Journal of Systems Engineering and Electronics
Vol. 24, No. 3, June 2013, pp.500-511

Fault detection and diagnosis for data incomplete industrial systems with new Bayesian network approach

Zhengdao Zhang*, Jinlin Zhu, and Feng Pan

Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi 214122, China

Abstract: For the fault detection and diagnosis problem in large-scale industrial systems, there are two important issues: the missing data samples and the non-Gaussian property of the data. However, most of the existing data-driven methods cannot be able to handle both of them. Thus, a new Bayesian network classifier based fault detection and diagnosis method is proposed. At first, a non-imputation method is presented to handle the data incomplete samples, with the property of the proposed Bayesian network classifier, and the missing values can be marginalized in an elegant manner. Furthermore, the Gaussian mixture model is used to approximate the non-Gaussian data with a linear combination of finite Gaussian mixtures, so that the Bayesian network can process the non-Gaussian data in an effective way. Therefore, the entire fault detection and diagnosis method can deal with the high-dimensional incomplete process samples in an efficient and robust way. The diagnosis results are expressed in the manner of probability with the reliability scores. The proposed approach is evaluated with a benchmark problem called the Tennessee Eastman process. The simulation results show the effectiveness and robustness of the proposed method in fault detection and diagnosis for large-scale systems with missing measurements.

Keywords: fault detection and diagnosis, Bayesian network, Gaussian mixture model, data incomplete, non-imputation.

DOI: 10.1109/JSEE.2013.00058

1. Introduction

With the development of industrial manufacturing, the industrial processes are becoming more and more complex, the fault in the entire system may easily happen. Thus, the task of fault detection and diagnosis (FDD) is turning more and more significant [1]. The target of fault detection is to decide whether an abnormal event or a fault has happened; once a fault has been detected, the fault diagnosis step is enabled to identify the type or root cause of the fault.

Manuscript received April 5, 2012.

*Corresponding author.

This work was supported by the National Natural Science Foundation of China (61202473), the Fundamental Research Funds for Central Universities (JUSRP111A49), "111 Project" (B12018), and the Priority Academic Program Development of Jiangsu Higher Education Institutions.

There are mainly three kinds of approaches for fault detection and diagnosis: the so-called analytical approaches, the knowledge-based approaches, and the data-driven approaches [2]. The analytical approaches need the detailed mathematical model of the system, and are not applicable with large-scale industrial systems since building those complicate and coupled multivariate models is very time-consuming and expensive [3]. The knowledge-based approaches establish the qualitative models like fault tree and expert system with prior system knowledge to make the detection and diagnosis, and it is analogous to the analytical approaches, the knowledge based approaches need great endeavors for conducting those qualitative models under the cases of large systems. Unlike the above two approaches, the data-driven approaches only rely on the measurements of process [4], and the statistical information extracted from real data can greatly enhance the monitoring quality for large-scale systems [2]. Therefore, the data-driven method has been widely studied.

In practice, many data-driven techniques have been widely and successfully used for FDD. For fault detection, the Shewhart control charts are designed for univariate statistical process control, and the T^2 and Q control charts are designed for multivariate process control. There are also subspace projection techniques like the principle component analysis (PCA) and the independent component analysis (ICA) for reducing the high-dimensional space in multivariate process into a more meaningful and lower-dimensional space. Many other methods have also been developed, such as artificial neural networks (ANN), support vector machine (SVM), and Bayesian network (BN) [5,6]. Among them, the BN is usually constructed as a BN classifier (BNC) when performing classification tasks. The BNC is still a BN with the class node as its root node.

Although the data-driven methods have been widely studied and adopted by researchers, problems still exist (e.g., the incapability of missing data handling). In industrial sampling and transmission systems, sample data may

be unavailable in some time points due to sensor failure, noise, multiple rate sampling scheme or network packet loss etc. [7,8]. Since most data analysis models are designed for well conditioned data sets and cannot use the incomplete data, which results in delays or failures in FDD. The BNC can deal with the missing data issue and avoid performance deterioration to some extent, but a lot of the existing well-behaved missing data techniques for BNC towards handling this issue are mainly based on the data imputation scheme, that is, to convert the incomplete cases into complete ones, and thus, the existing BNC learning and inference methods can directly be used. It should be noted that these processing methods like expectation maximization (EM) [9], data augmentation (DA) [10] are mainly based on the statistical theory, and use the current BNC model and parameter information to impute the missing values in a way so that there is no change in the statistical properties of the data set. Although unbias estimation could be obtained, the efficiency of such techniques largely depends on the missing rate of the original samples, especially when missing values account for large portion of the data set, those methods could be unreliable, in other words, despite the time has been taken up during the preprocessing for the missing values, the filled-in values may be misleading, thus, FDD has also been misled. Therefore, sometimes the imputation based methods are not suitable for those real-time industrial processes. Thus, there exists a great need to explore the non-imputation based method which is to conduct the FDD directly with the BN from the information of the observable part.

Many efficient algorithms have been developed that can perform efficient learning and inference for the BN (classifier) model [11,12]. A distinguishing feature for the model is that with its internal inference mechanism, the BNC can directly (i.e., without the data preparation) deal with the partially observed data instances [13]. However, a major limitation for most of the BNC techniques is that they work efficiently only for categorical and Gaussian variables. Most industrial process measurements have continuous magnitudes which could be non-Gaussian. Discretization can solve this problem, but the cost is a certain amount of information loss. The univariate Gaussian model can also deal with continuous variables and assumes that each variable obeys a specific Gaussian distribution. But, as the system behaves in different operating conditions, the system modal is various, thus, a singular Gaussian distribution does not effectively reflect the significant statistical difference between various conditions and may lead to serious false alarm (false positive) or alarm missing (false negative). There are also various other techniques for this issue, such as Markov chain Monte Carlo (MCMC) methods [14], mixture of truncated exponentials [15], mixture of polynomials [16]. However, these methods have difficulties in exact reasoning and are not readily applicable.

