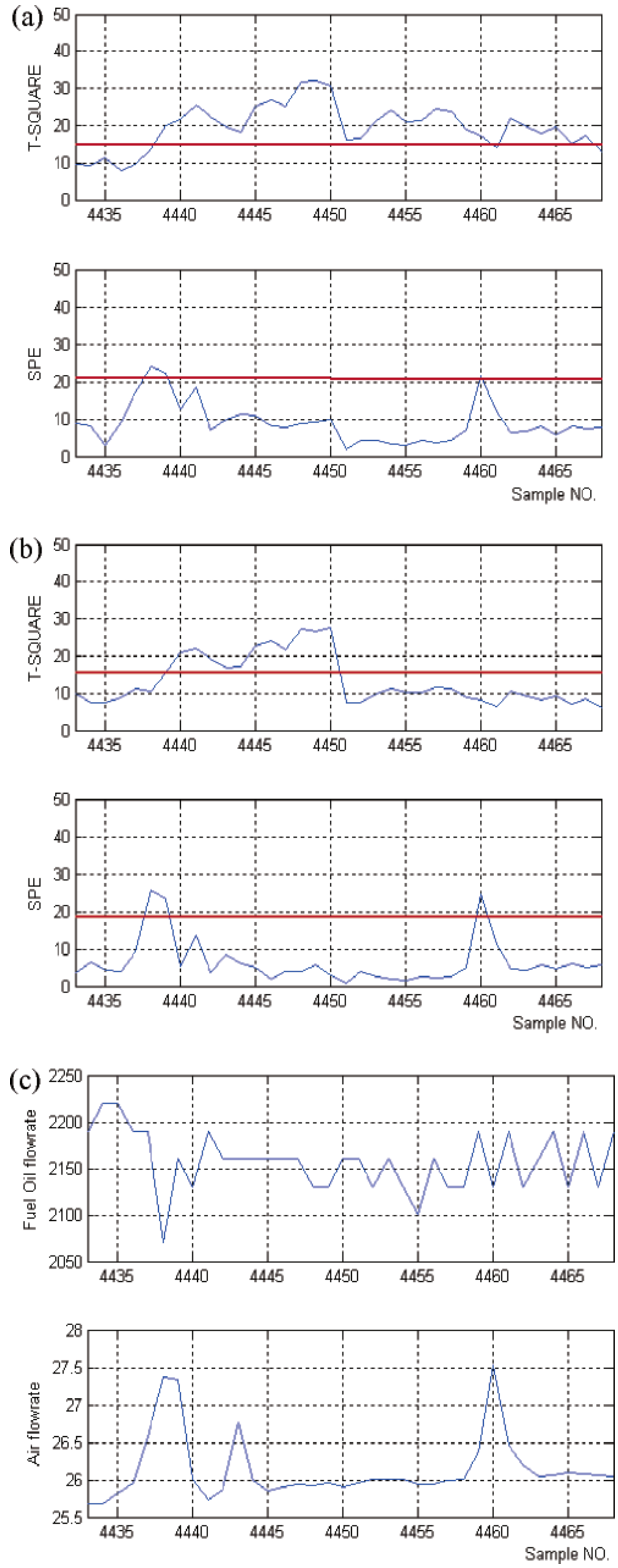

Figure 5. Monitoring results including a disturbance: (a) robust RPCA; (b) recursive PCA; (c) variables used for disturbance identification.

