Environ. Eng. Res. Vol. 11, No. 2, pp. 1～ , 2006
Korean Society of Environmental Engineers

FAULT DETECTION, MONITORING AND DIAGNOSIS OF SEQUENCING BATCH REACTOR FOR INTEGRATED WASTEWATER TREATMENT MANAGEMENT SYSTEM

Chang Kyoo Yoo[1,2,†], Peter A. Vanrolleghem[2], and In-Beum Lee[1]

[1]School of Environmental Science and Engineering/Department of Chemical Engineering, Pohang University of Science and Technology, San 31 Hyoja Dong, Pohang, 790-784, Korea
[2]BIOMATH, Department of Applied Mathematics, Biometrics and Process Control, Ghent University, Coupure Links 653, B 9000 Gent, Belgium
(received March 16, 2006, accepted March 16)

Abstract : Multivariate analysis and batch monitoring on a pilot-scale sequencing batch reactor (SBR) are described for integrated wastewater treatment management system, where a batchwise multiway independent component analysis method (MICA) are used to extract meaningful hidden information from non-Gaussian wastewater treatment data. Three-way batch data of SBR are unfolded batch-wisely, and then a non-Gaussian multivariate monitoring method is used to capture the non-Gaussian characteristics of normal batches in biological wastewater treatment plant. It is successfully applied to an 80L SBR for biological wastewater treatment, which is characterized by a variety of error sources with non-Gaussian characteristics. The batchwise multivariate monitoring results of a pilot-scale SBR for integrated wastewater treatment management system showed more powerful monitoring performance on a WWTP application than the conventional method since it can extract non-Gaussian source signals which are independent and cross-correlation of variables.

Key Words : Advanced monitoring, Batchwise unfolding, Integrated wastewater treatment management system, Multivariate statistical process control (MSPC), Multiway independent component analysis (MICA), Sequencing batch reactor (SBR)

INTRODUCTION

The increase in environmental restrictions in recent times has led to an increase in efforts aimed at attainment of better effluent quality of wastewater treatment plants. Achieving this goal requires advanced monitoring and control of plant performance. Wastewater treatment plants are slow when they have to recover from a 'bad' state to a 'normal' state. The early detection and isolation of faults in the biological process is therefore very effective since they allow corrective action to be taken well before the situation becomes dangerous. Some changes are not very obvious and may gradually grow until they become a serious operational problem.[1-6)]

Monitoring and fault detection of the environmental processes are very important tasks in environmental system engineering (ESE) since they aim to ensure the success of the planned operations and to improve the productivity of the related processes. Early detection of faults can help avoid major breakdowns and incidents. In general, four tasks are involved in the process monitoring: (1) *fault detection*, which gives an indication that something is going wrong in the process; (2) *fault identification* (or *diagnosis*), which determines the root cause of the fault; (3) *fault estimation*, which assesses the size of the fault; and (4) *fault*

† Corresponding author
E-mail: ckyoo@postech.edu, ChangKyoo.Yoo@biomath.ugent.be
Tel: +82-54-279-5966, Fax: +82-54-279-8299

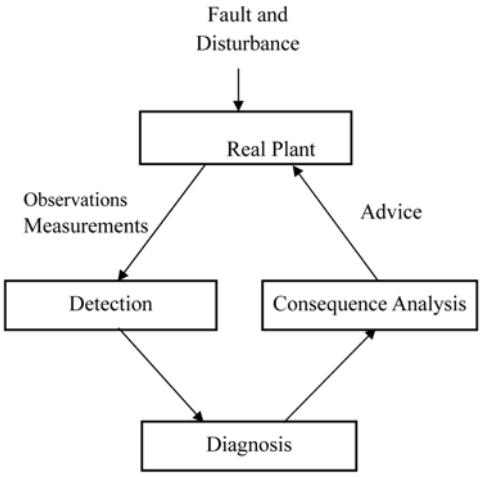

Figure 1. Fault detection and diagnosis scheme for a process monitoring.

reconstruction, which estimates the fault-free values. Figure 1 illustrates a environmental process monitoring scheme. Fault detection is defined as a combination of process observations and measurements, data analysis and interpretation to detect abnormal features or effects and the isolation of faults. Fault diagnosis involves the analysis of effects to identify aberrant variables and rank likely causes. Advice includes a synthesizing strategy to eliminate the causes and return the process to normal operating conditions.[1,5)]

In recent industrial plants, many variables are measured in various environmental system and are recorded in abundance. However, such data sets are highly correlated and are subject to considerable noise. In the absence of an appropriate processing method, only limited information can be extracted, which causes insufficient understanding of the process by the operator and may lead to unstable operation. If properly treated, this data can provide a wealth of information leading to keep the plant operators understand the status of the process and assist them to make appropriate actions to remove abnormalities resulting from the process.

Sequencing batch reactor (SBR) processes have demonstrated their efficiency and flexibility in the treatment of wastewaters with high concentrations of nutrient, nitrogen, phosphorous, and toxic compounds from domestic and industrial sources. A SBR has a unique cyclic batch operation, usually with five well-defined phases: fill, react, settle, draw and idle. Most of the advantages of SBR processes may be attributed to their single-tank designs and the flexibility that allows them to meet many different treatment objectives, and which is derived from the possibility of adjusting the duration of the different phases. But the SBR process is highly nonlinear, time-varying and subject to significant disturbances like hydraulic changes, composition variations and equipment failures. Small changes in concentrations or flows can affect effluent quality and microorganism growth. However, treatment performance, the key indicator of process performance, is often only examined off-line in a laboratory. Even though operators are aware that there are some problems in treatment performance, they cannot quickly find out or predict what the causes are and when the problems will occur because most batch processes are run without any effective form of real-time on-line monitoring. Therefore, multivariate analysis and process monitoring of SBR are crucial to detect faults that can be corrected prior to completion of the batch or can be corrected in subsequent batches because it may take several days, week or ever months for the biological process to recover from abnormal operation.[4,7)]

Multiway principal component analysis (MPCA) has been shown to be powerful monitoring tools in many industrial batch processes.[8-9)] However, for some complicated cases in industrial environmental processes which especially have non-Gaussian characteristics, principal component analysis (PCA) exhibits bad behaviour because of its Gaussianity assumption. They have the shortcoming that the measurement variables of the batch process should have Gaussian correlations. Recently, a new monitoring method to cope with this non-Gaussain and nonlinear biology has developed in monitoring the wastewater treatment process, which can capture the biological relationship using independent component analysis (ICA), are suggested and compared in order to overcome the drawbacks of conventional method and obtain better monitoring performance.[10-12)]

In this work, batchwise multiway independent

component analysis (MICA) suggested by Yoo et al.[13] is used to tackle the non-Gaussian problem and obtain better batch monitoring performance of the pilot-scale SBR. This article is organized as follows. Batchwise MPCA and batchwise MICA are described in material and method section. Case study in a pilot-scale SBR is given in an experimental result section. Finally, the conclusions will be shown.

