2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Soft Sensor Development Using Non-Gaussian Just-In-Time Modeling

Jiusun Zeng, Lei Xie, Chuanhou Gao and Jingjing Sha

Abstract— **This paper introduces a novel Just-In-Time (JIT) learning based soft sensor for modeling of non-Gaussian process. Most of JIT modeling uses distance based similarity measure for local modeling, which may be inappropriate for many industrial processes exhibiting non-Gaussian behaviors. Since most of industrial processes are non-Gaussian, a non-Gaussian regression (NGR) technique is used to extract non-Gaussian independent components that are correlated to response variable in the sense of mutual information. Support vector data description (SVDD) is then performed on the extracted independent components to construct a new similarity measure. Based on the similarity measure, a novel JIT modeling procedure is proposed. Application studies on a numerical example as well as an industrial process confirm that the proposed JIT model can achieve good predictive accuracy.**

I. INTRODUCTION

During the last decades, development of soft sensor for the task of on-line prediction, process monitoring and fault detection have attracted much attention. In many industrial settings, some important process and quality variables cannot be measured in real time [1] [2]. To overcome the lack of online measurements for such parameters, soft sensor has emerged as one of the most important tools, among which data-driven models are the most popular in the process industry. Data-driven methods like principal component analysis (PCR), partial least squares (PLS), artificial neural network (ANN) and support vector machine (SVM) have been successfully applied to a series of industrial applications [3 6].

Despite their popularity, construction of high performance soft sensor is not an easy task. Even though a good soft sensor is obtained, the performance will deteriorate after a certain time due to change of process characteristics. Therefore, soft sensor should be updated regularly to ensure a good performance, which would be a laborious task. To get a soft sensor which can be updated automatically, different kinds of recursive methods have been introduced, such as recursive PCR, recursive PLS [7] [8]. However, recursive methods cannot adapt to abrupt change of process characteristics as well as switch of operation conditions in time. Alternatively, Just-In-Time (JIT) learning [9 13] were proposed to deal with such kind of situations. In JIT modelling, local models are built from historical data when an estimated value is requested. By using local models, the current operation condition can be well tracked. Most JIT modelling techniques select samples for local modelling using distance-based similarity measure. However, distance-based similarity measure doesn't take the correlation among variables into account. As an alternative, correlation-based similarity measure [14] was proposed to develop soft sensors,which is based on the Q and T^2 statistics of principal component analysis (PCA). By using PCA, the method in [14] assumes the data is Gaussian-distributed, which is not valid for many industrial applications [15]. In contrast to the work in [14], this article proposes a novel JIT modelling technique by defining a new similarity measure suitable for non-Gaussian data. The similarity measure is constructed using support vector data description (SVDD) [16]. To utilize the non-Gaussian information in process data, the newly developed non-Gaussian regression (NGR) method [17] is used to build the local model. The new soft sensor takes the non-Gaussianity of process data into account and hence better modelling effects can be achieved.

The rest of the paper is as follows. In the next section, fundamentals of NGR and SVDD are introduced. The proposed JIT modelling technique is then proposed, followed by the simulation and application studies in Section 4. Finally, Section 5 gives a concluding summary of the presented article.

II. PRELIMINARIES

In this section, the principles of NGR and SVDD are briefly explained.

A. Non-Gaussian Regression Method

The NGR algorithm adopted in this article is developed in [17]. The purpose of the NGR algorithm is double folded. The first objective is to extract non-Gaussian components from the predictor and response variable sets, which is the same as the objective of ICA; the second objective is to maximize mutual information between extracted components and response variables. By considering the dual objective function, the algorithm considers higher-order statistics information between extracted components and response variable, and thus more suitable for modelling non-Gaussian process.

This work was supported in part by National Natural Science Foundation of China under Grant No. 60904039, 10901139, 60911130510, 60974100) and the Public Benefit Technologies R&D Program of Science and Technology Department of Zhejiang Province under Grant No. 2011C21020.

Jiusun Zeng is with the institute of Cyber Systems & Control, Zhejiang University, Hangzhou, 310027, China. He is also with College of Metrological & Measurement Engineering, China Jiliang University, Hangzhou, 310018, China. email:zjsl020@gmail.com

Lei Xie is with the Institute of Cyber Systems & Control, Zhejiang University, Hangzhou, 310027, China phone: 86-571-87952268; fax: 86-571-87952279, email: leix@csc.zju.edu.cn

Chanhou Gao is with the Department of Mathematics, Zhejiang University, Hangzhou, 310027, China email: gaochou@zju.edu.cn

Jingjing Sha is with the Institute of Cyber Systems & Control, Zhejiang University, Hangzhou, 310027, China email: shajj@zju.edu.cn

978-1-61284-799-3/11/$26.00 ©2011 IEEE

Denote the predictor and response variable sets to be $\mathbf{x} \in \mathbb{R}^M$ and $\mathbf{x} \in \mathbb{R}^N$ respectively. Similar to ICA, the measured predictor variable set $\mathbf{x} = (\mathbf{x}_1\mathbf{x}_2 \ldots \mathbf{x}_M)^T$ is assumed to be described by a linear combination of a set of $n(n \leq M)$ independent components (ICs) $\mathbf{s} = (\mathbf{s}_1\mathbf{s}_2 \ldots \mathbf{s}_n)^T$ as follows