This paper investigates a modified BNC approach for FDD based on missing data with the application to process monitoring. A novel non-imputation method based on marginalization is proposed that can directly and effectively deal with the incomplete data samples from the industrial process. A detailed proof is given on how to deal with incomplete data samples in the proposed framework. In order to deal with the multimode problem, we extend the BNC in a Gaussian mixture model (GMM) framework [17]. Besides, a joint mutual information based feature selection phrase is implemented to reduce the high-dimensional feature spaces and make the constructed classifier more efficient and robust. And more importantly, the joint mutual information criterion is used as the reliability scores to infer the credibility of results when the measurable data are partially missing. Finally, we use the synthetic BNC method to do the detection and diagnosis task in data incomplete conditions.

The rest of this paper is organized as follows. Section 2 gives the problem formation and preliminaries for the further development of FDD, which includes a brief overview of the BNC and the GMM theory. Section 3 proposes the synthesis mechanism between them, and follows the introduction of an effective feature selection criterion. And then the non-imputation missing data handling technique which is demonstrated with a detailed proof is proposed. After that, the entire feature selection procedure and reliability analysis of results under missing data are given. Section 4 presents detailed descriptions of the proposed method for the construction of the entire FDD model. Section 5 demonstrates the application of the proposed techniques to deal with the FDD on a benchmark problem: the Tennessee Eastman process (TE process). Finally, Section 6 gives conclusions and outlooks of this work.

2. Problem formulation and preliminaries

2.1 FDD as a classification task

Consider the general nonlinear multi-input multi-output system given as

$$\boldsymbol{y} = F(\boldsymbol{x_0}, \boldsymbol{u}, \boldsymbol{\beta}, \boldsymbol{f}, \boldsymbol{e})$$

where $\boldsymbol{x_0} \in \mathbf{R}^n$ is the initial state vector, $\boldsymbol{y} \in \mathbf{R}^p$ is the output vector, $\boldsymbol{u} \in \mathbf{R}^q$ is the input vector, and $\boldsymbol{e} \in \mathbf{R}^n$ is the noise. The model function F is unknown and is difficult to be modeled as analytical forms. The fault functions matrix $\boldsymbol{f} := [f_0, f_1, \ldots, f_m]^{\mathrm{T}}$ represents various potential faults, and their fault functions are unknown, among them,

class f_0 represents the normal condition, while other functions in the set indicate various faults in the system. $\boldsymbol{\beta} \in \mathbf{R}^{m+1}$ is a binary indicator vector that indicates the occurrence of certain faults. For example, in normal conditions, the indicator is $[1, \underbrace{0, \ldots, 0}_{\text{all } 0}]$, but if both faults i and j happen, $(i+1)$th and $(j+1)$th elements should be 1, and other elements are 0. In this work, the single fault situation is assumed for the validation of the proposed method.

As we have mentioned above, both fault detection and fault diagnosis can be regarded as a classification task. Assume that both the classifiers for fault detection and fault diagnosis have been constructed and trained respectively, then, both the fault detection task and the diagnosis task can be viewed as deciding the parameter of $\boldsymbol{\beta}$. The main problem then turns to the construction of the classifiers for $\boldsymbol{\beta}$. In this work, we focus on the BNC.

2.2 BNC

A BN is defined by a directed acyclic graph (DAG), together with a local probabilistic model for each node [18]. A formal definition of BN is given in Definition 1. Note that, for simplicity, in the following part, we regard the input variables $\boldsymbol{U} = (U_1, \ldots, U_q)$ and output variables $\boldsymbol{Y} = (Y_1, \ldots, Y_p)$ both as random variables, and use unified variables $\boldsymbol{X} = (X_1, X_2, \ldots, X_n) \in \mathbf{R}^n (n = p+q)$ to represent those random variables, thus, the data sample can be written as $\boldsymbol{x} = (X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)$.

Definition 1 An n-dimensional BN is a triple $B = \langle \boldsymbol{X}, \boldsymbol{E}, \boldsymbol{\Theta} \rangle$, where $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$ is an n-dimensional random variable and each random variable X_i is ranged over a finite domain D_i; $\boldsymbol{E}$ is a set of arcs, which together with $\boldsymbol{X}$ constitutes a DAG $G = \langle \boldsymbol{X}, \boldsymbol{E} \rangle$; $\boldsymbol{\Theta}$ is a set of conditional probability distributions that encodes the network parameters θ_{ijk} $(i = 1, \ldots, n, j \in D_{\Pi\{X_i\}})$. The parameters of each variable in the network are often represented as a conditional probability table (CPT).

As mentioned above, a BNC is a special BN in which the root node represents the class label. For example, as will be used later, the well-known naive BNC (NBNC) [19] has the simplest structure depicted as Fig. 1.

The discrete class label C is often represented as a square node and the continuous attribute nodes as circle nodes. In the NBNC, the continuous attributes are usually assumed as Gaussian nodes, which PDFs can be expressed as Gaussian nodes. The CPT between the class node C (with $m+1$ states $0, 1, \ldots, m$) and the attribute can be represented in Table 1, and $p(X_i|C = m) \sim N(\mu_m, \Sigma_m)$ in column m denotes that the conditional probability density function for X_i conforms to a Gaussian distribution with the mean and covariance expressed as $N(\mu_m, \Sigma_m)$.

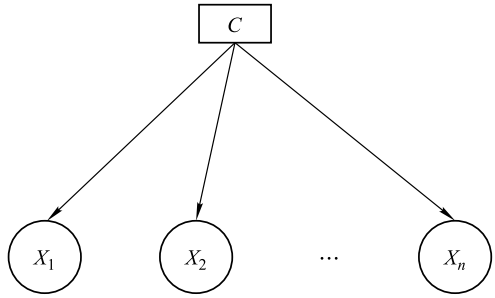


Fig. 1 NBNC

Table 1 CPT

C	$\mathrm{CPT}(X_i)$
0	$p(X_i\|C=0) \sim N(\mu_0, \Sigma_0)$
1	$p(X_i\|C=1) \sim N(\mu_1, \Sigma_1)$
$\vdots$	$\vdots$
m	$p(X_i\|C=m) \sim N(\mu_m, \Sigma_m)$

For the NBNC, the goal is to determine the class label given all attribute values, that is, to compute the posterior probability $p(C|X_1 = x_1, \ldots, X_n = x_n)$ according to the Bayesian rule and the network property:

$$p(C|X_1 = x_1, \ldots, X_n = x_n) \propto p(C) \cdot p(X_1 = x_1, \ldots, X_n = x_n|C) = p(C) \prod_{i=1}^{n} p(X_i = x_i|C). \tag{1}$$

As mentioned above, the target of the classifier is to decide the elements in $\boldsymbol{\beta}$, thus, the state number of the class label is determined by the element number in $\boldsymbol{\beta}$. In the case of fault detection, as the task is only to decide the first binary element of $\boldsymbol{\beta}$, thus the class label is also a binary variable with two states, while in fault diagnosis, assume that there are m types of faults, then the class label should be an m-category variable.