MATERIALS AND METHODS

MPCA

Multiway principal component analysis (MPCA) is used for the analysis and monitoring of batch process data. Batch data are typically reported in terms of batch numbers, variables and times. Therefore, batch processes are, by nature, leading to a 3-way matrix ($\underline{\mathbf{X}}(I \times J \times K)$) of data, where I is the number of batches, J is the number of variables and K is the number of times each batch is sampled. In a typical batch run, $j = 1, 2, \mathrm{K}, J$ variables are measured at $k = 1, 2, \mathrm{K}, K$ time intervals throughout the batch. There exists similar data on several ($i = 1, 2, \mathrm{K}, I$) similar process batch runs. This matrix can be decomposed using various three-way techniques, one of which is MPCA. MPCA is equivalent to performing ordinary PCA on a large two-dimensional matrix **X** constructed by unfolding the three-way data in the manner shown schematically in Figure 2.[8]

MPCA decomposes the three-way array $\underline{\mathbf{X}}$ into a summation of the product of a score t_r and a loading matrix P_r plus a residual array $\underline{\mathbf{E}}$ that is minimized in the least squares sense as follows:

$$\underline{\mathbf{X}} = \sum_{r=1}^{R} t_r \otimes P_r + \underline{\mathbf{E}} = \sum_{r=1}^{R} t_r p_r^T + \mathbf{E} = \hat{\mathbf{X}} + \mathbf{E} \qquad (1)$$

where $\otimes$ denotes the Kronecker product ($\underline{\mathbf{X}} = \mathbf{t} \otimes \mathbf{P}$ is $\underline{X}(i, j, k) = t(i)P(j, k)$), R denotes the number of principal components retained, t_r expresses the relationship among batches, p_r is related to variables and their time variation, **E** is the residual matrix. The first expression in Eq. (1) gives the 3-D decomposition while the second expression displays the more common 2-D decomposition.

The three-way array $\underline{\mathbf{X}}$ can be unfolded in three ways, which give rise to the following two-dimensional matrices[9] :

- Batches × variables at each time (time-wise unfolding)
- Variables × time for each batch (batch-wise unfolding)
- Batches × times for each variable (variable-wise unfolding)

Time-wise unfolding is useful for analyzing the variability among samples, and batch-wise unfolding facilitates the analysis of the variability among batches by summarizing the information related to the measured variables and their variations over time. Variable-wise unfolding can be used to obtain information about the variability among the batch variables. In previous studies, the batch-wise unfolding method has been the most widely used method for analyzing batch process data. Its aim is model the differences of each batch run from a theoretical normal operating condition. In addition, it is suitable for deriving estimate of final quality measures from process data that are usually not available until the batch terminates. Moreover, the majority of the nonlinear behavior of the process is eliminated by subtracting off the mean trajectory of each variable at each time. After the unfolded matrix has been mean-centered and scaled, PCA is performed. The results from PCA are the loading vectors, and the calculated scores for each batch. The loading vectors contain a weight for each variable at each time.[8-9] In this paper, the batch-wise unfolding scheme in Figure 2 is used.

The statistics used for a MPCA are Hotelling's

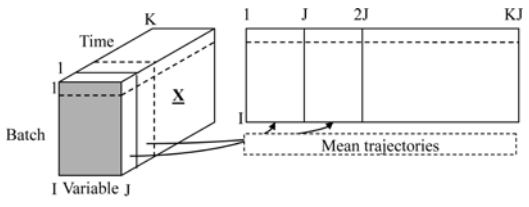

Figure 2. Batchwise unfolding method for a three-way batch.

T^2 and squared prediction error (*SPE*). If a new batch is good and consistent with the normal batches, its scores should fall within the normal range and the *SPE* or *Q*-statistic should be small. The T^2 and *Q*-statistics obtained at end-of-batch for batch *i* are calculated as

$$T_i^2 = \mathbf{t}_{new}{}^T \mathbf{S}^{-1} \mathbf{t}_{new} \sim \frac{R(I^2-1)}{I(I-R)} F_{R,I-R} \quad (2)$$

$$Q_i = \mathbf{e}_i \mathbf{e}_i^T = \sum_{c=1}^{KJ} \mathbf{E}(i,c)^2 \quad (3)$$

where $\mathbf{e}_i$ is the *i*th row of **E**, *I* is the number of batches in the reference set, $\mathbf{t}_r$ is a vector of *R* scores, **S** is the ($R \times R$) covariance matrix of the t-scores calculated during the model development (diagonal due to the orthogonality of the t score values), *R* is the number of principal components retained in the model, $F_{R,I-R}$ is the *F*-distribution value with *R* and *I-R-1* degrees of freedom. Statistical limits on the T^2 and *Q*-statistics are computed by assuming that the data have a multivariate normal distribution.[8)]

Independent Component Analysis for non-Gaussianity data

The extraction of hidden information from the multidimensional data set is crucial to successful fault detection. In this paper, we adopt the technique of independent component analysis (ICA) to extract such hidden information. It is well known that many of the variables monitored in process systems are not independent. The measured process variables may be combinations of independent variables that are not directly measurable (referred to as latent variables in probabilistic theory). Independent component analysis (ICA) can extract these underlying factors or components from multivariate statistical data. It defines a generative model for the observed multivariate data, which are typically in the form of a large database of samples. In this model, the data variables are assumed to be linear or nonlinear mixtures of some unknown latent variables, where the system governing the mixing of the latent variables is also unknown. The latent variables, which are called the independent components (ICs) of the observed data, are assumed to be non-Gaussian and mutually independent. ICA seeks to extract these ICs as well as the mixing process.[10-12)]

What distinguishes ICA from PCA is that it looks for components that are both statistically independent and non-Gaussian. PCA is a dimensionality reduction technique in terms of capturing the variance of the data which is capable of extracting uncorrelated latent variables from correlated data, while ICA is designed to separate the independent components (ICs) that are independent and constitute the observed variables. Furthermore, PCA can only impose independence up to second order statistics information (mean and variance) while constraining the direction vectors to be orthogonal, whereas ICA has no orthogonality constraint and also involves higher-order statistics. In case of ICA, the most interesting directions are those that show the non-Gaussian distributions, whereas the directions of PCA are looking for the Gaussian distribution. Thus, both models have a different projection pursuit. ICA may reveal more meaningful information in the non-Gaussian data than PCA.[12)]

