Correlation-Based Pattern Recognition and Its Application to Adaptive Soft-Sensor Design

Koichi Fujiwara * **Manabu Kano** * **Shinji Hasebe** *

* *Dept. of Chemical Engineering, Kyoto University, Katsura Campus, Nishikyo-ku, Kyoto 615-8510, Japan*
(e-mail: manabu@cheme.kyoto-u.ac.jp)

Abstract: Although soft-sensors have been widely used for estimating product quality or other key variables, they do not always function well in practice due to changes in process characteristics. The Correlation-based Just-In-Time (CoJIT) modeling has been proposed to cope with changes in process characteristics. In the CoJIT modeling, the samples used for local modeling are selected on the basis of correlation together with distance, since changes in process characteristics are expressed as the difference of the correlation. In addition, the individuality of production devices should be considered when they are operated in parallel. However, the CoJIT modeling cannot cope with the individuality of production devices because it is only applicable to time-series data. In the present work, a new pattern recognition method, referred to as the Nearest Correlation (NC) method is proposed, and it selects samples whose correlations are similar to the query. In addition, the proposed NC method is integrated with the CoJIT modeling. The advantages of the proposed CoJIT modeling with the NC method are demonstrated through a case study of a parallelized CSTR process.

Keywords: Soft-sensor, Estimation, Prediction, Just-In-Time modeling, Pattern recognition, Principal component analysis

1. INTRODUCTION

A soft-sensor, or a virtual sensor, is a key technology for estimating product quality or other important variables when on-line analyzers are not available. Partial least squares (PLS) regression and artificial neural network (ANN) have been widely accepted as useful techniques for soft-sensor design (Kano and Nakagawa (2008), Mejdell and Skogestad (1991), Kresta et al. (1994), Kano et al. (2000), Kamohara et al. (2004) and Radhakrishnan and Mohamed (2000)). In addition, the application of subspace identification (SSID) to soft-sensor design has been reported in Amirthalingam and Lee (1999) and Kano et al. (2008) for achieving higher estimation performance.

Generally, building a high performance soft-sensor is very laborious, since input variables and samples for model construction have to be selected carefully and parameters have to be tuned appropriately. In addition, even if a good soft-sensor is developed successfully, its estimation performance deteriorates as process characteristics change. In chemical processes, for example, process characteristics are changed by catalyst deactivation or fouling. Such a situation may deteriorate product quality. Therefore, maintenance of soft-sensors is very important in practice to keep their estimation performance. Ogawa and Kano (2008) indicate that soft-sensors should be updated as the process characteristics change, and also manual, repeated construction of them should be avoided due to its heavy workload.

To update statistical models automatically when process characteristics change, recursive methods such as recursive PLS (Qin (1998)) were developed. These methods can adapt models to new operating conditions recursively. However, when a process is operated within a narrow range for a certain period of time, the model will adapt excessively and will not function in a sufficiently wide range of operating conditions. In addition, recursive methods cannot cope with abrupt changes in process characteristics.

On the other hand, the individuality of production devices should be taken into account. In semiconductor processes, for example, parallelized production devices are used, and they have different characteristics even if their catalog specifications are the same. Therefore, a soft-sensor developed for one device is not always applicable to another device, and it is very laborious to customize soft-sensors according to their individuality.

The Just-In-Time (JIT) modeling has been proposed to cope with process nonlinearity (Bontempi et al. (1999) and Atkeson et al. (1997)) and changes in process characteristics (Cheng and Chiu (2004)). In the JIT modeling, a local model is built from past data around the query only when an estimate is required. The JIT modeling is useful when global modeling does not function well. However, its estimation performance is not always high because the samples used for local modeling are selected on the basis of the distance from the query and the correlation among variables is not taken into account. How should we determine the samples used for local modeling to build

a highly accurate statistical model? Distance is not the most important. A good model cannot be developed when correlation among input-output variables is weak, even if the distance between samples is very small. Conversely, a very accurate model can be developed when the correlation is strong even if the distance is large.