2.3 GMM

A GMM is used to deal with the multimode density with a mixture of Gaussian distributions. First, we give the definition of the GMM.

Definition 2 For a given random variable vector $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$, the standard GMM is defined as

$$p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_{k=1}^{K} w_k N(\boldsymbol{X}|\boldsymbol{\mu_k}, \boldsymbol{\Sigma}_k) \tag{2}$$

where $\boldsymbol{\theta} = \{w_k, \boldsymbol{\mu_k}, \boldsymbol{\Sigma}_k\}(k = 1, \ldots, K)$ is the set of parameters. w_k is called the mixing weights that satisfy $0 \leqslant w_k \leqslant 1$ and $\sum\limits_k w_k = 1$. K is the number of mixing components, $\boldsymbol{\mu_k}$ and $\boldsymbol{\Sigma}_k$ are the mean vector and the covariance matrix of the corresponding component k.

Given a training set and the GMM configuration, the parameters θ is usually estimated by the maximum likelihood method which is called EM algorithm. In order to save space, we omit the detailed descriptions, for more fundamental information, readers can refer the relative tutorials.

It is important to note that the covariance matrix should be full rank or constrained to be diagonal. As the different Gaussian components are acting together to approximate the feature space density, the mixture of diagonal covariance can also catch those relationships between features [20], so we choose the diagonal form. It can be seen later that it is also the important property which makes it possible to deal with missing values with a non-imputation method.

3. The proposed method

3.1 The proposed classifier

As we have discussed above, a main problem for BN is its limitation for the representation of the non-Gaussian density. In this section, a new BNC extended with the GMM framework is derived. Specifically, the following theorem characterizes the relationship between the GMM and the BNC.

Theorem 1 A GMM is a special case of the BNC.

Proof Consider the BNC depicted in Fig. 2. The class label C is the root node, the class number denotes the number of system's working conditions, node W denotes the component of the mixture model $1, \ldots, K$, and the components of the mixture $w_1^m, \ldots, w_k^m$ are determined by the CPT of W depended on $C = m$. Table 2 represents the CPT of W. The child node X represents the features $X_1, \ldots, X_n$, and is assumed as a multidimensional Gaussian node. In a formal way, given its parents' values $C = m$ and $W = k$, the conditional distribution of $\boldsymbol{X}$ is written as

$$p(\boldsymbol{X}|C = m, W = k) \sim N(\boldsymbol{X}|\boldsymbol{\mu}_k^m, \boldsymbol{\Sigma}_k^m). \quad (3)$$

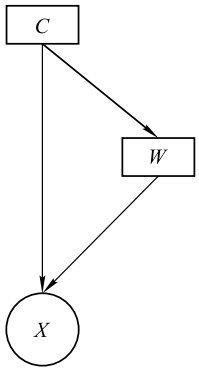


Fig. 2 BNC for GMM

Table 2 CPT for component node W

	W			
C	1	2	...	K
0	w_1^0	w_2^0	...	w_k^0
1	w_1^1	w_2^1	...	w_k^1
⋮	⋮	⋮	⋱	⋮
m	w_1^m	w_2^m	...	w_k^m

From the BNC model, given a set of continuous feature values $\boldsymbol{x} = \{X_1 = x_1, \ldots, X_n = x_n\}$, we can infer the likelihood of the class node as follows:

$$p(\boldsymbol{x}|C = m) = \sum_{k=1}^{K} p(\boldsymbol{x}|C = m, W = k) \cdot$$

$$p(W = k|C = m) = \sum_{k=1}^{K} w_k^m p(\boldsymbol{x}|C = m, W = k). \quad (4)$$

Using (3), (4) can be written as

$$p(\boldsymbol{x}|C = m) = \sum_{k=1}^{K} w_k^m N(\boldsymbol{x}|\boldsymbol{\mu}_k^m, \boldsymbol{\Sigma}_k^m). \quad (5)$$

On the basis of (2), (5) shows that for each working condition of the system, we use a separate GMM to approximate the different multimode density, i.e., if the system parameter's statistical property can be approximated by GMM, then it can also be approximated with BNC. □

Note that when the system behaves in different working conditions, the statistical information or probability density of the states is various. Thus, the proposed method can be adapted to the wide ranges of multimode detection and diagnosis problems.

However, due to the overfitting, the performance of classifiers may deteriorate with all variables. In FDD, different fault conditions may be implicated with different potential variables, so one needs a multivariate feature selection to find the most relevant and high-impact features for different faults.

Feature selection is to define a statistical criterion, then we use this criterion to rank all the features. The high-rank features are expected to be useful for the classification task. Generally speaking, a good set of features should be individually independent as well as not highly correlated [21]. In this work, we directly use the method proposed by Yang et al. called joint mutual information (JMI) [22]:

$$J_{\mathrm{JMI}} = \sum_{k=1}^{n-1} I(X_n X_k; C). \quad (6)$$

And this method computes the candidate X_n with all other current features, which can also be written as [21]

$$J_{\mathrm{JMI}} = I(X_n; C) - \sum_{k=1}^{n-1} [I(X_n; X_k) - I(X_n; X_k|C)] \quad (7)$$

where $I(X_n;C)$ is the relevance term, $I(X_n;X_k)$ is the correlation between two features, and $I(X_n;X_k|C)$ is the conditional correlation. It can be seen from (7) that the JMI method makes a tradeoff between relevance and redundancy. Note that the JMI here is only a criterion, the entire feature selection procedure with this criterion, however, will be elaborated and discussed later in Section 3.3 together with the reliability analysis.