To illustrate the superiority of ICA over PCA, we applied the two types of analysis to a simple example system. Let's consider two source variables that have the uniform distributions shown in Figure 3(a). The source variables are linearly independent, i.e., the values of one source variable do not convey any information about the other source variable. These sources are linearly mixed as follows:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 & 3 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \quad (4)$$

Figure 3(b) shows the scatter-plot of the mixtures. Note that the random variables x_1 and x_2 are not independent because it is possible to predict the value of one of them from the value of the other. When PCA is applied to these mixed variables, it gives two principal components. The axes of the first and second PCs (PC1, PC2) are shown in Figure 3(b). The first PC is the axis

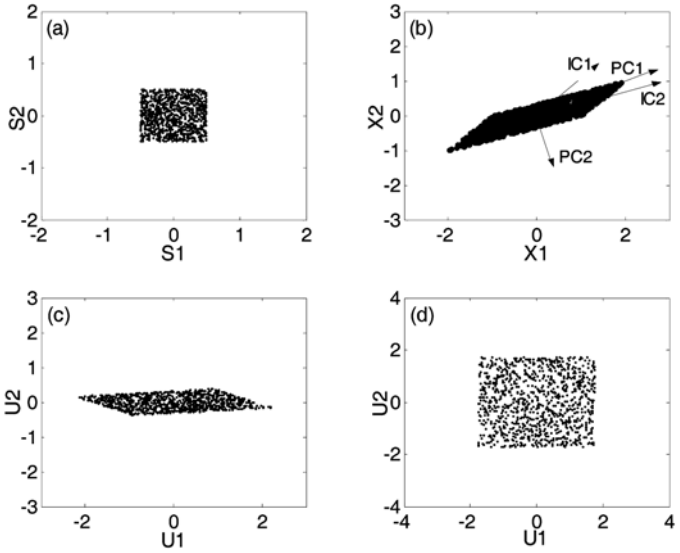

Figure 3. (a) Scatter plot of the original source data, (b) The mixtures and axes of PCA and ICA, (c) The recovered source data using PCA, (d) The recovered source data using ICA.

capturing the highest variance in the data and the second PC is the axis orthogonal to the first PC. Figure 3(c) shows the PCA solution, which differs from the original because the two principal axes are still dependent. However, the ICA solution shown in Figure 3(d) can recover the original sources since ICA not only decorrelates the data but also rotates it such that the axes of u_1 and u_2 are parallel to the axes of s_1 and s_2. The axes of the first and second independent components (IC1, IC2) are shown in Figure 3(b). The basic idea of this approach is to extract essential independent components that drive a process and to combine them with process monitoring techniques. The simple example given above clearly demonstrates that if the latent variables follow a non-Gaussian distribution, the ICA solution extracts the original source signal to a much greater extent than the PCA solution. Therefore, it is natural to infer that a monitoring system based on the ICA solution may give better results compared to PCA (for detail algorithm of independent component analysis, see the appendix A).

Batchwise Multiway Independent Component Analysis

The monitoring method based on batchwise multiway independent component analysis (MICA) suggested by Yoo et al.[13] is similar to that based on MPCA. The key idea is to exploit the ability of MICA to extract features from three-way batch data by projecting the data onto a low-dimensional space that summarizes both the variables and their time trajectories. MICA is equivalent to performing ICA on a large two-dimensional matrix **X** constructed by batchwise unfolding the three-way data matrix $\underline{\mathbf{X}}$. MICA decomposes the three-way array $\underline{\mathbf{X}}$ into a summation of the product of independent vectors s_r and loading matrices $\mathbf{A_r}$ plus a residual array $\underline{\mathbf{E}}$ so that the ICs s become as independent of each other as possible:

$$\underline{\mathbf{X}} = \sum_{r=1}^{R} \mathbf{s}_r \otimes \mathbf{A}_r + \underline{\mathbf{E}} = \sum_{r=1}^{R} s_r \mathbf{a}_r^T + \mathbf{E} = \hat{\mathbf{X}} + \mathbf{E} \tag{5}$$

where $\otimes$ denotes the Kronecker product ($\underline{\mathbf{X}} = \mathbf{s} \otimes \mathbf{A}$ is $\underline{X}(i,j,k) = s(i)A(j,k)$) and R denotes the number of ICs retained. The **S** and **A** matrices in Eq. (5) can be equivalent to the loading matrix and score matrices by analogy with MPCA, *i.e.* **S** can be regarded as the score matrix **T**, and **A** can be treated as the loading matrix **P**. The *i*th elements of the independent vector **s** correspond to the *i*th batch and summarize the overall variations

in this batch with respect to the other batches over the entire history of the batch. The mixing matrix, **A**, summarizes the time variations of the measured variables about their average trajectories. The elements of this matrix are the weights, which give the independent vectors s for a batch when applied to each variable at each time interval within that batch.[13] In MICA, an unfolded data matrix, **X**, representing multiple batch runs is decomposed, $\mathbf{X} = \mathbf{U}_r \mathbf{\Lambda}_r^{1/2} \mathbf{Z}_r$, and (r x r) orthogonal rotation matrix, **B**, is computed to produce rotated scores, $\mathbf{A} = \mathbf{U}_r \mathbf{\Lambda}_r^{1/2} \mathbf{B}$ and loadings, $\mathbf{S} = (\mathbf{B}^T \mathbf{Z}_r)$ giving $\mathbf{X} = \mathbf{U}_r \mathbf{\Lambda}_r^{1/2} \mathbf{B}\mathbf{B}^T \mathbf{Z}_r$, such that $\mathbf{B}\mathbf{B}^T = \mathbf{I}$ (for detail procedure of developing the MICA model, see the appendix B).