Recently, a new JIT modeling method based on the correlation among variables, referred to as the correlation-based JIT (CoJIT) modeling, has been proposed by Fujiwara et al. (2008). In the CoJIT modeling, the samples used for local modeling are selected on the basis of correlation together with distance. The CoJIT modeling can cope with abrupt changes of process characteristics and also achieve high estimation performance. However, it is applicable to only time-series data because it uses moving windows to generate data sets for local modeling. In other words, the original CoJIT modeling cannot generate a data set consisting of such data that represent characteristics of a query sample and are obtained from various devices operated in parallel.

To make the CoJIT modeling applicable to soft-sensor design for parallelized production devices, samples obtained from various devices have to be discriminated on the basis of the correlation among variables. This discrimination problem is one of the unsupervised pattern recognition problems because the teacher signal is not used for sample classification.

The Nearest Neighbor (NN) method and k-means method are well-known conventional unsupervised pattern recognition algorithms. The NN method can detect samples that are similar to the query, and k-means method can cluster samples without the teacher signal. However, they are distance-based methods and do not take into account the correlation among variables. Self organizing map (SOM) also has been used as an unsupervised pattern recognition method (Kohonen (2001)). SOM is a machine learning process that imitates the brain ' s learning process, and it not only can classify samples but also can visualize high dimensional data. However, it requires high computational load, and the preprocessing data are complicated.

In the present work, to cope with the individuality of production devices as well as changes in process characteristics, a new unsupervised pattern recognition method based on the correlation among variables, referred to as the Nearest Correlation (NC) method, is proposed. The proposed NC method can detect samples that have correlation similar to the query on the basis of sample geometry. In addition, the proposed NC method is integrated with the CoJIT modeling. The usefulness of the integration is demonstrated through a case study of a parallelized CSTR process.

2. INDICES OF CORRELATION

In this section, several measures for quantifying correlation among variables are briefly explained.

2.1 Correlation coefficient

The correlation coefficient $C_{i,j}$ can be used as an index of the similarity between two vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j \in \Re^M$.

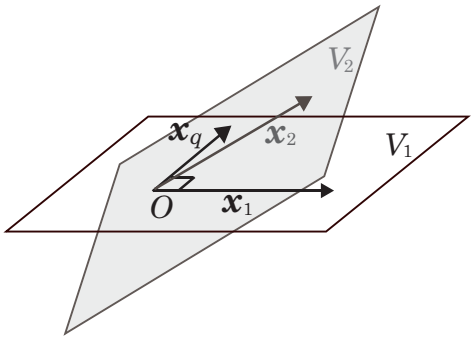

Fig. 1. An example of vector geometry in 3-dimensional space

$$C_{i,j} = \frac{\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j}{||\boldsymbol{x}_i|| ||\boldsymbol{x}_j||} = \cos\theta \tag{1}$$

where, θ is the angle between two vectors.

Suppose that the samples in the three-dimensional data consist of two classes K_1 and K_2, and samples belonging to classes K_1 and K_2 span the two-dimensional linear subspaces V_1 and V_2, respectively, as shown in Fig. 1.

Now, the query $\boldsymbol{x}_q$ is newly measured, and its class should be identified as K_1 or K_2. The correlation coefficients can be used as the index of sample discrimination. For example, $\boldsymbol{x}_1 \in K_1$ and $\boldsymbol{x}_2 \in K_2$ are selected from each class in a random manner, and the correlation coefficients between $\boldsymbol{x}_q$ and them are calculated respectively, and the class including the sample with the largest correlation coefficient can be identified as the class of $\boldsymbol{x}_q$.

In many cases, however, this method is inappropriate. In Fig. 1, the selected sample $\boldsymbol{x}_1$ and the query $\boldsymbol{x}_q$ are orthogonal to each other even though both vectors belong to K_1. In such a case, $\boldsymbol{x}_q$ is identified as an element of K_2 because the correlation coefficient between $\boldsymbol{x}_q$ and $\boldsymbol{x}_2$ is larger than the correlation coefficient between $\boldsymbol{x}_q$ and $\boldsymbol{x}_1$.

2.2 The Q statistic

In this work, the Q statistic is used as an index of sample discrimination.

The Q statistic is derived by principal component analysis (PCA), and it expresses the distance between the sample and the subspace spanned by principal components (Jackson and Mudholkar, 1979). The Q statistic is defined as

$$Q = \sum_{m=1}^{M} (x_m - \hat{x}_m)^2 \tag{2}$$

where x_m and $\hat{x}_m$ are the mth measurement and its estimate by the PCA model, respectively. The Q statistic is a measure of dissimilarity between the sample and the modeling data from the viewpoint of the correlation among variables.