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{e} \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{M \times n}$ is the mixing matrix,$\mathbf{e}$ is a Gaussian distributed residual vector with zero mean and covariance $\mathbf{\Sigma_e}$,$\mathbf{e} \sim \mathbb{N}(\mathbf{0}, \mathbf{\Sigma_e})$. The ICs can be extracted by estimating an orthogonal demixing matrix $\mathbf{W} \in \mathbb{R}^{M \times n}$, such that

$$\hat{\mathbf{s}} = \mathbf{W}^T\mathbf{x} = \mathbf{W}^T\mathbf{A}\mathbf{s} \approx \mathbf{s} \tag{2}$$

The process of determining the demixing matrix $\mathbf{W}$ is often preceded by a pre-whitening transform on the predictor variable set by applying principal component analysis (PCA)

$$\hat{\mathbf{s}} = \mathbf{W}^T\mathbf{z} = \mathbf{W}^T\mathbf{Q}\mathbf{x}$$

$$= \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_n^T \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \lambda_b \end{bmatrix}^{-\frac{1}{2}} \begin{bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \vdots \\ \mathbf{p}_n^T \end{bmatrix} \mathbf{x} \tag{3}$$

where $\mathbf{Q}$ is the whitening matrix, $\mathbf{P} = (\mathbf{p}_1\mathbf{p}_2 \cdots \mathbf{p}_n)^T \in \mathbb{R}^{M \times b}$ is an orthogonal matrix of eigenvectors,$\lambda_1, \ldots, \lambda_b$ are the first b eigenvalues of the covariance matrix $E(\mathbf{x}\mathbf{x}^T)$.The pre-whitening transform ensures that each elements of $\hat{\mathbf{s}}$ has zero mean and unit variance.

The NGR algorithm relies on information theoretic measures like negentropy and mutual information. The negentropy of a random variable θ is defined as

$$J(\theta) = H(\upsilon) - H(\theta) \tag{4}$$

where υ is a Gaussian distributed random variable with the same mean and covariance as θ. $H(\upsilon)$ and $H(\theta)$ are the entropy of υ and θ. Given the probability density function $p(\theta)(\theta \in \Theta \subseteq \mathbb{R})$, the entropy is defined as

$$H(\theta) = -\int_{\Theta} p(\theta)\log(p(\theta))d\theta \tag{5}$$

A Gaussian variable υ has the greatest entropy among all random variables of equal variance, so that maximizing Eq. (4) leads to maximizing the non-Gaussianity of θ. The negentropy can be approximated by

$$J(\theta) \approx [E\{G(\theta)\} - E\{G(\upsilon)\}]^2 \tag{6}$$

Here,$G(\cdot)$ is a nonquadratic function, the proposed functions for $G(\cdot)$ incluede [18]:

$$G_1(\theta) = \frac{1}{a_1}\log\cosh(a_1\theta) \quad G_2(\theta) = -\exp(-\frac{a_2\theta^2}{2})$$

where $1 \leq a_1 \leq 2, a_2 = 1$. The mutual information between θ and a random variable $\phi \in \Phi \subseteq \mathbb{R}$ is defined as

$$I\left(\theta,\phi\right) = \int_{\Theta}\int_{\Phi} p_{\theta\phi}\left(\theta,\phi\right)\log\left(\frac{p_{\theta\phi}\left(\theta,\phi\right)}{p_{\theta}\left(\theta\right)p_{\phi}\left(\phi\right)}\right)d\phi d\Theta$$

where $p(\theta,\phi)$ is the joint probability density of θ and ϕ,$p(\phi)$ is the marginal density of ϕ. An alternative expression for mutual information can be obtained by using Eq. (5)

$$I(\theta,\phi) = H(\theta) + H(\phi) - H(\theta,\phi)$$

Here,$H(\theta,\phi)$ is the joint entropy of θ and ϕ, defined by

$$H(\theta,\phi) = \int_{\Theta}\int_{\Phi} p_{\theta\phi}\left(\theta,\phi\right)\log\left(p_{\theta\phi}\left(\theta,\phi\right)\right)d\phi d\theta$$

For simplicity, consider the case of $N = 1$ and the response variable y having been normalized. For a specific IC, say $\mathbf{w}_i^T\mathbf{z}$, the i-th weight vector $\mathbf{w}_i$ can be obtained by solving the following dual-objective function

$$\begin{aligned} \widehat{w}_i = \arg\max_{\mathbf{w}_i} \alpha(H(\nu) - H(\mathbf{w}_i^T\mathbf{z})) \\ +\beta(H(\mathbf{w}_i^T\mathbf{z}) + H(y) - H(\mathbf{w}_i^T\mathbf{z}, y)) \end{aligned} \tag{7}$$

where $0 \leq \alpha \leq 1, 0 \leq \beta \leq 1, \alpha + \beta = 1$ are design coefficients. Neglecting constant terms $H(y)$ and $H(\upsilon)$, and then reformulating Eq.(7) yields

$$\begin{aligned} \widehat{\mathbf{w}}_i = \arg\max_{\mathbf{w}_i}(\alpha - \beta)(H(\upsilon) - H(\mathbf{w}_i^T\mathbf{z})) \\ -\beta H(\mathbf{w}_i^T\mathbf{z}, y) \end{aligned} \tag{8}$$