In MICA,[13] two statistics are deduced from the process model in normal operation: the *D*-statistic for the systematic part of the process variation and the *Q*-statistic for the residual part of the process variation. The *D*-statistic for a batch *k*, also known as the I^2 statistic, is the sum of the squared independent scores and is defined as follows:

$$I^2(k) = \hat{\mathbf{s}}_{newd}(k)^T \hat{\mathbf{s}}_{newd}(k) \tag{6}$$

The *Q*-statistic for a batch *k*, also known as the *SPE* statistic, is defined as follows:

$$SPE(k) = \mathbf{e}(k)^T \mathbf{e}(k) = (\mathbf{x}(k) - \hat{\mathbf{x}}(k))^T (\mathbf{x}(k) - \hat{\mathbf{x}}(k)) \tag{7}$$

where $\hat{\mathbf{x}}$ can be calculated as follows:

$$\hat{\mathbf{x}} = \mathbf{Q}^{-1} \mathbf{B}_d \hat{\mathbf{s}} = \mathbf{Q}^{-1} \mathbf{B}_d \mathbf{W}_d \mathbf{x} \tag{8}$$

In MPCA monitoring, the confidence limit is based on a specified distribution based upon the assumption that the latent variables follow a Gaussian distribution. In batchwise MICA monitoring, however, the independent components do not conform to a specific distribution; hence, the confidence limits of the I^2 and *SPE* statistics cannot be determined directly from a particular approximate distribution. An alternative approach to defining the nominal operating regions is to use data-driven techniques such as non-parametric empirical density estimates using kernel density estimation (KDE).[14] In this paper, the confidence limits of the two statistics, I^2 and *SPE* were obtained by kernel density estimation, where the Gaussian kernel and the least squares cross-validation (LSCV) methods for selecting *h* were used. Here, the I^2 value is used to detect faults associated with abnormal variations within an MICA model subspace, whereas the *SPE* value is used to detect new events that are not taken into account in an MICA model subspace.

The I^2 and *Q*-statistic methods provide reliable tools for detecting that a multivariable process has gone out-of-control. The principal component loadings and detection limits for multivariate statistical process control (MSPC) are computed from data representative of normal operating conditions. The process data obtained under normal operating conditions contain only common cause variation, i.e., variation in the process that is not due to a fault or disturbance. Confidence limits can be developed around the common cause variation for both the systematic component of the variation and the residual component. The systematic component of the process data, which is described by the process model, is monitored using the I^2-statistic chart. The pattern of the residuals is monitored using the *Q*-statistic, which is a summation of the squared residuals of a specific time. The I^2-statistic monitors systematic variations in the independent variable space, while the *Q*-statistic represents variations not explained by the retained ICs. That is, faults in the process that violate the normal correlation of variables are detected in the independent component (IC) subspace by the I^2-statistic, whereas faults that violate the batchwise MICA models are detected in the residual space by the *Q*-statistic.

Process description of the pilot-scale SBR system

The batch monitoring method is applied to a pilot-scale SBR system shown in Figure 4. A fill-and-draw sequencing batch reactor (SBR) with an 80 *l* working volume is operated in a 6 h cycle mode and each cycle consists of fill/anaerobic (1

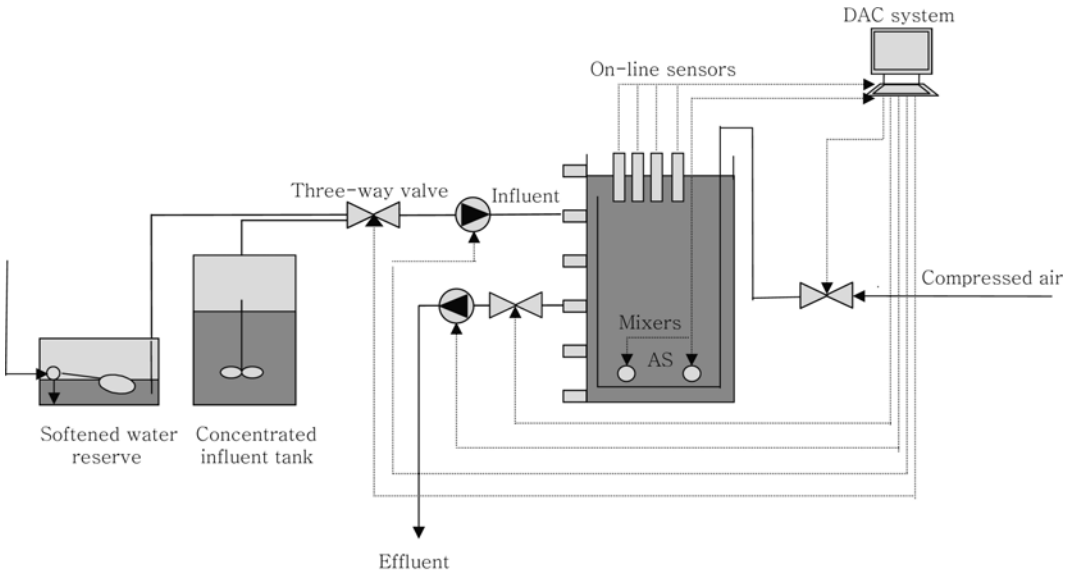

Figure 4. Schematic diagram of the pilot-scale sequencing batch reactor.

h), aerobic (2 h 30 min), anoxic (1 h), re-aerobic (30 min) and settling/draw (1 h) phases. The hydraulic retention time (HRT) and the solid retention time (SRT) are maintained at 12 h and 10 d, respectively. Like synthetic municipal-like sewage, loading amounts of COD, NH_4^+-N and PO_4^{3-}-P per cycle in standard conditions are 440, 60 and 9.5 mg/*l*, respectively. The control of the duration/sequence of phases and the on/off status of peristaltic pumps, mixer and air supply are automatically achieved by a Labview program. Six electrodes for pH, oxidation-reduction potential (ORP), dissolved oxygen (DO), temperature, conductivity and weight are connected to the individual sensors to check the status of the SBR, where a set of on-line measurements is obtained every one minute. Thus, no advanced nutrient or expensive measurement devices were installed in order to run an on-line monitoring algorithm of the SBR process.

It has been reported that on-line sensor values collected in SBR are related with dynamic characteristics of the nutrient concentrations (COD, NH_4^+-N, PO_4^{3-} and NO_3^-) in SBRs.[15,16] The on-line values of pH, ORP and DO profiles can detect the ends of phosphate release, ammonia conversion, and phosphate uptake, which also

Table 1. Real-time measured variables of a SBR

No.	Variables
1	Conductivity (*mS*)
2	Dissolved oxygen concentration (*mg*/*l*)
3	Oxidation reduction potential (*m*V)
4	pH
5	Temperature (°C)
6	Weight (*g*)

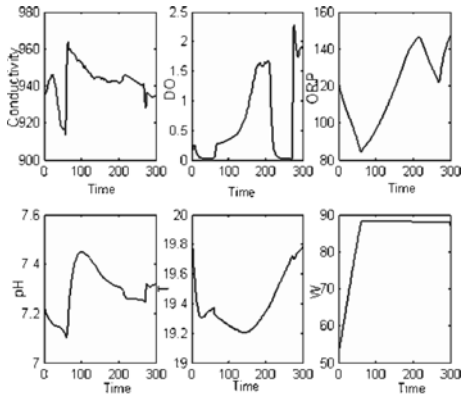

Figure 5. Typical batch trajectory profiles of a SBR.