In addition, to take into account the distance between the sample and the origin, Hotelling's T^2 statistic can be used. The T^2 statistic is defined as

$$T^2 = \sum_{r=1}^{R} \frac{t_r^2}{\sigma_{t_r}^2} \tag{3}$$

where σ_{t_r} denotes the standard deviation of the rth score t_r. The T^2 statistic expresses the normalized distance from the origin in the subspace spanned by principal components. The Q and T^2 statistics can be integrated into a single index for sample selection as proposed by Raich and Cinar (1994):

$$J = \lambda T^2 + (1-\lambda)Q \tag{4}$$

where $0 \leq \lambda \leq 1$.

3. NEAREST CORRELATION METHOD

The NN method and the k-means method can discriminate or cluster samples on the basis of the distance without a teacher signal. However, they do not take into account the correlation among variables. In this section, a new unsupervised pattern recognition method based on the correlation among variables, referred to as the nearest correlation (NC) method, is proposed. In the proposed NC method, sample geometry is used for sample discrimination.

3.1 Concept of the NC method

Suppose that the hyper-plane P in Fig. 2 (left) expresses the correlation among variables and the samples on P have the same correlation. Although samples $\boldsymbol{x}_1$ to $\boldsymbol{x}_5$ have the same correlation and they are on P, samples $\boldsymbol{x}_6$ and $\boldsymbol{x}_7$ have different correlation from the others. The NC method aims to detect samples whose correlation is similar to the newly measured query $\boldsymbol{x}_q$. In this example, $\boldsymbol{x}_1$ to $\boldsymbol{x}_5$ on P should be detected.

At first, the whole space is translated so that the query becomes the origin. That is, $\boldsymbol{x}_q$ is subtracted from all samples $\boldsymbol{x}_i (i = 1, 2, \cdots, 7)$. Since the hyper-plane P is translated to the plane containing the origin, it becomes the linear subspace V.

Next, a line connecting each sample and the origin is drawn. Suppose another sample can be found on this line. In this case, $\boldsymbol{x}_1$-$\boldsymbol{x}_4$ and $\boldsymbol{x}_2$-$\boldsymbol{x}_3$ satisfy such a relationship as shown in Fig. 2 (right). The correlation coefficients of these pairs of samples must be 1 or -1. On the other hand, $\boldsymbol{x}_6$ and $\boldsymbol{x}_7$ that are not the elements of V cannot make such pairs. Therefore, the samples of the pairs whose correlation coefficients are ± 1 are thought to have the same correlation as $\boldsymbol{x}_q$.

However, $\boldsymbol{x}_5$ that does not make a pair cannot be detected by this method even though it is on V. To detect $\boldsymbol{x}_5$, a linear subspace is derived from the selected pairs by using PCA, and the derived linear subspace corresponds to V.

Finally, the Q statistics for all samples $\boldsymbol{x}_i$ $(i = 1, 2, \cdots, 7)$ are calculated by using the PCA model expressing V. The samples with small Q statistics are located close to the linear subspace V, and such samples have correlation similar to the query. Although $\boldsymbol{x}_5$ cannot be detected in the previous step, it can be detected in this step because its Q statistic is 0. On the other hand, $\boldsymbol{x}_6$ and $\boldsymbol{x}_7$ are not detected in this step since they have large Q statistics.

In addition, the T^2 statistic can be used to take into account the distance from the origin. In the present work,

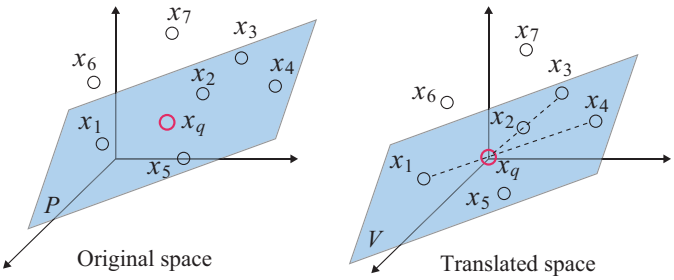

Fig. 2. An example of the procedure of the NC method

J in Eq. (4) is used as the index for sample selection. The samples with small J are selected as the samples similar to the query.