Using Eq. (6), the first term on the right hand side can be easily approximated. Now the key problem becomes approximation of the joint entropy $H(\mathbf{w}_i^T\mathbf{z}, y)$, which can be approximated by Edgeworth expansion [19] as follows

$$\begin{aligned} H(\mathbf{w}_i^T\mathbf{z}, y) = H(\phi_p) - \frac{1}{12}(\kappa^{1,1,1})^2 - \\ \frac{1}{4}(\kappa^{1,1,2})^2 - \frac{1}{4}(\kappa^{1,2,2})^2 - \frac{1}{12}(\kappa^{2,2,2})^2 \end{aligned} \tag{9}$$

Here,Φ_p is the Gaussian estimate of $(\mathbf{w}_i^T\mathbf{z}, y)$, i.e., ϕ_p has the same mean and covariance matrix as $(\mathbf{w}_i^T\mathbf{z}, y)$. $H(\phi_p)$ is the familiar expression for the 2-dimensional entropy:$H(\phi_p) = \frac{1}{2}\log|\Sigma| + \log(2\pi e)$,where $|\Sigma|$ is the determinant of covariance matrix. $\kappa^{1,1,1}, \kappa^{1,1,2}, \kappa^{1,2,2,}$ and $\kappa^{2,2,2}$ are the standardized 3rd order cumulants. Since both $\mathbf{w}_i^T\mathbf{z}$ and y have zero mean and unit variance, the standardized 3rd order cumulants are equal to their corresponding moments, so that we have

$$\begin{aligned} |\Sigma| = 1 - (E\left\{(\mathbf{w}_i^T\mathbf{z})y\right\})^2 \\ \kappa^{1,1,1} = E\left\{(\mathbf{w}_i^T\mathbf{z})^3\right\} \qquad \kappa^{1,1,2} = E\left\{(\mathbf{w}_i^T\mathbf{z})^2 y\right\} \\ \kappa^{1,2,2} = E\left\{\mathbf{w}_i^T\mathbf{z}y^2\right\} \qquad \kappa^{2,2,2} = E\left\{y^3\right\} \end{aligned} \tag{10}$$

The approximation of the joint entropy function therefore becomes

$$\begin{aligned} H\left(\mathbf{w}_i^T\mathbf{z}, y\right) = \log\left(2\pi e\right) + \frac{1}{2}\log\left(1 - \left(E\left\{\left(\mathbf{w}_i^T\mathbf{z}\right)y\right\}\right)^2\right) \\ -\frac{1}{12}\left(\left(E\left\{\left(\mathbf{w}_i^T\mathbf{z}\right)^3\right\}\right)^2 + 3E\left\{\left(\mathbf{w}_i^T\mathbf{z}\right)^2 y\right\}^2 \right. \\ \left. +3E\left\{\left(\mathbf{w}_i^T\mathbf{z}\right)y^2\right\}^2 + \left(E\left\{y^3\right\}\right)^2\right) \end{aligned} \tag{11}$$

Eq. (8) can finally be expressed as the following form

$$\begin{aligned}\widehat{\mathbf{w}}_i = \arg\max_{\mathbf{w}_i}(\alpha-\beta)\left(H(\upsilon)-H(\mathbf{w}_i^T\mathbf{z})\right)\\ -\frac{\beta}{2}\log\left(1-\left(E\left\{\left(\mathbf{w}_i^T\mathbf{z}\right)y\right\}\right)^2\right)+\frac{\beta}{12}\left(E\left\{(\mathbf{w}_i^T\mathbf{z})^3\right\}\right)^2\\ +\frac{\beta}{4}\left(E\left\{(\mathbf{w}_i^T\mathbf{z})^2y\right\}\right)^2+\frac{\beta}{4}\left(E\left\{(\mathbf{w}_i^T\mathbf{z})y^2\right\}\right)^2+\frac{\beta}{12}\left(E\left\{y^3\right\}\right)^2 \quad (12)\end{aligned}$$

and is subject to the following constraints

$$\begin{bmatrix}\widehat{\mathbf{w}}_1^T\\ \widehat{\mathbf{w}}_2^T\\ \vdots\\ \widehat{\mathbf{w}}_i^T\end{bmatrix}\widehat{\mathbf{w}}_i=\begin{bmatrix}0\\0\\ \vdots\\ 1\end{bmatrix} \quad (13)$$