are useful information sources. Therefore, six on-line variables of Table 1 including pH, ORP, DO signals were calculated from on-line sensor profiles and included into the database. Figure 5

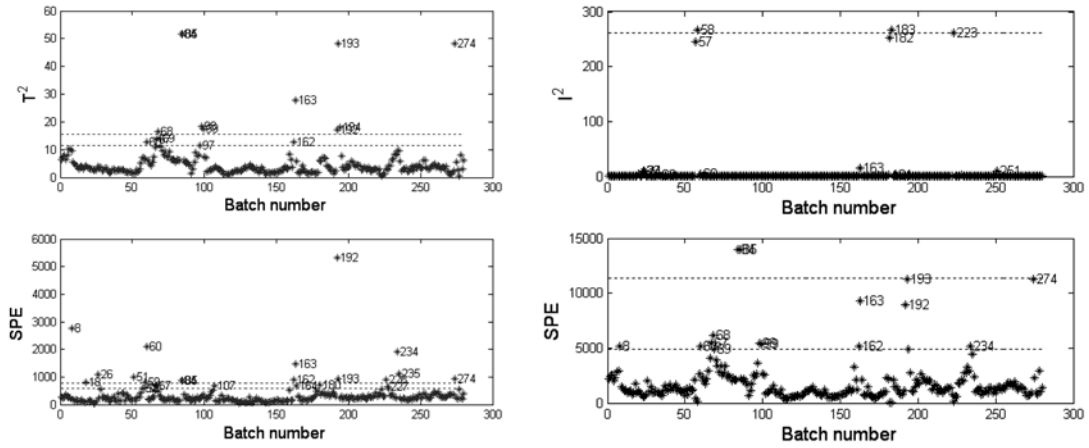

Figure 6. Multivariate analysis of all 280 batches using (a) MPCA, (b) MICA.

shows typical batch profiles of the 6 variables during a batch. We considered 280 batches in the historical data set of the SBR for which 6 variables were available at 300 time instants.

EXPERIMENTAL RESULTS AND DISCUSSION

Multivariate analysis of historical data set in SBR

The MPCA and MICA models for the SBR monitoring are used to analyze the historical SBR data set. Both methods are used to all 280 batches to interpret the historical batch and get rid of the abnormal batches in SBR. Figure 6 shows the monitoring result of 280 batches of the SBR using the MPCA and MICA methods, where the dotted lines correspond to the 95 and 99% confidence limits. Five components of the MPCA model were selected by the cross-validation method.[13] To ensure comparison of equivalent models, five ICs were selected for the MICA model. From this figure, we notice that the MICA plot shows characteristics dissimilar from the MPCA one. Compared to MPCA, MICA points to a lower number of abnormal batches in SBR. This difference can be explained by the density estimation of the SBR data. Figure 7 shows that the density estimate of the first score (t_1) in MPCA does not follow the Gaussian distribution but the

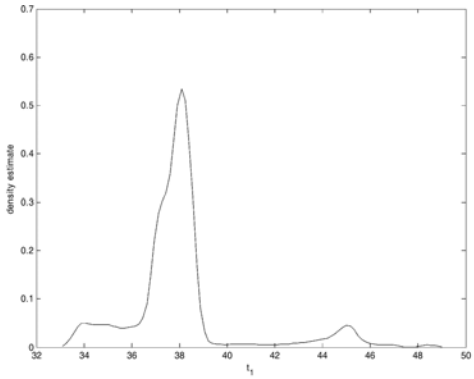

Figure 7. The density estimate of non-Gaussian distribution of the first principal score (t_1) obtained from MPCA.

'*supergaussian distribution*' in which process variables take relatively more often values that are very close zero, where the probability density of the data is peaked in the middle and has heavy tails (large values far from zero).[4] Thus, the T^2 and *SPE* charts of MPCA that are based on the assumption that the data are Gaussian distributed may cause a false result when it is used for SBR monitoring. This observation is the motivation of the MICA method because MICA is sensitive to modes whose influences on the measured variables follow a supergaussian distribution.

Figure 8 represents the loading plot of each variable of each time interval of the first IC. It shows the types of information that can be extracted when MICA is used in batch modeling.

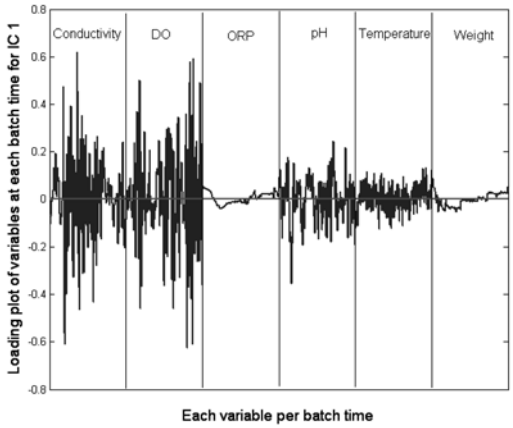

Figure 8. The variable loading plot of the first independent score (i_1) obtained from MICA.

The loading plot obtained from MICA gives the history and identified important features of the SBR. From this figure, we notice that the DO, conductivity, and pH show large variations and have large influences during a batch, whereas ORP and weight show relatively small variations. This biological interpretation of MICA is meaningful to the multivariate monitoring of batch processes in case of non-Gaussian distributions.

Batch monitoring of SBR

The MPCA and MICA models for the SBR monitoring were developed after an analysis of the historical SBR data set. The MPCA model selected 143 batches to create a rather broad scope of normal batches, where 7 abnormal batches (batch number: 8,18,26,51,60,84,85) were excluded for the normal operating condition (NOC) model. The MICA model selected 146 batches, where 4 abnormal batches (batch number: 57, 58, 84, 85) were excluded for the normal NOC model. The test data set that consisted of the following 30 batches was projected onto the reduced MPCA and MICA model spaces. Figure 9 shows the batch monitoring result of 30 test batches by MPCA and MICA. Both of them could detect two abnormal batches (batch 12, 13). Both the MPCA and MICA methods show similar detection times for two non-conforming batches. Figure 10 shows an univariate plot of the on-line DO measurements of the normal batches and the abnormal batch no. 12 which was over-aerated at the start of the aeration phase. Thus, when a fault is detected in biological batch process, the batch run will control the current batch and moreover, operators will use the information on the fault to correct the following batch.