In the implementation of the above procedure, the threshold of the correlation coefficient γ $(1 \geq \gamma > 0)$ has to be used since there are no pairs whose correlation coefficient is strictly ± 1. That is, the pairs should be selected when the absolute values of their correlation coefficients are larger than γ.

3.2 Algorithm of the NC method

Assume that the samples stored in the database are $\boldsymbol{x}_n \in \Re^M$ $(n = 1, 2, \cdots, N)$ and the query is $\boldsymbol{x}_q \in P$ $(\dim(P) = R)$. The samples belonging to P should be detected in a manner similar to $\boldsymbol{x}_q$. The algorithm of the proposed NC method is as follows:

(1) Set R, $\gamma(1 \geq \gamma > 0)$, $\delta(\delta > 0)$ and K or $\bar{J}$.
(2) $\boldsymbol{x}'_n = \boldsymbol{x}_n - \boldsymbol{x}_q$ for $n = 1, 2, \cdots, N$.
(3) Calculate the correlation coefficients $C_{k,l}$ between all possible pairs of $\boldsymbol{x}'_k$ and $\boldsymbol{x}'_l$ $(k \neq l)$.
(4) Select the pairs satisfying $|C_{k,l}| \geq \gamma$, and set the number of the selected pairs S.
(5) If $S < R$, then $\gamma = \gamma - \delta$ $(\delta > 0)$ and return to step 4. If $S \geq R$, then go to the next step.
(6) Arrange the samples of the pairs selected in step 4 as the rows of the matrix $\boldsymbol{X}'$.
(7) Derive the linear subspace V from $\boldsymbol{X}'$ by using PCA. The number of principal components is R.
(8) Calculate the index J of $\boldsymbol{x}'_n$, and $J_n = J$ for $n = 1, 2, \cdots, N$.
(9) Detect the first K samples in ascending order of J_n or the samples whose J_n is smaller than $\bar{J}$ as samples similar to the query $\boldsymbol{x}_q$, where $\bar{J}$ is the threshold.

In step 5, when S is smaller than R, the threshold γ has to be relaxed to increase the number of selected pairs since the linear subspace V is not spanned by the samples of the selected pairs. R can be used as the tuning parameter.

3.3 Numerical example

The discrimination performance of the proposed NC method is compared with that of the NN method through a numerical example. In this example, data consist of three classes that have different correlations, and the samples belonging to the same class as the query should be detected. The discrimination rate is defined as

$$\text{Discrimination Rate } [\%] = \frac{L}{K} \times 100 \tag{5}$$

where K is the number of detected samples and L ($L \leq K$) is the number of samples that belong to the same class as the query among the detected samples. Samples in each of three classes are generated by using the following equation.

$$\boldsymbol{x}_i = \boldsymbol{A}_i \boldsymbol{s} + \boldsymbol{n} \ (i = 1, 2, 3) \tag{6}$$

$$\boldsymbol{s} = [s_1 \ s_2 \ s_3]^T \tag{7}$$

$$\boldsymbol{n} = [n_1 \ n_2 \ n_3]^T \tag{8}$$

where $\boldsymbol{A}_i$ is a coefficient matrix, $s_i \sim N(0, 10)$ and $n_i \sim N(0, 0.1)$. $N(m, \sigma)$ is the random number following the normal distribution whose mean is m and standard deviation is σ. The coefficient matrices are as follows:

$$\boldsymbol{A}_1 = \begin{bmatrix} 1 & 2 \\ 1 & 4 \\ 1 & 1 \\ 2 & 3 \\ 1 & 3 \end{bmatrix}, \ \boldsymbol{A}_2 = \begin{bmatrix} 3 & 3 \\ 2 & 1 \\ 3 & 1 \\ 3 & 2 \\ 2 & 0 \end{bmatrix}, \ \boldsymbol{A}_3 = \begin{bmatrix} 2 & 1 \\ 3 & 4 \\ 1 & 3 \\ 0 & 4 \\ 3 & 1 \end{bmatrix}. \tag{9}$$

100 samples are generated in each of three classes. In addition, a query belonging to each class is prepared. The number of detected samples K is fixed at 20.