After determining the weight vectors, the maximum can be achieved by using optimization methods like gradient search or particle swarm optimization. The optimization terminates until desired number of ICs have been obtained, which can be determined by making independence test between the extracted IC and the response variable. The ICs can be computed as follows

$$\hat{\mathbf{s}}=\widehat{\mathbf{W}}^T\mathbf{Q}\mathbf{x} \quad (14)$$

The above modelling technique can be easily extended to the case of $N>1$ by considering the non-Gaussianity of response variables in Eq. (7). The regression relationship between the output variable and the extracted ICs can now be determined by using ordinary least squares

$$\mathbf{y}=\hat{\mathbf{c}}\hat{\mathbf{s}}+\mathbf{e} \qquad \hat{\mathbf{c}}=\mathbf{y}\mathbf{z}^T\widehat{\mathbf{W}}\left\{\widehat{\mathbf{W}}^T\mathbf{z}\mathbf{z}^T\widehat{\mathbf{W}}\right\}^{-1} \quad (15)$$

B. Support Vector Data Description

SVDD is a classification tool designed to detect whether new samples resemble the properties of the reference set. It has been recognized as a useful tool in process monitoring to construct monitoring statistics for non-Gaussian data [20] [21]. The main idea is to envelop the data within a feature space by a minimal spherical volume. This volume should contain as many samples as possible. The sphere is determined by its centre $\mathbf{a}$ and radius R. Assume we have L training samples and SVDD is performed on the extracted non-Gaussian components $\{\hat{\mathbf{s}}_i\}, \hat{\mathbf{s}}_i \in \mathbb{R}^n, i=1,2,\ldots,L$.The minimal sphere can be obtained by solving the following optimization problem

$$\begin{aligned}\min F(R,\mathbf{a}) = R^2 + C\sum_i \xi_i\\ \text{s.t.}\|\mathbf{s}_i-\mathbf{a}\|^2 \le R^2+\xi_i, \xi_i \ge 0, i=1,2,\ldots,L \quad (16)\end{aligned}$$

Here,slack variable ξ_i is the penalty term for misclassification; the parameter C controls the trade-off between the volume and errors. The dual problem of Eq. (16) can be derived as follows

$$\begin{aligned}\max_{\mathbf{a}}\sum_i a_i(\hat{\mathbf{s}}_i,\hat{\mathbf{s}}_i)-\sum_i\sum_j a_ia_j(\hat{\mathbf{s}}_i,\hat{\mathbf{s}}_j)\\ \text{s.t.}\sum_i\gamma_i=1, \gamma_i\in[0,C], i=1,2,\ldots,L \quad (17)\end{aligned}$$

The centre $\mathbf{a}$ and radius R of the hypersphere are given by

$$\begin{aligned}\mathbf{a}=\sum_i\gamma_i\hat{\mathbf{s}}_i\\ R=\sqrt{(\hat{s}_k,\hat{s}_k)-2\sum_i\gamma_i(\hat{s}_k,\hat{s}_i)+\sum_i\sum_j(\hat{s}_i,\hat{s}_j)} \quad (18)\end{aligned}$$

where $\hat{\mathbf{s}}_k, k=1,\ldots,K$ are support vectors with $\gamma_k>0$.

After the hypersphere is constructed in the feature space, the hypothesis that a new sample $\bar{\bar{\mathbf{s}}}$ belongs to the reference set is accepted if the following conditions hold

$$\begin{aligned}f(\bar{\bar{\mathbf{s}}})=\|\bar{\bar{\mathbf{s}}}-\mathbf{a}\|^2\le R^2\\ \|\bar{\bar{\mathbf{s}}}-\mathbf{a}\|^2=(\bar{\bar{\mathbf{s}}},\bar{\bar{\mathbf{s}}})-2\sum_i\gamma_i(\bar{\bar{\mathbf{s}}},\hat{\mathbf{s}}_i)+\sum_i\sum_j\gamma_i\gamma_j(\hat{\mathbf{s}}_i,\hat{\mathbf{s}}_j) \quad (19)\end{aligned}$$

To maintain the numerical efficiency in determining the hypersphere, the kernel trick is often employed by replacing $(\hat{\mathbf{s}}_i,\hat{\mathbf{s}}_j)$ in Eq. (17) by the kernel function $K(\hat{\mathbf{s}}_i,\hat{\mathbf{s}}_j)=\exp\left(-\|\hat{\mathbf{s}}_i-\hat{\mathbf{s}}_j\|^2/\sigma^2\right)$

The statistic D can then be constructed to measure the distance between a new sample and the reference set in the hypersphere

$$D=\frac{\bar{\bar{\mathbf{s}}}-\mathbf{a}}{R} \quad (20)$$

Higher D value indicates the new sample is more likely to be from a different data set and hence can serve as the similarity measure.

III. JUST-IN-TIME MODELING BASED ON NGR AND SVDD

The previous section discussed details of NGR and SVDD. This section introduces how to build a JIT model for non-Gaussian data through NGR and SVDD.