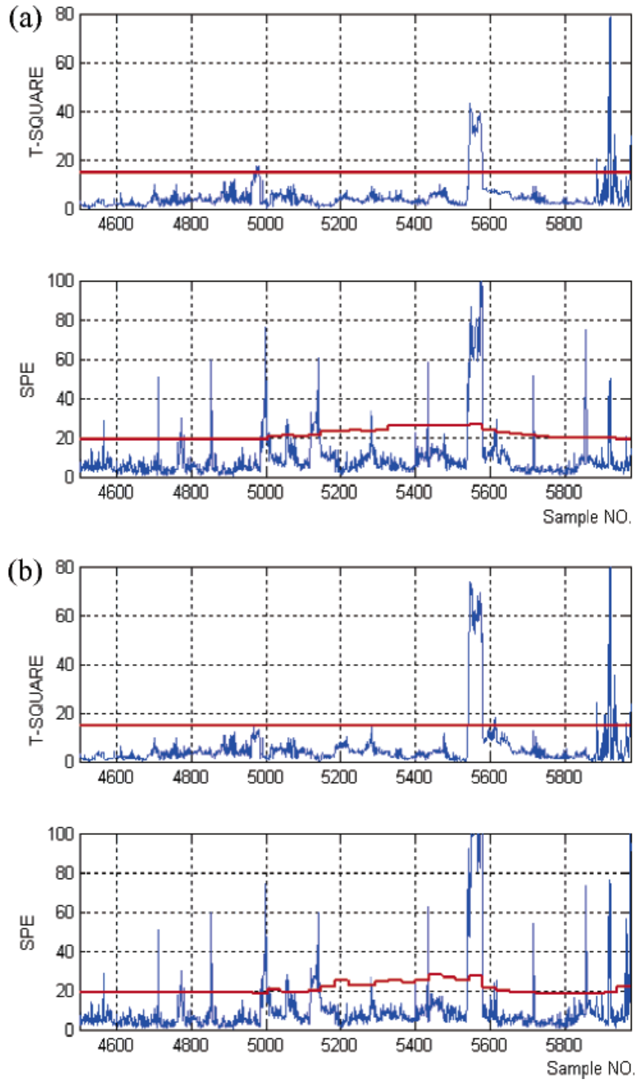

Figure 6. Monitoring results for the update performance: (a) robust RPCA, only 9 model updates; (b) recursive PCA, 29 model updates.

These two types of errors commonly arise when testing the null and alternative hypotheses: type I and II errors are given by[13]

$$\mathbf{P}(\text{type I error}) = \mathbf{P}(\text{reject } H_0|H_0 \text{ is true}) \tag{20}$$

$$\mathbf{P}(\text{type II error}) = \mathbf{P}(\text{reject } H_1|H_1 \text{ is true}) \tag{21}$$

where H_0 is the hypothesis that the current process change is equal to a mode change and H_1 is the hypothesis that the current process change is not equal to a mode change. The efficiency of the proposed approach can be seen in Figure 6. The number of model updates resulting from the proposed approach is just nine, because a model update is not performed for values in the approximate range 5350−5500 and at other points where no mode changes occur. In contrast, the number of model updates resulting from recursive PCA is 29. Although the proposed approach results in a lesser update load, the monitoring performance is similar to or better than the monitoring performance of recursive PCA, because the proposed approach focuses on process changes. In brief, the proposed approach improves the efficiency by about 4.7 times and reduces the error rate by about 5.8 times. Table 2 provides a summary of the results, demonstrating clearly that the proposed approach shows better performance.

Table 2. Comparison of the Proposed Robust RPCA with the Conventional RPCA

		RPCA	robust RPCA
number of normal samples: 8823	number of “accept” samples	7171	8475
	number of “reject” samples	1649	348
	type I error rate	0.2287	0.0411
number of abnormal samples: 127	number of “accept” samples	32	0
	number of “accept” samples	95	127
	type II error rate	0.3368	0
update efficiency	number of updates	245	52

5. Conclusions

The performance of the model can be assessed by means of the following factors: (a) whether it decreases both type I and type II errors or not and (b) whether fewer model updates are performed or not. Although the performance of the proposed approach depends on how well the process changes are separated into the two groups, mode changes and disturbances, this feasibility study shows that monitoring using the proposed approach offers sufficient efficiency. Since it does not identify the current status, recursive PCA monitoring leads to such shortcomings as adaptation to disturbances and unnecessary updates being performed under steady-state conditions. Moreover, since recursive PCA is based on the slow confirmation of process changes, not only are model updates performed at inappropriate times, but model adaptation also takes a lot of time. The proposed approach makes use of all kinds of process knowledge to determine the current status, with the result that this approach can be applied to a wide range of operating modes and various process conditions. The proposed approach enables both type I and type II errors to be reduced because the proposed approach is to continue monitoring without performing a model update until the next mode changes occurs; the number of updates is reduced, because the forgetting factor is applied in the form of a variable whose value changes according to the quantitative magnitude of the process change, and a model update is not performed at steady state, even under conditions of frequent and sudden process changes, thus, allowing end-user demand to be satisfied with respect to safety, product quality, and the optimal distribution of inventory.