In this example, the number of principal components is $R = 2$, the threshold is $\gamma = 1 - 10^{-4}$, the parameters are $\lambda = 0$ and $\delta = 0.9999$. Sample generation and sample detection by the NN method and the NC method are repeated 100 times and the average discrimination rates [%] and the average CPU time [ms] are calculated. The computer configuration used in this numerical example is as follows: OS: Windows Vista Business (64bit), CPU: Intel Core2 Duo 6300 (1.86GHz×2), RAM: 2G byte, and MATLAB® 7.5.0 (2008a).

Table 1 shows the discrimination results of the NN method and the NC method. The proposed NC method can achieve higher discrimination performance than the NN method. On the other hand, the computational load of the NC method is relatively heavy since singular value decomposition (SVD) is used for calculating the correlation among variables. In fact, the computation of SVD occupies most of the computation time of the NC method.

4. CORRELATION-BASED JUST-IN-TIME MODELING

The conventional JIT modeling uses the distance for sample selection when a temporary local model is constructed. However, its estimation performance is not always high since it does not take into account the correlation among variables. Recently, The Correlation-based JIT (CoJIT) modeling that selects samples for local modeling on the basis of the correlation among variables has been proposed by Fujiwara et al. (2008).

Figure 3 shows the difference of sample selection for local modeling between the JIT modeling and the CoJIT modeling. The samples are classified into two groups that have

Table 1. Discrimination performance of the NC method and the NN method

	Discrimination rate [%]			CPU time [ms]
	Class 1	Class 2	Class 3	
NC method	97.5	95.9	96.9	13.9
NN method	78.7	68.0	51.1	1.1

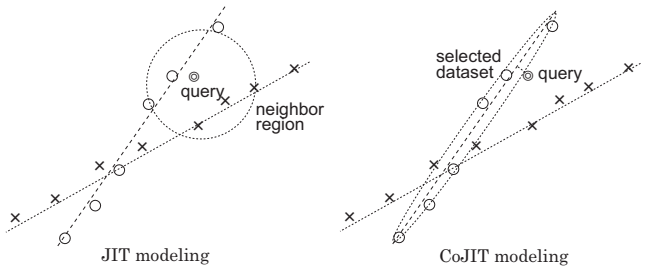

Fig. 3. Sample selection in the JIT modeling (left) and the CoJIT modeling (right)

different correlations. In conventional JIT modeling, samples are selected regardless of the difference of correlation as shown in Fig. 3 (left), since a neighbor region around the query point is defined only by distance. On the other hand, the CoJIT modeling can select samples whose correlation is best fit for the query as shown in Fig. 3 (right).

The procedure of the CoJIT modeling is as follows: 1) several data sets are generated from data stored in the database. 2) The index J is calculated from the query and each data set. 3) The data set whose J is the smallest is selected. 4) A temporary local model is constructed from the selected data set. In the above procedure, each data set is generated so that it consists of successive samples included in a certain period of time, because the correlation in such a data set is expected to be very similar (Fujiwara et al. (2008)).

However, the NC method can detect samples that have correlation similar to the query regardless of whether the objective data is time-series data or not. This is the motivation for integrating the proposed NC method with the CoJIT modeling.

Assume that the sampling interval of the output is longer than that of the input, and the output at time t, $\boldsymbol{y}_t$, should be estimated. Now, the input and the output measured at the same time are stored in the database, and the sth input-output sample $\boldsymbol{x}^{\{s\}} \in \Re^M$ ($s = 1, 2, \cdots, S$) and $\boldsymbol{y}^{\{s\}} \in \Re^L$ are stored as matrices $\boldsymbol{X}_S \in \Re^{S \times M}$ and $\boldsymbol{Y}_S \in \Re^{S \times L}$, respectively. To cope with process dynamics, measurements at different sampling times can be included in $\boldsymbol{x}^{\{s\}}$. The algorithm of the proposed CoJIT modeling with the NC method is as follows:

(1) When the input at time t, $\boldsymbol{x}_t$, is measured, the index J is calculated from $\boldsymbol{x}_t$ and $\boldsymbol{X}_{t-1}$ that was used for building the previous local model f_{t-1}, and $J_I = J$.
(2) If $J_I \leq \bar{J}_I$, $f_t = f_{t-1}$, $\boldsymbol{X}_t = \boldsymbol{X}_{t-1}$, and f_t is used for estimating the output $\boldsymbol{y}_t$. Then, return to step 1. If $J_I > \bar{J}_I$, go to the next step. Here, $\bar{J}_I$ is the threshold.
(3) K input samples whose correlation is similar to the query are detected from $\boldsymbol{X}_S$ by the NC method, and they are arranged as the rows of $\boldsymbol{X}_t \in \Re^{K \times M}$. In addition, K output samples corresponding to the detected input samples are selected from $\boldsymbol{Y}_S$, and they are arranged as the rows of $\boldsymbol{Y}_t \in \Re^{K \times L}$, where K is the number of the detected samples.
(4) A new local model f_t whose input is $\boldsymbol{X}_t$ and output is $\boldsymbol{Y}_t$ is built.
(5) The output $\boldsymbol{y}_t$ is estimated by using f_t.

(6) The above steps 1 through 5 are repeated until the next output sample $\boldsymbol{y}_{S+1}$ is measured. When $\boldsymbol{y}_{S+1}$ is measured, $\boldsymbol{y}_{S+1}$ and its corresponding input $\boldsymbol{x}_{S+1}$ are stored in the database, and return to step 1.

In the above algorithm, any modeling method can be used for building a local model f. In the present work, partial least squares regression (PLS) is used to cope with the colinearity problem. In addition, steps 1 and 2 control the model update frequency. When the threshold $\bar{J}_I$ is large, the update frequency becomes low. The local model is updated every time when new input measurements are available in the case where $\bar{J}_I = 0$.

5. CASE STUDY

In this section, the estimation performance of the proposed CoJIT modeling with the NC method is compared with that of the conventional JIT modeling through their applications to product composition estimation for a parallelized CSTR process. The detailed CSTR model used in this case study is described in Johannesmeyer and Seborg (1999).

5.1 Problem setting

In this process, CSTR1 and CSTR2 are operated in parallel. Although these CSTRs have the same structure

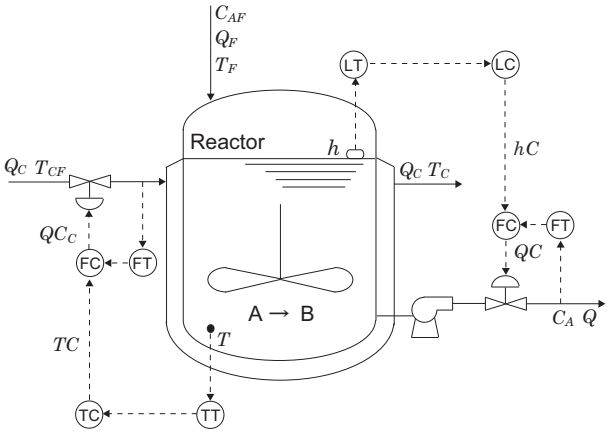

Fig. 4. Schematic diagram of CSTR with cascade control systems

Table 2. Process variables of the CSTR processes

Variable	Caption
C_A	Reactant concentration [mol/m^3]
T	Reactor temperature [K]
T_C	Coolant temperature [K]
h	Reactor level [m]
Q	Reactor exit flow rate [m^3/min]
Q_C	Coolant flow rate [m^3/min]
Q_F	Reactor feed flow rate [m^3/min]
C_{AF}	Feed concentration [mol/m^3]
T_F	Feed temperature [K]
T_{CF}	Coolant feed temperature [K]
hC	Level controller instruction
QC	Outlet flow rate controller instruction
TC	Temperature controller instruction
QC_C	Colorant flow rate controller instruction
T_{set}	Reactor temperature set point [K]

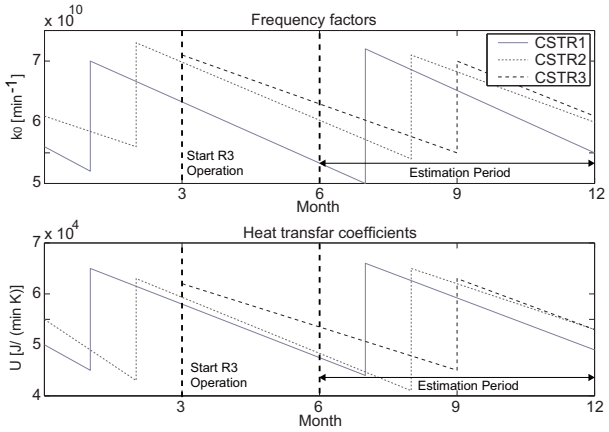

Fig. 5. Changes of overall heat transfer coefficients and frequency factors of the CSTRs