Assume n ICs $\hat{\mathbf{s}}=(\hat{\mathbf{s}}_1,\hat{\mathbf{s}}_2,\ldots,\hat{\mathbf{s}}_n)^T$ have been extracted from the predictor variable set $\mathbf{x}=(\mathbf{x}_1\mathbf{x}_2\ldots\mathbf{x}_M)^T$ using the NGR technique and there are a total of L samples. The ICs exhibit high non-Gaussianity and are correlated to response variable $\mathbf{y}$ in the sense of mutual information. By projecting the non-Gaussian components into the kernel space,D statistic for each sample can be calculated using Eq. (20). In addition, Hotelling's T^2 statistic is also included to avoid extrapolation and guarantee that the sample is located in the modeling data. The T^2 statistic is defined as

$$T^2=\sum_{t=1}^{n}\frac{\hat{\mathbf{s}}_t^2}{\delta_{\hat{\mathbf{s}}_t}^2} \quad (21)$$

where $\delta_{\hat{\mathbf{s}}_t}^2$ denotes the standard deviation of the t-th IC $\hat{\mathbf{s}}_t$. When the value of T^2 statistic is small, the sample is close to the mean of the modeling data in the original space. By integrating D and T^2 statistic one can get the following the following index for data set selection

$$\mathfrak{J}=\lambda D+(1-\lambda)T^2 \quad (22)$$

where $0\le\lambda\le1$.

To construct the JIT model, the samples are firstly divided into several data sets; each data set consists of successive samples included in a certain period of time, since characteristics of such data set is expect to be similar. First, NGR is used to extract non-Gaussian components and build local regression models between predictor and response variables for these data sets. As a new sample arrives, the evaluation index $\mathfrak{J}$ in Eq. (22) is calculated and the data set that minimizes $\mathfrak{J}$ is selected as the modeling data to get the required prediction for response variables.

Assume the first through L^{th} input-output measurements are stored in database and $\mathbf{z}^l = \left[\mathbf{x}_l^T, \mathbf{y}_l^T\right] \in \mathbb{R}^{M+N}(l = 1, 2, \dots, L)$. When a newly measured input sample $\mathbf{x}_{L+1}$ is available, one need to estimate the response variables $\hat{\mathbf{y}}_{L+1}$, which should be as close to the real value $\mathbf{y}_{L+1}$ as possible. Denote the data set that was used to build the l^{th} local model to be $\mathbf{Z}^l$ and the local model to be f^l. The procedure of our JIT modeling is as follows:

1) The index $\mathfrak{J}$ is calculated from $\mathbf{x}_{L+1}$ and the data set $\mathbf{Z}^L$ that was used to build the previous local model f^L, denoted to be $\mathfrak{J}_{L+1}$;
2) Determine a threshold $\bar{\mathfrak{J}} \geq 0$,if $\mathfrak{J}_{L+1} \leq \bar{\mathfrak{J}}$, then $f^{L+1} = f^L$ and $\mathbf{Z}^{L+1} = \mathbf{Z}^L$. f^{L+1} is then used to estimate the response variables until a new measurement $\mathbf{x}_{L+2}$ is available; return to Step 1. If $\mathfrak{J}_{L+1} > \bar{\mathfrak{J}}$, $k = 1$, then go to the next step;
3) Extract the k^{th} data st $\mathbf{Z}^k = \left[\mathbf{z}^k, \dots, \mathbf{z}^{k+W-1}\right] \in \mathbb{R}^{(M+N)\times W}$, where W is the window length, which is used to control the number of data samples for modelling;
4) Build the local model f^k using NGR and compute the centre $\mathbf{a}^k$ and the radius R^k; calculate the index $\mathfrak{J}_{L+1}$ between $\mathbf{x}_{L+1}$ and the data set $\mathbf{Z}^k$;
5) $k = k + d$ if $k \leq S - W + 1$, then return to Step 4. If $k > S - W + 1$, then go to the next step. Here d is the window moving width, which is used to control the updating frequency of local models;
6) The data set $\mathbf{Z}^k$ that minimizes $\mathfrak{J}_{L+1}$ is selected and defined as the modeling data set $\mathbf{Z}^{L+1}$; the local model f^k is then used to estimate the response variables $\hat{\mathbf{y}}_{L+1}$;
7) When a new measurement $\mathbf{x}_{L+2}$ is available, return to Step 1.

In the above procedure, the window length W and window width d are used to control the model updating frequency. Meanwhile, by setting a larger threshold $\bar{\mathfrak{J}}$, the update frequency can also be lowered, so that an ergodic modeling process can be avoided. Considering the fact that our modeling process involves an optimization procedure, this becomes even more important. The JIT model can also be extended to model dynamic processes by including measurements at different sampling times into the modeling data.

IV. Application studies

In order to evaluate the performance of the proposed JIT modelling method, we consider both numerical and real examples. The numerical study relates to a simulation example that includes 7 inputs and 1 output. For the real example, we consider the sulfur recovery unit.

A. Numerical study

For the numerical study, the 7 predictor variables are simulated as linear combinations of a total of 5 source variables. The performance of the proposed JIT technique will be compared with that of [14] using different numbers of ICs.