3.2 Missing data handling approach

In industrial processes, many data samples may be incomplete. A lot of missing data handling approaches are inclined to impute the missing values with statistical methods to obtain a complete set, perhaps the most commonly used approach is the EM method [23]. However, the effectiveness of those statistical imputation methods depends partially on the missing data mechanism, and the efficiency of those methods may largely depend on the missing rate of the incomplete set. For example, the EM method assumes the missing at random (MAR) mechanism [24]. Although, in many cases, the MAR assumption could meet the occasions and the imputed values do not distort the statistical property of the collected information and also make sensible imputations. However, the assumption itself is a limitation which implies that it may behave poor in some other missing mechanisms, for example, MAR methods may not be so effective under the cases of not MAR (NMAR). What's more, in industrial processes, as the complexity of working conditions, a singular missing data mechanism may not suit for all cases, and the demand of real-time requires us to find an effective method to deal with incomplete cases and make the in-time decision. Based on the above feature selection techniques, and the marginalization idea proposed in [25], we present a new non-imputation method.

As mentioned above, the feature selection approach to choose those high-impact features related to current condition is used. In actual classification problems, the missing data by low-relevant features will have no impact if our selected features are all available. The problem to be considered now is when selected features have missing values, how can we do the classification task.

Given an incomplete feature sample $\boldsymbol{x'}$, first partition the $\boldsymbol{x'}$ as the available part x_o and the unavailable part $\boldsymbol{x_u}$, that is $\boldsymbol{x'}=(\boldsymbol{x_o},\boldsymbol{x_u})$. Now, the likelihood function for the observable part can be written as

$$p(\boldsymbol{x_o}|C=m)=\int p(\boldsymbol{x_o},\boldsymbol{x_u}|C=m)\,\mathrm{d}\boldsymbol{x_u}. \tag{8}$$

Meanwhile, (4) can be adapted and rewritten as

$$p(\boldsymbol{x'}|C=m)=\sum_{k=1}^{K}p(\boldsymbol{x'}|C=m,W=k)\cdot$$

$$p(W=k|C=m)=$$

$$\sum_{k=1}^{K}w_k^m p(\boldsymbol{x_o},\boldsymbol{x_u}|C=m,W=k). \tag{9}$$

Since the features within the diagonal Gaussian in the mixture are independent [25], thus we obtain

$$p(\boldsymbol{x'}|C=m)=\sum_{k=1}^{K}w_k^m p(\boldsymbol{x_o}|C=m,W=k)\cdot$$

$$p(\boldsymbol{x_u}|C=m,W=k). \tag{10}$$

Therefore, (8) can be marginalized as

$$p(\boldsymbol{x_o}|C=m)=\int p(\boldsymbol{x_o},\boldsymbol{x_u}|C=m)\,\mathrm{d}\boldsymbol{x_u}=$$

$$\int\sum_{k=1}^{K}w_k^m p(\boldsymbol{x_o}|C=m,W=k)p(\boldsymbol{x_u}|C=m,W=k)=$$

$$\sum_{k=1}^{K}w_k^m p(\boldsymbol{x_o}|C=m,W=k)\cdot$$

$$\int p(\boldsymbol{x_u}|C=m,W=k)\mathrm{d}\boldsymbol{x_u}=$$

$$\sum_{k=1}^{K}w_k^m p(\boldsymbol{x_o}|C=m,W=k) \tag{11}$$

where $\int p(\boldsymbol{x_u}|C=m,W=k)\mathrm{d}\boldsymbol{x_u}=1$, thus, the conditional likelihood function is only related with the observable part.

According to (11), it can be seen that the missing part has been marginalized out. Next, we need to consider how to compute the observable part likelihood, that is, the probability $p(\boldsymbol{x_o}|C=m,W=k)$. Note that only the constrained full-rank diagonal covariance matrix is chosen as the covariance matrix for GMM, and the observable part of the diagonal matrix is also a full-rank diagonal matrix which is readily computable, thus, the distribution of observable likelihood $p(\boldsymbol{x_o}|C=m,W=k)$ is also a diagonal Gaussian and is easily determined. It can be easily seen that compared with the EM method, the proposed method does not require an increase in pre-processing of missing values, thus it ensures the real-time requirement in FDD.

It should also be noted that there also exists an extreme condition that all the selected features could be unavailable at some time point. In this case, we can do nothing but to shift the current features with the values of the last time, which is also called the nearest neighborhood shift. This makes sense, as 'better last then never'.

3.3 Feature selection and reliability analysis

Although the missing part has been marginalized out, it does not mean that the remaining observed part can always

deal with the FDD task in an equally elegant manner. The high-impact features may play an important role during the classification, therefore, different missing patterns in incomplete samples could make a big difference. For example, in the proposed method, high-impact features may be necessary to a certain fault and the missing of these features may lead to unreliable results, which directly affect the credibility and robustness of the method. In order to measure the reliability of the results from various missing patterns, a uniform criterion to quantify the reliability is needed. First, the feature selection procedure is given.

The feature selection procedure can be conducted in two separate ways. The first approach is to directly score all the features with (7) only by once, and then one picks out a set of high rank features for the model. This approach can be useful when all the features are always available. In this paper, we use another approach: choose iteratively and then pick out. Assume that select $n_{\max}$ features from original features $X = \{X_1, X_2, ..., X_n\}$, and the selected feature set is represented as S. In iteration i, the algorithm picks out the best ranked feature X^i_{best} from X, sets $X = X - \{X^i_{\text{best}}\}$ and adds X^i_{best} to the selected feature set $S = S + \{X^i_{\text{best}}\}$. The iteration step repeats $n_{\max}$ times to choose the desirable amount of features. Note that during the feature selection procedure, the features picked out during the former iteration in feature set S can be viewed as 'missing' in the next iteration, that is, the algorithm chooses the optimal features from the remaining feature set X, which is quite different from the first approach. In missing cases, when some informative features are unavailable, the best thing we can do is to select the most informative features from the remaining features, thus compared with the first approach, the second one is more reasonable under missing data cases.