as shown in Fig. 4, they have different characteristics. In each CSTR, an irreversible reaction A $\longrightarrow$ B takes place. The set point of the reactor temperature $T^{[d]}(d = 1, 2)$ is independently changed between $\pm$2K every ten days. Although 15 process variables listed in Table 2 are calculated in the simulations, measurements of only five variables $T^{[d]}$, $h^{[d]}$, $Q^{[d]}$, $Q_C^{[d]}$, $Q_F^{[d]}$ are used for analysis, and their sampling interval is one minute. In addition, reactant concentration $C_A^{[d]}$ is measured in a laboratory once a day.

In this case study, to take into account catalyst deactivation and fouling as changes in process characteristics and individuality of each CSTR, the frequency factor $k_0^{[d]}$ and the heat transfer coefficient $UAc^{[d]}$ are assumed to decrease with time. In addition, each CSTR is maintained every half year (180 days). Figure 5 shows changes of the frequency factors $k_0^{[d]}$ and heat transfer coefficients $UAc^{[d]}$. The operation data of each CSTR for the half years (180 days) were stored in the database.

The soft-sensor for estimating reactant concentration of the newly developed CSTR3 is designed. The estimation of CSTR3 starts the 90th day after the start of its operation, and the soft-sensor is updated in the next half year. Although CSTR3 has only a small amount of data due to its short operation term, the soft-sensor is updated searching samples similar to the current operation of CSTR3 from the other CSTR operation data in the past.

5.2 Estimation result

The reactant concentration $C_A^{[3]}$ is estimated by the JIT modeling and the proposed CoJIT modeling with the NC method. To take into account process dynamics, the input data consist of the present sample and the sample measured one minute before.

In the JIT modeling, linear local models are built and Euclidean distance is used as the measure for selecting samples to build local models. The MATLAB Lazy Learning Toolbox developed by Bontempi et al. (1999) is used.

In the CoJIT modeling, samples for local modeling are selected by the NC method, and PLS is used for model building. The parameters of the NC method are deter-

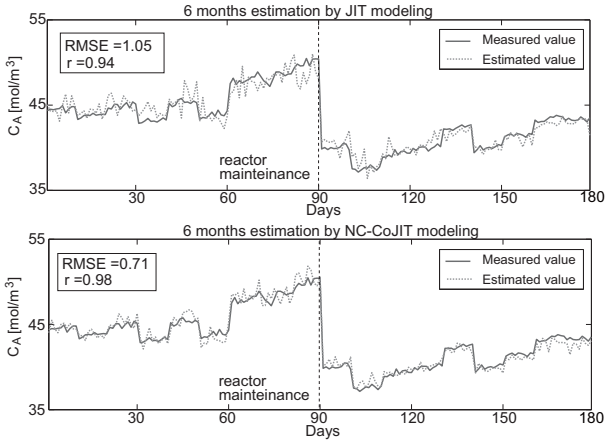

Fig. 6. Prediction result of $C_A^{\{3\}}$ by the JIT modeling (top) and the CoJIT modeling (bottom)

mined by trial and error, the threshold is $\gamma = 1 - 10^4$, the parameter is $\lambda = 0.01$, and the parameter for update frequency $\bar{J}_I = 0$.

The soft-sensor design results are shown in Fig. 6. Although $C_A^{[3]}$ is estimated every minute, only estimates corresponding to the measurements are plotted. In this figure, r denotes the correlation coefficient between measurements and estimates, and RMSE is the root-mean-squares error.

This result shows that the JIT modeling does not function well. On the other hand, the estimation performance of the proposed CoJIT modeling with the NC method is very high. With the proposed CoJIT modeling, RMSE is improved by about 35% in comparison with the JIT modeling. These results of this case study clearly show that the proposed CoJIT modeling can cope with not only abrupt changes in process characteristics but also the individuality of production devices. In addition, it can construct a high performance soft-sensor for a newly develop device, even if only a small amount of operation data is available.