There is a single batch (batch no. 9) which shows the different monitoring result from two methods. While MPCA detected batch 9 as an abnormal batch, MICA left batch 9 as a normal batch. Figure 11 shows on-line monitoring charts of MPCA and MICA for *normal* batch 9. The MICA result shows that the I^2 and *SPE* statistics for this batch are within the control limits for the whole duration of the batch run. Therefore, this batch in MICA is assigned as being “in

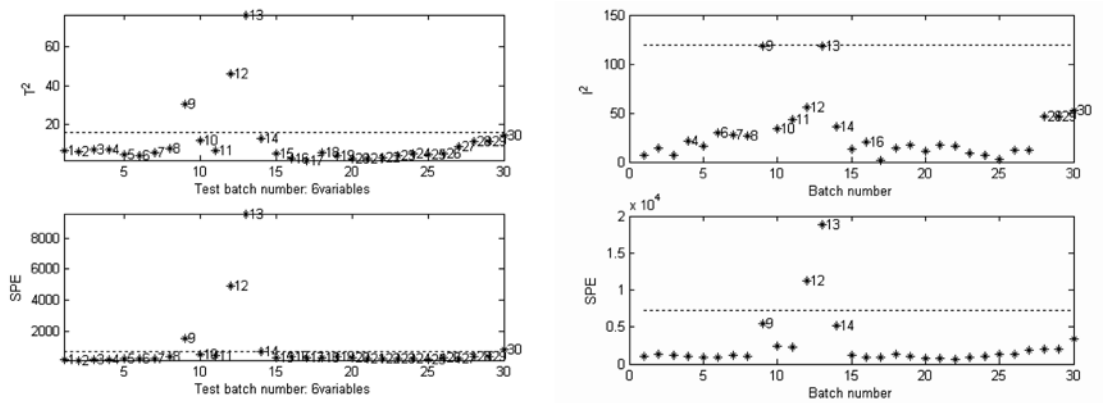

Figure 9. Monitoring result of 30 test batches. (a) MPCA and (b) MICA. The dotted lines correspond to the 99% confidence limit.

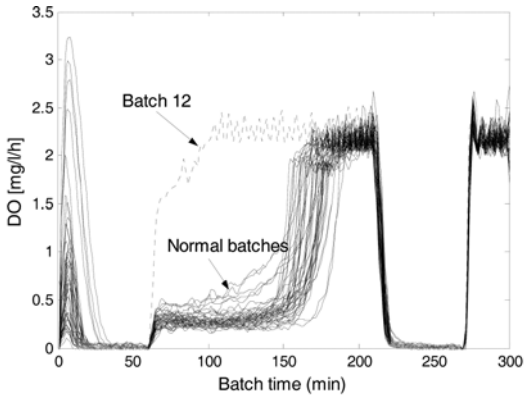

Figure 10. Univariate plot of on-line DO measurement of the normal batches and the abnormal batch 12.

control" or "normal". Actually, batch 9 is a normal batch. But the *SPE* chart of MPCA exceed the upper control limits once, at around the 150th sampling time. When MPCA is applied to non-Gaussian data, the T^2 chart of MPCA may suffer oversensitivity for normal batches, e.g., batch 9. As a data set deviates from a Gaussian distribution, the variance tends to increase and hence the T^2 statistic tends to decrease. Hence, it results in less false alarms. Typically, this increases the false alarm rate of the MPCA in which a normal batch might be judged as a non-conforming one. Obviously, this deteriorates the reliability of the monitoring system. This false alarm of MPCA comes from a bad modeling result. The above clearly showed that the MICA monitoring technique can solve the non-Gaussian dynamics of the SBR better than MPCA. On the other hand, the I^2 chart of MICA moves up and down four times from the beginning of a batch to the end time of SBR process since the operating conditions of SBR process have a unique cyclic batch operation (anaerobic, 1st aerobic, anoxic and 2nd aerobic phases). It shows that the MICA model can capture the biological phenomena and their relationships which occur during the batch.

In summary, the capability difference between the MPCA and MICA methods mainly originates from the extracted feature components. Both methods find hidden information from the multidimensional data set. While MPCA looks for Gaussian components, MICA searches for non-Gaussian components. If a data set contains any non-Gaussian component, MICA can show better feature extraction performance than MPCA. Therefore, MICA may improve the monitoring performance by extracting the key hidden variables that influence the process.[4] In the SBR operation, the influent wastewater is fed into the reactor and mixed with already existing microorganisms. Therefore, the performance of the current batch highly depends on microorganism activity in the previous batches. In addition, the SBR process is subject to significant disturbances like hydraulic changes and composition variations. Small changes in concentrations or flows can have a large effect on the kinetics of biological rations leading to batch-to-batch variability in effluent quality and microorganism growth. Compared to the previous ones, the MICA method provides more meaningful in-

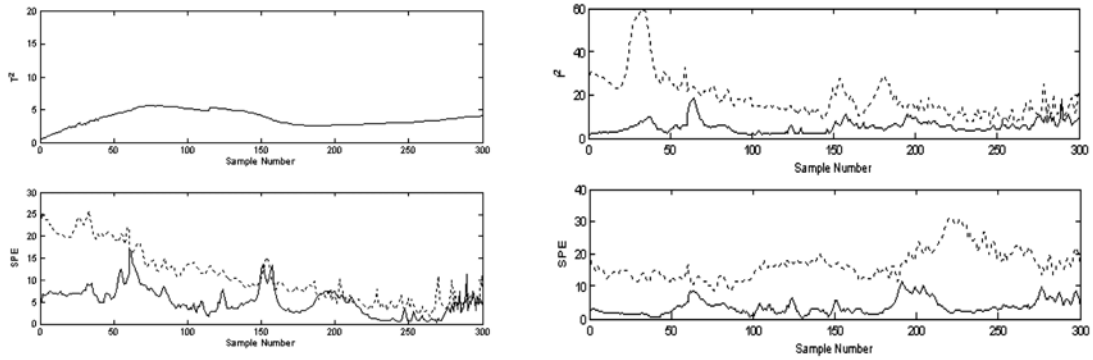

Figure 11. On-line monitoring charts for *normal* batch 9: (a) T^2 and *SPE* charts of MPCA, (b) I^2 and *SPE* charts of MICA.

formation on the evolving biological process and captures the biological relation among batches, which results in more robust monitoring performance.

CONCLUSIONS

The SBR poses an interesting challenge for process monitoring of systems characterized by non-stationary, batchwise, multiscale, and non-Gaussian characteristics. This paper presents the application of a batchwise monitoring of MICA to a pilot-scale SBR. Three-way batch data from SBR is unfolded batch-wisely and then MICA is used to capture the non-Gaussian relation among batches. The results of this pilot-scale SBR monitoring system using simple on-line measurements clearly demonstrated that the batchwise MICA monitoring technique showed lower false alarm rate and physically meaningful, that is, robust monitoring results, since it can to extract meaningful hidden information from non-Gaussian data of a SBR. For integrated wastewater treatment management system, we are developing on-going research,[16] that is, the integrated framework of data-driven statistical model, activated sludge models, and molecular microbiology information for sustainable wastewater treatment operation. It is able to detect the faults, to integrate the monitoring and control systems and to develop model-based optimization for sustainable operation. The proposed framework will be easily applied to a modular internet-based remote supervision and control system of other wastewater treatment plants.