In the above feature selection phrase, those high-impact features are selected according to their relatively high scores within JMI, thus, the scores can be used as an conclusive index for reliability analysis. Since it can be inferred from (7) that these scores mainly indicate the information contributions of the corresponding features during the classification, the features with high scores are supposed to contribute more information to FDD. Notice that due to the interaction of variables in each iteration differs, scores for selected variables generally change during the course of the algorithm. We need to normalize the best scores within each iteration step. Let $SC_{X^i_k}$ represent the scores of each variables in the ith iteration, among them, the X^i_{best} represents the variable with the best score, then the normalization can be computed as

$$W_{X^i_{\text{best}}} = SC_{X^i_{\text{best}}} / \sum_{k=1}^{j} SC_{X^i_k} \tag{12}$$

where $W_{X^i_{\text{best}}}$ denotes the normalized result, the result can be considered as the specific weight of the information contribution from X^i_{best}. It is noted that since the variables are passed down from the last $i-1$ iterations, thus, the weight from the current iteration should multiply by the corresponding remaining weights (information contribution rate excludes the best variable in each iteration) of the former $i-1$ iterations:

$$W_{X^i_{\text{best}}} = W_{X^i_{\text{best}}} \prod_{m=1}^{i-1} (1 - W_{X^m_{\text{best}}}). \tag{13}$$

Next, the scores of the selected features should be normalized again to form a 100% information, since when we use the selected feature to conduct the classification task, we usually assume that these features make up the complete information:

$$\bar{W}_{X^i_{\text{best}}} = W_{X^i_{\text{best}}} / \sum_{k=1}^{n_{\max}} W_{X^i_{\text{best}}}. \tag{14}$$

The normalized scores are called the reliability scores. The reliability scores act as the auxiliary information for FDD. In the FDD course, each result from incomplete samples is combined with a reliability score according to the missing pattern of the sample, which provides the operator a more complete and credible result before taking the relevant measures. The following algorithm describes the detailed procedure for feature selection and calculation of reliability scores.

Algorithm 1 Feature selection and reliability scores calculation

Input Feature set $X = \{X_1, X_2, \ldots, X_n\}$.

Output Selected feature set $S = \{X^1_{\text{best}}, \ldots, X^{n_{\max}}_{\text{best}}\}$ and corresponding reliablity scores $\{\bar{W}_{X^1_{\text{best}}}, \ldots, \bar{W}_{X^{n_{\max}}_{\text{best}}}\}$ that has been normalized and satisfies $\sum_{k=1}^{n_{\max}} \bar{W}_{X^k_{\text{best}}} = 1$.

Initialization Selected set $S = \varnothing$.

1: for $i = 1$ to $n_{\max}$
2: $\quad j = n - i$;
3: $\quad$ for $k = 1$ to j
4: $\quad\quad SC_{X^i_k} = J_{\text{JMI}}\{X^i_k\}$
5: $\quad$ endfor
6: $\quad SC^i_{\sum} = \sum_{k=1}^{j} SC_{X^i_k}$;
7: $\quad W_{X^i_k} = SC_{X^i_k} / SC^i_{\sum}$;
8: $\quad X^i_{\text{best}} = \arg\max_{X^i_k} \{W_{X^1_k}, \ldots, W_{X^j_k}\}$;
9: $\quad S = S + \{X^i_{\text{best}}\}$;
10: $\quad X = X - \{X^i_{\text{best}}\}$;
11: $\quad$ if $i \neq 1$
12: $\quad W_{X^i_{\text{best}}} = W_{X^i_{\text{best}}} \prod_{m=1}^{i-1} (1 - W_{X^m_{\text{best}}})$;

13: endif
14: endfor
15: for $i = 1$ to $n_{\max}$
16: $\bar{W}_{X^i_{\text{best}}} = W_{X^i_{\text{best}}} / \sum_{k=1}^{n_{\max}} W_{X^i_{\text{best}}}$;
17: endfor

4. Construction of the entire FDD model

For fault detection and fault diagnosis, we consider two different classification tasks and construct the different classifiers. Firstly, the detection classifier should be constructed. A feature selection phrase is pre-processed for the two-category problem. After that, the detection classifier is constructed with a BNC extended with GMM as depicted in Fig. 2 with a binary root node for normal and abnormal conditions separately. Note that the question is how many features should be selected for the child node of the BNC. On the one hand, in order to ensure that enough redundant information among features is provided in the cases of incomplete feature samples, there should be as many features as possible for the classifier. On the other hand, in order to prevent overfitting, the selected features cannot be too many. Therefore, a trade-off should be made during this phrase. In order to ensure the performance of the proposed classifier, the simulation iteratively adds each high-rank feature to test the most appropriate feature number, also during the training procedure, an m-fold cross validation technique is used to evaluate the performance of th proposed classifier. In the validation, the training set is randomly divided into m subsets. At each run, one of the m subsets is chosen as a test set, while the other $m-1$ subsets are put together as the training set. Then, the average accuracy across all m trials is computed as the performance of the classifier. The model with a higher accuracy can be considered as the model with the most appropriate features. Therefore, the classifier for detection has been obtained.

The next step is to design the classifier for diagnosis. In fault diagnosis, there are more than two classes of faults. Theoretically, a single BNC with a multi-class root node is enough, in that case, although there are m classes of faults, only a certain amount of features are selected for all of those faults. But this conclusion may not be exactly right in practices. For example, in this work, a total of five features may not completely separate those m faults. Since each fault may have different implicated features, it would be better that each fault has a separate classification surface with those other $m-1$ faults, thus, for each of those m faults, a separate BNC extended with GMM is constructed with its own selected features. For example, for fault i, a separate BNC—F_i is designed, the root node F_i of each BNC is a binary indicator which has two states: State 1 (fault i negative) and State 2 (fault i positive), W_i is the component node with the CPT of mixture weight, and the features for child node $\boldsymbol{X}_i$ is selected from the corresponding two-class data set (class of data for fault i negative and class of data for fault i positive). In each sub-BNC, information is collected by the sub-root node F and then 'integrated' and analyzed by the root node DA. The entire model for fault diagnosis is depicted as Fig. 3.