6. CONCLUSION

A new unsupervised pattern recognition method that can detect samples whose correlation is similar to the query is proposed. In addition, the JIT modeling is integrated with the proposed the NC method. The proposed CoJIT modeling with the NC method can cope with not only changes in process characteristics but also the individuality of production devices and improve the estimation performance of a soft-sensor since it can select samples for local modeling by appropriately accounting for the correlation among variables. The proposed CoJIT modeling has the potential for realizing efficient maintenance of soft-sensors.

REFERENCES

Atkeson, CG., Moore, AW., and Schaal, S. (1997). Locally Weighted Learning, Artificial Intelligence Review, 11, 11-73.

Amirthalingam, R., and Lee J. (1999). Subspace Identification Based Inferential Control Applied to a Continuous Pulp Digester. J Proc Cont, 9,397-406.

Bontempi, G., Birattari, M., and Bersini, H. (1999). Lazy Learning for Local Modeling and Control Design. Int J Cont, 72, 643-658.

Bontempi, G., Birattari, M., and Bersini, H. (1999). Lazy Learners at Work: The Lazy Learning Toolbox. EUFIT'99: The 7th European Congress on Intelligent Techniques and Soft Computing, Aachen, Germany. Sep.13-16.

Cheng C., and Chiu, MS. (2004). A New Data-Based Methodology for Nonlinear Process Modeling. Chem Engng Sci, 59, 2801-2810.

Fujiwara, K., Kano, M., and Hasebe, S. (Accepted). Soft-Sensor Development Using Correlation-Based Just-In-Time Modeling. AIChE J.

Jackson, JE., and Mudholkar, GS. (1979). Control Procedures for Residuals Associated with Principal Component Analysis. Technometrics, 21, 341-349.

Johannesmeyer, M., and Seborg, DE. (1999). Abnormal Situation Analysis Using Pattern Recognition Techniques and Histrical Data. AIChE Annual meeting, Dallas, TX, Oct.31-Nov.5.

Kamohara, H., Takinami. A., Takeda. M., Kano, M., Hasebe, S., and Hashimoto, I. (2004). Product Quality Estimation and Operating Condition Monitoring for Industrial Ethylene Fractionator. J Chem Eng Japan, 37, 422-428.

Kano, M., Lee, S., and Hasebe, S. (2008). Two-Stage Subspace Identification for Softsensor Design and Disturbance Estimation. J Proc Cont, 1016/j.jprocont.2008.04.004.

Kano, M., Miyazaki, K., Hasebe, S., and Hashimoto, I. (2000). Inferential Control System of Distillation Compositions Using Dynamic Partial Least Squares Regression. J Proc Cont, 10, 157-166.

Kano, M., and Nakagawa, Y. (2008). Data-Based Process Monitoring, Process Control, and Quality Improvement. Recent Developments and Applications in Steel Industry, Comput Chem Engng, 32, 12-24.

Kohonen, T. (2001). Self-organizing maps. New York, Springer, 3rd edition.

Kresta, VJ., Marlin, TE., and MacGregor, JF. (1994). Development of Inferential Process Models Using PLS. Comput Chem Engng, 18, 597-611.

Mejdell, T., Skogestad, S. (1991). Estimation of Distillation Compositions from Multiple Temperature Measurements Using Partial-Least-Squares Regression. Ind Eng Chem Res, 30, 2543-2555.

Ogawa, M., and Kano, M. (2008). Practice and Challenges in Chemical Process Control Applications in Japan. The 17th IFAC World Congress, Paper WeC25.3.

Qin, SJ. (1998) . Recursive PLS Algorithms for Adaptive Data Modeling. Comput Chem Engng, 22, 503-514.

Raich, A. and Cinar, A. (1994). Statistical Process Monitoring and Disturbance Diagnosis in Multivariable Continuous Processes. AIChE J, 42, 995-1009.

Radhakrishnan, V., and Mohamed, A. (2000). Neural networks for the identification and control of blast furnace hot metal quality. J Proc Cont, 10,509-524.

Vesanto, J., Himberg, J., Alhoniemi, E., and Parhankangas, J. (1999). Self-Organizing Map in Matlab: the SOM Toolbox. The Matlab DSP Conference 1999, Espoo, Finland, Nov.16-17.