Consider the following five unknown sources

$$\begin{cases} s_1(k) = 2\cos(0.08k)\sin(0.06k) \\ s_2(k) = \sin(0.3k) + 3\cos(0.1k) \\ s_3(k) = \sin(0.4k) + 3\cos(0.1k) \\ s_4(k) = \cos(0.1k) - \sin(0.05k) \\ s_5(k) = \text{uniformly distributed noise in}[-1, 1] \end{cases}$$

Process data are generated from the five source signals as $\mathbf{x}^T = \mathbf{s}^T\mathbf{A} + \mathbf{e}, \mathbf{e} \sim \mathcal{N}(0, 0.01)$.To highlight the effects of process change, 3000 samples belonging to 3 different modes are generated. For the first 1000 samples, the mixing matrix $\mathbf{A}$ is set as follows

$$\mathbf{A} = \begin{bmatrix} 0.86 & -0.55 & 0.17 & -0.33 & 0.89 & 0.2 & 0.8 \\ 0.79 & 0.65 & 0.32 & 0.12 & -0.97 & 0.4 & 0.5 \\ 0.67 & 0.46 & -0.28 & 0.27 & -0.74 & -0.3 & -0.45 \\ 0.23 & 0.15 & 0.56 & 0.84 & 0.23 & 0.13 & 0.14 \\ 0.34 & 0.95 & 0.12 & 0.47 & 0.92 & 0.19 & 0.56 \end{bmatrix}$$

The response data is generated by $y = 0.8x_1 + 0.6x_2 + 1.5x_3$. For the second 1000 samples,$\mathbf{A}$ is left multiplied by the following matrix

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

The response data is generated by $y = 2.4x_2 + 1.6x_3 + 4x_4$. The mixing matrix for the last 1000 samples is set as $\mathbf{B}^2\mathbf{A}$, and the response data is generated by $y = 1.2x_1 + 0.4x_2 + x_4$. Finally, a normally distributed noise is added to the response data,$\mathbf{y} = \mathbf{y} + \mathbf{h}, \mathbf{h} \sim \mathcal{N}(0, 0.1)$. The proposed JIT technique is then used to build local models. From each mode, the first 900 samples are selected and combined into a data set with 2700 samples and the remaining 300 samples are used to test the performance of the JIT technique. The parameters are set as $\alpha = 0.2, \sigma = 1, C = 0.05, \lambda = 0.999$. By setting $\alpha = 0, 2$ higher emphasis is put on mutual information so that less independent components are needed to construct an accurate model. The window length and window width are set as $W = 500, d = 60$ respectively and the threshold $\bar{\mathfrak{J}} = 0$. The number of ICs used in the proposed JIT technique is determined by trial and error. The estimation results are shown in Fig. 1, which shows that the correlation coefficient r is 0.9940 and a root mean square error (RMSE) of 3.1 with 3 ICs extracted.

In contrast, the prediction result using the CoJIT technique proposed in [14] are shown in Fig.2. For the CoJIT technique, the weights for T^2 and Q statistics are 0.01 and 0.99 respectively; and the window length and width are also set as $W = 500, d = 60$. Fig.2 shows a correlation coefficient of 0.9815 and RMSE of 5.38. Both evaluation criterions show

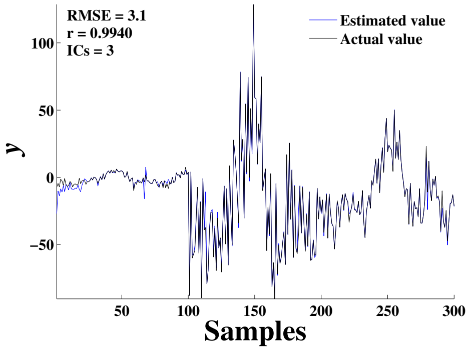

Fig. 1. Prediction results by the proposed JIT for the numerical study

that the proposed JIT technique produces better predictions than CoJIT. Moreover, the proposed JIT technique achieves higher prediction accuracy with only 3 ICs retained, while CoJIT needs to retain 4 PCs.

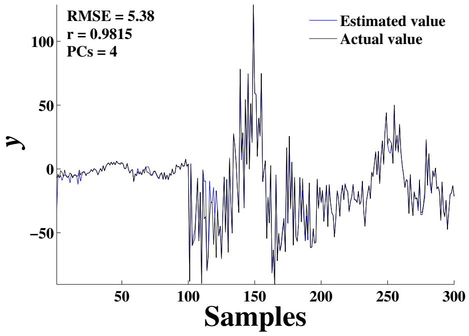

Fig. 2. Prediction results by CoJIT for the numerical study

B. *Sulfur recovery unit*

The sulfur recovery unit (SRU) [22] [23] is an important refinery processing unit which is utilized to remove environmental pollutants from acid gas streams before releasing into the atmosphere. The main environmental pollutant to be removed is hydrogen sulfide which is extremely dangerous. The SRU process considered in this paper is fed by 2 kinds of acid gases. The first kind, called MEA gas, comes from the gas washing plant and is rich in H2S. The second kind, called SWS, comes from the sour water stripping (SWS) plant and is rich in H2S and NH3. The acid gases are fed into the SRU and burnt in reactors, where H2S is transformed into pure sulfur. The combustion products are then cooled and generated to liquid sulfur. The liquid sulfur is further processed to form water vapor and sulfur, which is a valuable byproduct of SRU. The final gas stream (tail gas) is then released to the atmosphere. It should be noted that the tail gas contains residual H2S and SO2, which should be minimized. In order to monitor the performance of SRU and improve the sulfur recovery rate, soft sensors are needed to measure the concentration of H2S(y_1)and SO2(y_2) in the tail gas. To make predictions on the concentration of H2S and SO2, 5 process variables are selected and listed in Table 1.