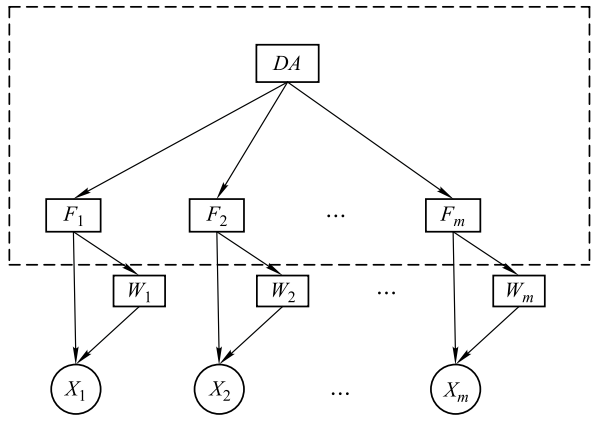


Fig. 3 BNC for fault diagnosis

From Fig. 3, it can also be seen that the root DA and all sub-roots $F_1, \ldots, F_m$ form an NBNC which is represented in a dash box. In fact, the whole diagnosis model can be viewed as an NBNC with each of its child node extended by a subsidiary BNC for each fault. Once the diagnosis model is established, the training set will be used to learn the parameters of the integrated BNC, therefore, the classifier construction for diagnosis has also been accomplished. Then, the detection model and diagnosis model are all ready for the application.

5. Application to the TE process

5.1 Presentation of TE process and data set

The TE process is a chemical process. This chemical process is an open-loop unstable plant-wide process control problem considered as a benchmark simulation for various process monitoring techniques and also a challenging problem for FDD methods [26]. The TE process is composed of five major transformation units: a reactor, a condenser, a compressor, a stripper and a separator. Four gaseous reactants A, C, D, E and an inert B are fed to the reactor and form the liquid products F, G, and H. There are 11 input variables (without agitator speed) and 41 output variables in the process. The entire flow diagram is shown as Fig. 4.

In the TE process, there are 20 types of identified faults. Among them, faults 4, 9 and 11 (see Table 3) are especially focused by authors in [27,28]. These three types

of faults are good representations of overlapping data and are difficult to classify. Thus, this work will also focus on these three faults, and do the FDD task with the proposed classifier.

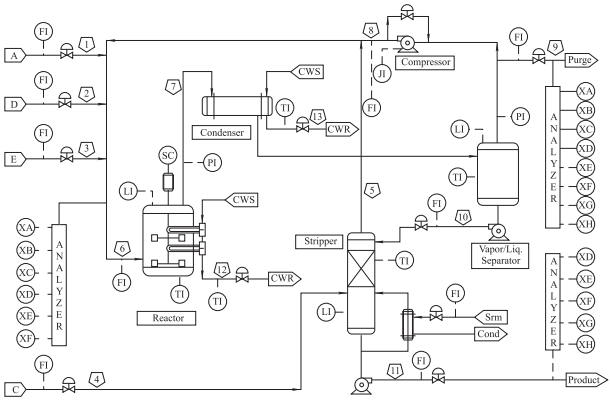


Fig. 4 Flow diagram of TE process

Table 3 Description for three types of faults and their data sets

Class	Fault description	Training set	Testing set
Fault 4	Step change in the reactor cooling water inlet temperature	480	800
Fault 9	Random variation in D feed temperature	480	800
Fault 11	Random variation in the reactor cooling water inlet temperature	480	800

As described in Table 3, there are two data sets for each type of faults: a training set and a testing set, with 480 samples and 800 samples respectively, with each sample a 52-dimensional vector (Some process variables used in this article are listed in Table 4 [29]). In fault detection phrase, there is another type of data set: the normal samples. Like the data sets in other three faults, the normal class also has a training set with 480 samples and a testing set with 800 samples. In fault diagnosis, the classification task should be a three-class problem with the goal of separating each of the three faults from the other faults. All of our simulations are accomplished on Matlab with the BN toolbox developed by Murphy [30].

Table 4 Some process variables of the TE process

Variable	Description
25	Component C(Stream 6)
28	Component F(Stream 6)
32	Component D(Stream 9)
34	Component E(Stream 9)
36	Component H(Stream 9)
37	Component D(Stream 11)
38	Component E(Stream 11)
39	Component F(Stream 11)
40	Component G(Stream 11)
41	Component H(Stream 11)
50	Stripper steam valve
51	Reactor cooling water flow

5.2 Fault detection with BNC

To begin with the detection, a feature selection is first conducted on the training set. The normal data are labeled with class 1 and all of those three faults are labeled with class 2. The feature selection phrase is called to rank features. The BNC is then constructed with training set as depicted in Fig. 2. After the training phrase, the testing set is used to test the classifier. The mixing component number is set to 2. For comparison, training results are given with increasingly features in the model, a 10-fold cross validation is used to evaluate the classifier. Fig. 5 shows the different performances of BNC with different feature sets, the Y-axis represents the accurate rate (1=100%) with the corresponding feature set, from which one can see that 5 features are appropriate for the classifier. Table 5 gives the corresponding reliability scores for the 5 best features.

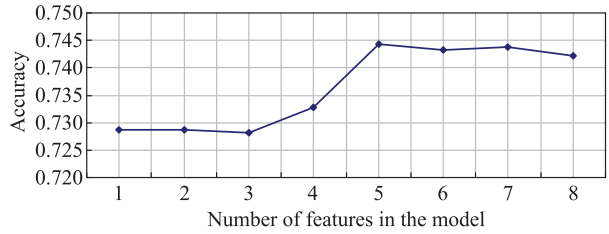


Fig. 5 BNC cross validation result on training set with different feature set

Table 5 Reliability scores for features selected

Variable in the model	Reliability score/%
51	45.99
40	13.54
41	13.51
39	13.50
38	13.46

It can be summarized from Table 5 that variable 51 contributes the most information for detection and the missing of 51 may deteriorate the reliability from 100% into 54.01%. After the training course, we use the obtained BNC to conduct the testing phrase. The test results for faults 4, 9, and 11 are 100%, 28%, and 86.62%, respectively. Results show that fault 9 is hardly to be detected. In fact, fault 9 samples are overlapping with the normal class in the feature space and are hardly to be detected. In [27], authors obtained the same conclusion for fault 9 with a different technique. Fault 4 samples are completely separated with the normal class, therefore, are totally classifiable. While fault 11 samples are partially overlapping with the normal class, thus, as is reflected in results, are not totally classifiable.