ACKNOWLEDGEMENTS

This work was supported by Brain Korea 21.

REFERENCES

1. Olsson, G. and Newell, B., Wastewater Treatment Systems: Modelling, diagnosis and Control, IWA, UK (1999).
2. Choi, K. S., Kim, B. G., Han, G. B., and Kim, C. W., "Detection of toxicity of wastewater in the activated sludge processes," *Environ. Eng. Res.*, **4**(1), 59-70 (1999).
3. Kim, I. S., Young, J. C., Kim, S. Y., and Kim, S. M., "Development of monitoring methodology to fingerprint the activated sludge processes using oxygen uptake rate," *Environ. Eng. Res.*, **6**(4), 251-259 (2001).
4. Yoo, C.K., Lee, D. S. and Vanrolleghem, P.A., "Application of multiway ICA for on-line monitoring of a sequencing batch reactor," *Wat. Res.*, **38**(7), 1715-1732 (2004).
5. Yoo, C.K., Son, H. R., and Lee, I. B., "Modeling and multiresolution analysis in a full-scale industrial plant," *Environ Eng. Res.*, **10**(2), pp. 88-103 (2005).
6. He, Q. P., Qin, S. J., and Wang, J. A., new fault diagnosis method using fault directios in fisher discriminatn analysis, *AIChE J.*, **51**(2). 555-571 (2005).
7. Lee, D. S., and Vanrolleghem, P. A., "Monitoring of a sequencing batch reactor using adaptive multiblock principal component analysis," *Biotech. & Bioeng*, **82**, 489-497 (2003).
8. Nomikos, P. and MacGregor, J. F., "Monitoring batch processes using multiway principal component analysis," *AIChE J.* **40**(8), 1361-1375 (1994).
9. Nomikos, P. and MacGregor, J. F., Multivariate SPC charts for monitoring batch processes, *Technometrics*, 37 (1995) 41 59.
10. Hyvärinen, A., "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Trans. on Neural Networks.*, **10**, 626-634 (1999).
11. Hyvärinen, A., "Survey on independent component analysis," *Neural Computing Surveys*, **2**, 94-128 (1999).
12. Hyvärinen, A., Karhunen, J., and Oja, E., "Independent component analysis," John Wiley & Sons, INC., USA, (2001).
13. Yoo, C. K., Lee, J., Vanrolleghem, P. A., and Lee, I. B., "On-line monitoring of batch processes using multiway independent component analysis," *Chemom. and Intel. Lab. Sys.* **71**(2), 151-163 (2004).
14. Martin, E. B. and Morris, A. J., Non-parametric confidence bounds for process performance

monitoring charts, *J. Proc. Cont.*, 6 (6) 349 358 (1996).
15. Chang, C. H. and Hao, O. J., "Sequencing batch reactor system for nutrient removal: ORP and pH profiles," *J. Chem. Tech. Biotechnol.* **67**, 27-38 (1996).
16. Yoo, C. K., Vanrolleghem, P. A., and Lee, I. B., "Integrated framework of model-based monitoring, control and optimization for a sustainable biological wastewater treatment operation," Proceedings of KOSENV Fall Meeting (2004).

Appendix A. Detailed algorithm of independent component analysis

In this appendix A, the detailed algorithm of independent component analysis is explained. In the ICA algorithm, it is assumed that d measured variables $x_1, x_2, \cdots, x_d$ can be expressed as linear combinations of $m(\leq d)$ unknown independent components $s_1, s_2, \mathrm{K}, s_m$. The relationship between them is given by

$$\mathbf{X} = \mathbf{AS} + \mathbf{E} \tag{A1}$$

where $\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \mathrm{K}, \mathbf{x}(n)] \in R^{d \times n}$ is the data matrix (in contrast to PCA, ICA employs the transposed data matrix.), $\mathbf{A}=[a_1, \cdots, a_m] \in R^{d \times m}$ is the unknown mixing matrix, $\mathbf{S} = [\mathbf{s}(1), \mathbf{s}(2), \mathrm{K}, \mathbf{s}(n)] \in R^{m \times n}$ is the independent component matrix, $\mathbf{E} \in R^{d \times n}$ is the residual matrix, and n is the number of samples. Here, we assume $d \geq m$ (when $d=m$, the residual matrix, E, becomes the zero matrix). The basic problem of ICA is to estimate both the mixing matrix A and the independent components S from only the observed data X. Alternatively, one could define the objective of ICA as follows: to find a demixing matrix W whose form is such that the rows of the reconstructed matrix $\hat{\mathbf{S}}$, given as

$$\hat{\mathbf{S}} = \mathbf{WX} \tag{A2}$$

become as independent of each other as possible. Using the ICA algorithm, we can obtain the rows of $\hat{\mathbf{S}}$ whose norm is 1.[12)]

Here, we assume that d equals m except where specified otherwise. The initial step in ICA is whitening (also known as sphering) which eliminates all the cross-correlations among random variables. Consider a d-dimensional random vector $\mathbf{x}(k)$ at sample k with covariance $\mathbf{R}_\mathbf{x} = E(\mathbf{x}(k)\mathbf{x}^T(k))$. The eigen-decomposition of $\mathbf{R}_\mathbf{x}$ is given by

$$\mathbf{R}_\mathbf{x} = \mathbf{U}\Lambda\mathbf{U}^T. \tag{A3}$$

The whitening transformation is expressed as

$$\mathbf{z}(k) = \mathbf{Q}\mathbf{x}(k) \tag{A4}$$

where $\mathbf{Q} = \Lambda^{-\frac{1}{2}}\mathbf{U}^T$. One can easily verify that $\mathbf{R}_z = E(\mathbf{z}(k)\mathbf{z}^T(k))$ is the identity matrix under this transformation. After the whitening transformation we have

$$\mathbf{z}(k) = \mathbf{Q}\mathbf{x}(k) = \mathbf{QAs}(k) = \mathbf{Bs}(k) \tag{A5}$$

where $\mathbf{B}$ is an orthogonal matrix, as verified by the following relation:

$$E\{\mathbf{z}(k)\mathbf{z}^T(k)\} = \mathbf{B}E\{\mathbf{s}(k)\mathbf{s}^T(k)\}\mathbf{B}^T = \mathbf{BB}^T = \mathbf{I}. \tag{A6}$$