TABLE I

SELECTED PROCESS VARIABLES

Predictor variable	Variable Description
x_1	MEA gas flow
x_2	first air flow
x_3	second air flow
x_4	gas flow in SWS zone
x_5	air flow in SWS zone

To make a fair comparison with CoJIT, two separate JIT models are built for prediction of H2S and SO2 concentration and a total of 2700 samples are used. To account for process dynamics, the input sample at the present and previous time instance as well as the output sample at the previous time instance are included in the predictor space. So that there are 11 predictor variables and 1 response variable for each model. For prediction of y_1, the parameters are set as $\alpha = 0.2, \sigma = 1, C = 0.01, \lambda = 0.99$. The window length and window width are set as $W = 700, d = 40$ respectively and the threshold $\bar{\mathfrak{J}} = 0$. The number of ICs used in the proposed JIT technique is determined by trial and error. The estimation results are shown in Fig. 3, which shows that the correlation coefficient r is 0.9336 and a root mean square error (RMSE) of 0.0152 with 4 ICs extracted. For comparison, Fig. 4 gives the estimation results for CoJIT, with $\lambda = 0.99, W = 700, d = 40$ and the threshold $\bar{\mathfrak{J}} = 0$ respectively.

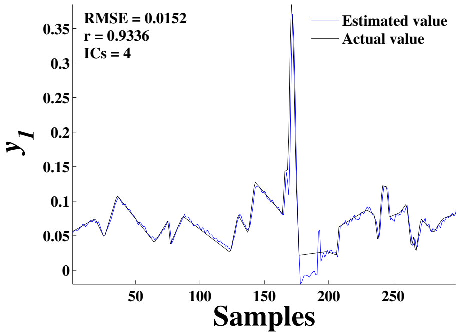

Fig. 3. Prediction result by the proposed JIT for RSU data (y_1)

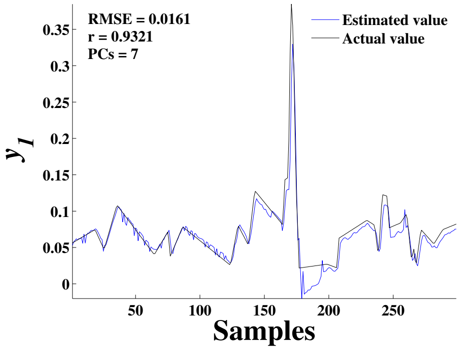

Fig. 4. Prediction result by CoJIT for RSU data (y_1)

From Fig. 3 and Fig. 4 it can be seen that the proposed JIT technique gives better predictions for y_1 than CoJIT with regard to both evaluation criterions, while the ICs retained is 4 with comparison to 7 PCs for CoJIT.

For prediction of y_2, the same parameters are set for both methods, however, with different number of ICs and PCs. The results are shown in Fig. 5 and Fig. 6 respectively. Similar results can be observed from Fig. 5 and Fig. 6.

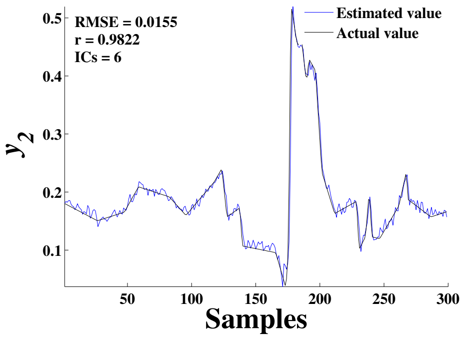

Fig. 5. Prediction result by the proposed JIT for RSU data (y_2)

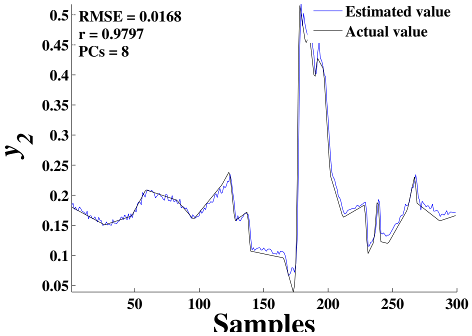

Fig. 6. Prediction result by CoJIT for RSU data (y_2)

For our method, the correlation coefficient r between the estimated value and actual value is 0.9822 and RMSE is 0.0155 with 6 ICs retained. As for CoJIT, it produces a correlation coefficient r of 0.9797 and RMSE value of 0.0168, both shows poorer predictive accuracy than our method. Moreover, to produce the desired result, 8 PCs should be retained for CoJIT. In this regard, our JIT method can produce better predictions with simpler model.