Now, we will evaluate the BNC detecting performance by increasing the random missing rate of the test set step by step. In order to simulate the missing data points in real

situations, we use the Bernoulli distributed model in [31], in which 0 stands for the missing of measurements and 1 denotes the complete data. Unlike them, in this paper, if a data point is considered to be missing, then the data record is set to be blank rather than zero. Table 6 represents the relationship between data missing rate and the detecting performance of the classifier. The non-imputation method proposed in this paper is used to deal with the missing values. Due to the uncertainty of performance caused by random missing, each result in the table is given by an average over 5 Monte Carlo simulations. Moreover, as an illustration for reliability analysis, we randomly extract 50 continuous samples and the corresponding reliability scores from one of the 10% missing simulations which form the Fig. 6. Subplots 1–5 shows the 5 important features separately, the incomplete values are represented with red stars. Subplot 6 gives the corresponding reliability scores of the results based on these missing samples.

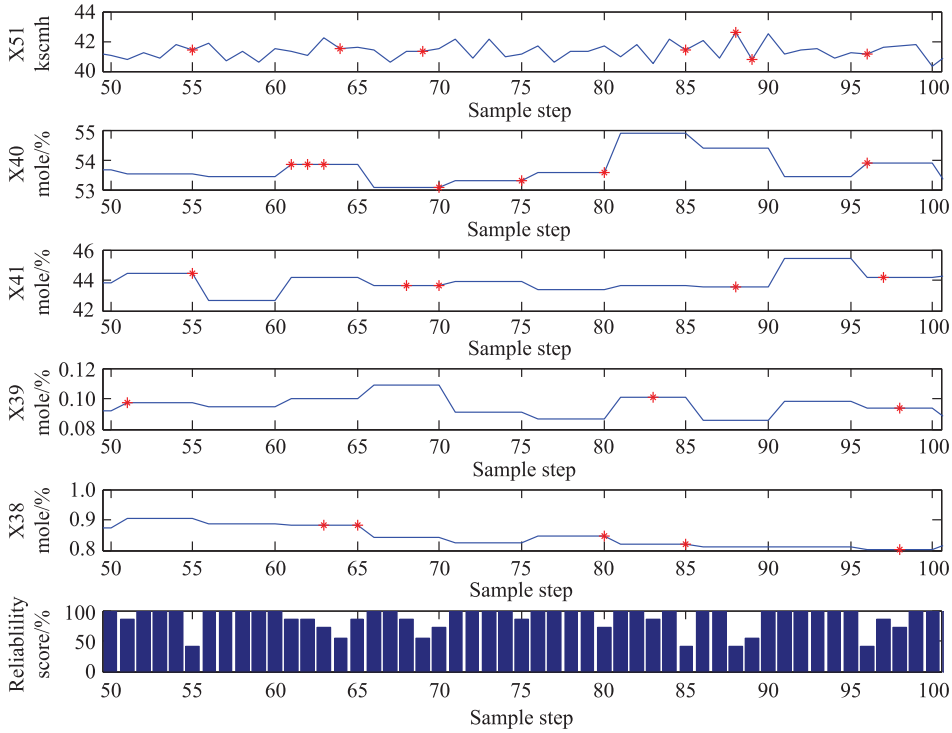


Fig. 6 Features missing 10% at random and the corresponding results reliability

Table 6 Detecting performance of BNC with different missing rate %

Missing rate	0	2	4	6	8	10
Fault 4	100	100	100	100	100	100
Fault 9	28.00	29.57	30.95	32.05	34.15	34.75
Fault 11	86.62	86.85	87.08	87.48	88.01	88.25
Average	71.54	72.14	72.68	73.18	74.05	74.33

It can be inferred from Table 6 that the proposed classifier's overall performance is not highly impacted by the random missing data: the detection average is increased by 2.79% (from 71.54% to 74.33%) within 10% missing rate. It seems that the performance is becoming much better with more missing data, but we can see from the following table as well as explanation that it is not the truth. Anyway, the relatively small performance change within large amount of missing data means that our BNC combined with the non-imputation method is very effective. As the missing rate increases from 0% to 10%, the performance on fault 4 is unchanged and keeps the 100% of detection. However, there are some differences for those overlapping faults: fault 9 and fault 11. We have mentioned above that both of the faults 9 and 11 overlap with the normal case, especially for fault 9: the fault 9 and the normal samples are almost centralized on the same region in the feature space. Table 7 contains two confusion matrixes extracted from random Monte Carlo simulations. This

Table 7 Confusion matrix for comparison between no missing and 10% missing rates

Confusion matrix							
No missing				10% missing			
	Normal	Fault	Total		Normal	Fault	Total
Normal	599	201	800	Normal	546	254	800
Fault 4	0	800	800	Fault 4	0	800	800
Fault 9	576	224	800	Fault 9	527	273	800
Fault 11	107	693	800	Fault 11	95	705	800

table clearly illustrates the difference between the no missing case and the 10% missing case and explains why the detection performance is better with the increased missing rate.

In the no missing case, fault 9 has a very low detection rate and most of the samples are misclassified into the normal class (false negative); as a similar case but not so seriously, fault 11 are partially overlapped with normal case, therefore, about 13.38% are misclassified (false negative). The normal samples are blended with both faults, and 25.13% of which are misclassified into the fault class (false alarm). As the missing rate increases, more normal data are misclassified (false alarm increased), at the meantime, more fault 9 and fault 11 are classified into the fault class (right classification increased), therefore, the detection rate for both fault 9 and fault 11 are increased, which can be seen in Table 6.

5.3 Fault diagnosis with BNC

Once a fault is detected, we need to diagnosis the belonging class of the fault. Like detection, the features are first selected for the BNC model. As mentioned in Section 4, each sub-BNC will select the feature set of its own. In this work, each sub-BNC has five features like the case in detection. The model is constructed with a 3-fault case as described in Fig. 3 and the entire BNC model has three sub-BNCs for fault 4, fault 9 and fault 11 respectively. The mixing components of GMM are also set to 2. Use the same training set as in the detection task to learn the parameters of BNC, and then use a testing set described in Table 3 with different missing rates to testify our model and the non-imputation technique. The JMI selected features and the corresponding reliability scores for each sub-BNC are given in Table 8; the performance of our diagnosis classifier is given by Table 9. Like the results in the detection table, each result in the diagnosis table is also given by an average over 5 Monte Carlo simulations.