We have therefore reduced the problem of finding an arbitrary full-rank matrix $\mathbf{A}$ to the simpler problem of finding an orthogonal matrix $\mathbf{B}$, which then gives

$$\mathbf{s}(k) = \mathbf{B}^T\mathbf{z}(k) = \mathbf{B}^T\mathbf{Q}\mathbf{x}(k). \tag{A7}$$

From Eqs. (A2) and (A7), the relation between $\mathbf{W}$ and $\mathbf{B}$ can be expressed as

$$\mathbf{W} = \mathbf{B}^T\mathbf{Q}. \tag{A8}$$

To calculate $\mathbf{B}$, this matrix is initialized and then updated so that $\mathbf{s}(k)=\mathbf{B}^T\mathbf{z}(k)$ has great non-Gaussianity. There are two common measures of non-Gaussianity: kurtosis and negentropy. Kurtosis is sensitive to outliers, whereas negentropy is based on the information-theoretic quantity of (differential) entropy. Previously, a fast and robust

fixed-point algorithm has been proposed for ICA that entails maximizing the negentropy under the constraint of $\|\mathbf{b}_i\| = 1$, where $\mathbf{b}_i$ is the i^{th} column of **B**. A detailed description of the FastICA algorithm based on maximizing the non-Gaussianity to calculate **B** is given by Hyärinen[11]. After finding **B**, the demixing matrix **W** is obtained using Eq. (A8).

Appendix B. Multivariate analysis and monitoring of the MICA model

In this appendix B, the detailed procedure of the multivariate analysis and monitoring scheme of the MICA model are followed to supervise the progress of a batch process.

A. Develop the batchwise MICA model

1. Acquire an operating data set during normal batch operation.
2. Unfold $\underline{\mathbf{X}}(I \times J \times K)$ to $\mathbf{X}(I \times JK)$ using batch-wise unfolding scheme.
3. Normalize the data $\mathbf{X}(I \times JK)$ using the mean and standard deviation of each variable at each time in the batch cycle over all batches.
4. Transpose the scaled $\mathbf{X}(I \times JK)$. The transposed matrix is designated $\mathbf{X}_{normal}(JK \times I)$.
5. Apply the whitening procedure to acquire the uncorrelated whitened matrix

$$\mathbf{Z}_{normal} = \mathbf{Q}\mathbf{X}_{normal} \tag{A1}$$

where $\mathbf{Z}_{normal} = [\mathbf{z}(1), \mathbf{z}(2), \mathrm{K}, \mathbf{z}(I)] \in R^{r \times I}$, Here, we can extract r columns of **U** in Eq. (7), r is the rank of the covariance matrix of $\mathbf{X}_{normal}$.
6. Carry out the ICA procedure to obtain the following matrices **W**, **B**, and $\mathbf{S}_{normal}$ so that $\mathbf{s}(n) = \mathbf{B}^T\mathbf{z}(n)$ has great non-Gaussianity.

$$\mathbf{S}_{normal} = \mathbf{W}\mathbf{X}_{normal} = \mathbf{B}^T\mathbf{Z}_{normal} \tag{A2}$$

where $\mathbf{A} = (\mathbf{Q}^T\mathbf{Q})^{-1}\mathbf{Q}^T\mathbf{B}$ and $\mathbf{W} = \mathbf{B}^T\mathbf{Q}$.
7. Apply the ordering and dimension reduction of ICA. Thus, the dimension of the batchwise unfolded data matrix is reduced by selecting a few rows of **W** based upon the assumption that the rows with the largest sum of squares coefficient have the greatest effect on the variation of **S**. The m rows of **W** which are separated into deterministic part of **W** ($\mathbf{W}_d$), and the excluded part of **W**, ($\mathbf{W}_e$). The resulting matrices have the following forms:

$$\mathbf{S}_{normal} = \begin{bmatrix} \mathbf{S}_d \\ \mathbf{S}_e \end{bmatrix} \tag{A3}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_d \\ \mathbf{A}_e \end{bmatrix} \tag{A4}$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_d \\ \mathbf{W}_e \end{bmatrix} \tag{A5}$$

8. Finally, the MICA model is constructed. For each batch of $\mathbf{X}(I \times JK)$, $\mathbf{x}'(1 \times JK)$ is projected into the reduced space of the MICA model. For all I batches, the ICs and residuals are calculated from $\mathbf{S}_d$ and *SPE*.
9. Calculate the I^2 statistic:

$$I^2(i) = \mathbf{s}_d(i)^T\mathbf{s}_d(i) \tag{A6}$$

where $1 \le i \le \mathrm{I}$.
10. Calcualte the *Q*-statistic for a batch *i*, also known as the *SPE* statistic, is defined as follows:

$$SPE(i) = \mathbf{e}(i)^T\mathbf{e}(i) = (\mathbf{x}(i) - \hat{\mathbf{x}}(i))^T(\mathbf{x}(i) - \hat{\mathbf{x}}(i)) \tag{A7}$$

10. Obtain the control limits of the I^2 and *SPE* statistics for off-line batch analysis using kernel density estimation.

B. Batch monitoring by the batchwise MICA model

1. For a new batch data, $\mathbf{X}_{test}(K \times J)$, batchwise unfold it to $\mathbf{X}_{test}(KJ \times 1)$. Apply the same scaling used in the modeling.
2. For the scaled and filled matrix, $\mathbf{X}_{test}(JK \times 1)$, calculate the ICs of $\mathbf{s}_{testd}$

$$\mathbf{s}_{testd}(i) = \mathbf{W}_d\mathbf{X}_{test} \tag{A8}$$

3. Calculate the independent components and *SPE* of a test batch $I_{testd}^{\ 2}(i)$ and $SPE(i)$

$$I_{testd}^{\ 2}(i) = \mathbf{s}_{testd}(i)^T \mathbf{s}_{testd}(i) \quad \text{(A9)}$$

$$SPE(i) = \sum_{j=1}^{KJ} (\mathbf{X}_{test} - \hat{\mathbf{X}}_{test\ predj})^2 \quad \text{(A10)}$$

where $\hat{\mathbf{x}}_{pred} = \mathbf{A}_d \mathbf{W}_d \hat{\mathbf{X}}_{pred}$ and $\hat{\mathbf{x}}_{predj}$ is the j^{th} element of $\hat{\mathbf{X}}_{predj}$.

4. Compare the $I_{testd}^{\ 2}(i)$ and $SPE(i)$ statistics of a test batch with control limits of the MICA model.