V. Conclusion

This article proposes a novel JIT modeling technique by utilizing non-Gaussian information of the operation data. Non-Gaussian components are extracted from predictor data through the NGR method, which considers both non-Gaussianty of the extracted components and correlation between response variables in the sense of mutual information. SVDD is performed on the extracted ICs by projecting the data into kernel space to calculate the D statistic. The D statistic is then mixed with the T^2 statistic to get a new similarity measure. Based on the new similarity measure, a JIT modeling procedure is proposed and tested on both numerical and application studies. The new similarity is more suitable for non-Gaussian data; hence better predictive accuracy can be obtained.

References

[1] M. Kano, Y. Nakagawa, ”Data-based monitoring, process control and quality improvement: recent developments and applications in steel industry,” *Comput. Chem. Eng.*, vol. 32, 2008, pp 12-24.

[2] V. Radhakrishnan, A. Moham, ”Neural networks for the identification and control of blast furnace hot metal quality,” *J. Process Control*, vol. 10, 2000, pp 509-524.

[3] Z. Q. Ge, F. R. Gao, Z. H. Song, ”Mixture probabilistic PCR model for soft sensing of multimode processes,”*Chemometr. Intell. Lab. Syst.*, vol. 104, 2010, pp 91-105.

[4] P. Facco, F. Doplicher, F. Bezzo, M. Barolo, ”Moving average PLS soft sensor for online product quality estimation in an industrial batch polymerization process,”*J. Process Control*, vol. 19, 2009, pp 520-529.

[5] D. G. Zhou, G. Li, S. J. Qin, ”Total projection to latent structures for process monitoring,”*AIChE J.*, vol. 56, 2009, pp 168-178.

[6] A. S. Kamalabady, K. Salahshoor, ”Affine modeling of nonlinear multivariable process using a new adaptive neural network-based approach,”*J. Process Control*, vol. 19, 2009, pp 380-393.

[7] W. Li, H. H. Yue, S. Valle-Cervantes, S. J. Qin, ”Recursive PCA for adaptive process monitoring,”*J. Process Control*, vol. 10, 2000, pp 471-486.

[8] S. J. Qin, ”Recursive PLS algorithms for adaptive data modeling,” ,*Comput. Chem. Eng.*, vol.22, 1998, pp.503-514.

[9] G. Bontempi, M. Birattari, H. Bersini, ”Lazy learning for local modeling and control,” *Int. J. Control* , vol.72, 1999, pp.643-658.

[10] C. G. Atkeson, A. W. Moore, S. Schaal, ”Locally weighted learning,” *Artif. Intell. Rev.*, vol. 11, 1997, pp. 11-73.

[11] C. Cheng, M. S. Chiu, ”A new data-based methodology for nonlinear process modeling,” *Chem. Eng. Sci.*, vol. 59, 2004, pp. 2801-2810.

[12] C. Cheng, M. S. Chiu, ”Nonlinear process monitoring using JITL-PCA,” *Chemometr. Intell. Lab. Syst*, vol. 76, 2005, pp. 1-13.

[13] Z. Q. Ge, Z. H. Song, ”A comparative study of just-in-time-learning based methods for online soft sensor modeling,” *Chemometr. Intell. Lab. Syst*, vol. 104, 2010, pp. 306-317.

[14] K. Fujiwara, M. Kano, S. Hasebe, A. Takinami, ”Soft-sensor development using correlation-based just-in-time modeling,” *AIChE J.*, vol. 55, 2009, pp. 1754-1765.

[15] J. M. Lee, S. J. Qin, I. B. Lee, ”Fault detection and diagnosis based on modified independent component analysis,” *AIChE J.*, vol. 55, 2008, pp. 3501-3514.

[16] D. M. J. Tax, R. P. W., Duin, ”Support vector data description,” *Pattern Recog. Lettes*, vol. 20, 1999, pp. 1191-1199.

[17] J. S. Zeng, L. Xie, U. Kruger, C. H. Gao, ”A non-Gaussian regression algorithm based on mutual information maximization,” accepted by *Chemometr. Intell. Lab. Syst.*

[18] A. Hyvarinen, ”Fast and robust fixed-point algorithms for independent component analysis,” *IEEE Trans. Neural Networks*, vol. 10, 1999, pp. 626-634.

[19] M. M. Van Hulle, ”Edgeworth approximation of multivariate differential entropy,” *Neural Comput.*, vol. 17, 2005, pp. 1903-1910.

[20] X. Q. Liu, L. Xie, U. Kruger, T. Littler, ”Statistical-based monitoring of multivariate non-Gaussian systems,” *AIChE J.*, vol. 54, 2008, pp. 2379-2391.

[21] Z. Q. Ge, L. Xie, U. Kruger, L. Lamont, Z. H. Song, S. Q. Wang, ”Sensor fault identification and isolation of multivariate non-Gaussian processes,” *J. Process Control*, vol. 19, 2009, pp. 1707-1715.

[22] L. Fortuna, A. Rizzo, M. Sinatra, M. G. Xibilia, ”Soft analyzers for a sulfur recovery unit,” *Control Eng. Pract.*, vol. 11, 2003, pp. 1491-1500.

[23] Z. Q. Ge, Z. H. Song, ”Nonlinear soft sensor development based on relevance vector machine,” *Ind. Eng. Chem. Res.*, vol. 49, 2010, pp. 8685-8693.