Acknowledgment

The authors gratefully acknowledge the partial financial support of the Korea Science and Engineering Foundation provided through the Advanced Environmental Biotechnology Research Center (R11-2003-006-02003-0) and the Basic Research Program (R01-2004-000-10345-0). We are thankful for the partial financial support of the Brain Korea 21 program initiated by the Ministry of Education, Korea. This work is also partially funded by the Korea Institute of Science Technology (Development of Dried Liquid Fuel Cells), by Hyundai Motor Company and Korea Energy Management Corporation (Development of Dynamic Model and Optimized Operation Technology of Polymer Electrolyte Membrane Fuel Cell for Bus), and by 2005 Research Fund of Seoul National University.

Nomenclature

$\mathbf{X}$ = the normalized matrix
$\mathbf{T}$ = the score matrix
$\mathbf{P}$ = the loading matrix
R = the forgetting factor
Λ = the diagonal matrix containing the eigenvalues
T^2 = the Hotelling’s value
Q = the squared prediction error

$\mathbf{I}$ = the identity matrix
N = the number of observations in the model training set
F = the F-distribution
$\mathbf{P}_o$ = the orthogonal matrix of eigenvectors
D = the similarity index employed for model updating
$\hat{D}$ = the similarity index for steady-state data
C = the sample variance at modes
$\mathbf{S}$ = the data matrix at modes
t = the score vector
p = the loading vector
m = the sample mean of the model training set
v = the sample variance of the model training set
n = the number of samples at modes

Greek Letters

χ^2 = the χ-square distribution
α = the significant level
λ = the eigenvalues

Subscripts

c = threshold of an index value
k = the number of principal components
i = the index of data matrix order
j = the index of samples at modes

Literature Cited

(1) Rosen, C.; Lennox, J. A. Multivariate and Multiscale Monitoring of Wastewater Treatment Operation. *Water Res.* **2001**, *35*, 3402.

(2) Kano, M.; Hasebe, S.; Hashimoto, I.; Ohno, H. Statistical Process Monitoring Based on Dissimilarity of Process Data. *AIChE J.* **2002**, *48*, 1231.

(3) Li, W.; Yue, H. H.; Valle-Cervantes, S.; Qin, S. J. Recursive PCA for Adaptive Process Monitoring. *J. Process Control* **2000**, *10*, 471.

(4) Choi, S. W.; Park, J. H.; Lee, I. B. Process Monitoring Using a Gaussian Mixture Model via Principal Component Analysis and Discriminant Analysis. *Comput. Chem. Eng.* **2004**, *28*, 1377.

(5) Wold, S. Exponentially Weighted Moving Principal Components Analysis and Projections to Latent Structures. *Chemom. Intell. Lab. Syst*. **1994**, *23*, 149.

(6) Gallagher, N. B.; Wise, B. M.; Butler, S. W.; White, D. D.; Barna, G. G. Development and Benchmarking of Multivariate Statistical Process Control Tools for a Semiconductor Etch Process: Improving Robustness Through Model Updating. *Proceedings of IFAC workshop on ADCHEM ′97*, Banff, Canada, 1997; 78.

(7) Hwang, D. H.; Han, C. Real-time Monitoring for a Process with Multiple Operating Modes. *Control Eng. Pract.* **1999**, *7*, 891.

(8) Chen, J.; Liu, J. Using Mixture Principal Component Analysis Networks to Extract Fuzzy Rules from Data. *Ind. Eng. Chem. Res.* **2000**, *39*, 2355.

(9) Lee, Y.-H.; Kim, M.; Han, C. Robust Recursive Modeling for Adpative Monitoring. *Proceedings of PSE ASIA 2005*, Seoul, Korea. 2005; 259.

(10) Jin, H. D.; Lee, Y.-H.; Han, C. A Novel Model Adaptation Method for Multivariate Statistical Process Control. *Proceedings of 2004 AIChE Annual Meeting*, Austin, TX, 2004; 429h.

(11) Montgomery, D. C. *Introduction to Statistical Quality Control*; Wiley: New York, 1996.

(12) Nomikos, P.; Macgregor, J. F. Multivariate SPC Charts for Monitoring Batch Processes. *Technometrics* **1995**, *37*, 41

(13) Liu, H.; Shah, S.; Jiang, W. On-line outlier detection and data cleaning. *Comput. Chem. Eng.* **2004**, *28*, 1635.

Received for review July 21, 2005
Revised manuscript received October 21, 2005
Accepted November 3, 2005

IE050850T