Table 8 Selected features for each sub-BNC with JMI and reliable scores

Fault	Feature	Reliability score/%
Fault 4	51,38,37,39,28	62.48,9.43,9.40,9.36,9.33
Fault 9	51,36,25,34,32	62.31,9.53,9.42,9.38,9.36
Fault 11	51,50,40,37,39	52.09,12.02,11.97,11.96,11.96

Table 9 Diagnosis performance of BNC with different missing rates %

Missing rate	0	2	4	6	8	10
Fault 4	99.13	97.63	96.13	95.33	93.60	92.90
Fault 9	96.50	95.05	93.33	92.60	90.20	89.02
Fault 11	60.25	60.18	59.95	59.33	59.13	59.08
Average	85.29	84.29	83.13	82.42	80.98	80.33

It can be inferred from Table 8 that feature 51 plays an important role for each sub-BNC and the lose of 51 may cause unreliable results for the entire model. In Table 9, as the missing rate increases from 0% to 10%, the overall diagnosis average is only decreased by 4.96% (from 85.29% to 80.33%), the results testify the effectiveness of our Bayesian network technique as well as the non-imputation method. With the non-imputation method, one can make a relatively reliable decision without having to do the time-consuming preprocessing works such as data repairing, thus the requirement of real-time for fault detection and diagnosis is fulfilled.

As a further comparison, we also contrast our diagnosis performance with those of the other methods. Since those methods cannot deal with incomplete samples, the comparison is made under complete data cases, and both results are conducted from the same data cases as described in Table 3. The comparison result of different misclassification rates is given by Table 10. Among them, the fisher discriminant analysis (FDA), support vector machine (SVM), and proximal support vector machine (PSVM) are extracted from [27], the conditional Gaussian network (CGN) is extracted from [28]. From Table 10, we can infer that our BNC outperforms all of the other methods. In fact, integrated with the idea of GMM, our Bayesian network classifier can deal with multimode processes more precisely. What's more, those other methods in the table cannot deal with incomplete testing sets, or cannot behave so robust within missing values. On the contrary, our method shows more robustness considering either complete cases or incomplete ones, which is an important improvement comparing with those of others.

Table 10 Misclassification rate of different method (complete data cases) %

Method	FDA	SVM	PSVM	CGN	BNC
Misclassification	38	44	35	18.87	**14.71**

6. Conclusions

This work presents a novel BNC based on the GMM theory and a new non-imputation method which can deal with the FDD problems for those high-dimensional data incomplete industrial processes in an efficient way. The main contributions of this work include the following two aspects.

On the one hand, the multimode probability density can be effectively approximated by the GMM method; moreover, the BNC can effectively do the probability inference with uncertainties. In this work, we combine the both advantages of the two methods together and give a detailed explanation on how the BNC can be extended with the GMM theory. Thus, the standard BNC with Gaus-

sian nodes can successfully manage the multimode non-Gaussian data.

On the other hand, the high-dimensional incomplete data are reasonably processed by the JMI based feature selection and the marginalization-based non-imputation method for missing data handling. The feature selection method is reasonable and efficient, while the marginalized method is proved to be able to obtain the likelihood probability with the partially observable relevant features effectively. Thus, the proposed method makes a detour which avoids the imputation step which might be a time-consuming as well as assumption-limited. Moreover, in order to figure out the reliability of those results which are derived from incomplete samples, we develop the reliability analysis scheme, thus the results are more credible.

The simulation results for TE process detection and diagnosis reveal the robustness of our method within a 10% missing data rate. The detection performance of the classifier is not very ideal partially due to the feature selection which does not completely separate those overlapping cases, so one of the further works is to design an observer for the fault detection task. With the observer, the residuals can be obtained which contain the information of the difference between the measurement of a variable and its theoretical value of the model, thus, the residuals can be taken as inputs to our BNC for abnormal detection. Another outlook of our work is to extend our method from the static BN into the dynamic BN (DBN) framework. The DBN combines the static probabilistic model with the time information to form a new statistical probabilistic model that can deal with time series data samples. Most of the process monitoring problems are dealing with the time series data, thus the DBN may fulfill the fault detection and diagnosis in a better way.

References

[1] V. Venkatsubramanian, R. Rengaswamy, K. Yin, et al. A review of process fault detection and diagnosis–Part I: quantitative model-based methods. *Computers & Chemical Engineering*, 2003, 27 (3): 293–311.

[2] L. H. Chiang, E. L. Russell, R. D. Braatz. *Fault detection and diagnosis in industrial systems*. London: Springer Verlag, 2001.

[3] S. Verron, J. Li, T. Tiplica. Fault detection and isolation of faults in a multivariate process with Bayesian network. *Journal of Process Control*, 2010, 20 (8): 902–911.

[4] S. Verron, T. Tiplica, A. Kobi. Procedure based on mutual information and bayesian networks for the fault diagnosis of industrial systems. *Proc. of the American Control Conference*, 2007: 420–425.

[5] Z. Tang, X. Gao. Study of testability measurement method for equipment based on Bayesian network model. *Journal of Systems Engineering and Electronics*, 2009, 20 (5): 1017–1023.

[6] N. Khakzada, F. Khana, P. Amyotteb. Safety analysis in process facilities: comparison of fault tree and Bayesian network approaches. *Reliability Engineering & System Safety*, 2011, 95 (8): 925–932.

[7] S. Khatibisepehr, B. Huang. Dealing with irregular data in soft sensors: Bayesian method and comparative study. *Industrial & Engineering Chemistry Research*, 2008, 47 (22): 8713–8723.

[8] F. Qi, B. Huang, E. C. Tamayo. A Bayesian approach for control loop diagnosis with missing data. *American Institute of Chemical Engineers Journal*, 2010, 56 (1): 179–195.

[9] T. Denoeux. Maximum likelihood estimation from fuzzy data using the EM algorithm. *Fuzzy Sets and Systems*, 2011, 183 (1): 72–91.

[10] C. Riggelsen. Learning parameters of Bayesian networks from incomplete data via importance sampling. *International Journal of Approximate Reasoning*, 2006, 42 (1/2): 69–83.

[11] M. G. Madden. On the classification performance of TAN and general Bayesian networks. *Knowledge-Based Systems*, 2009, 22(7): 489–495.

[12] K. P. Murphy. Dynamic Bayesian networks: representation, inference and learning. California: UC Berkeley, 2002.

[13] R. Daly, Q. Shen, S. Aitken. Learning Bayesian networks: approaches and issues. *The Knowledge Engineering Review*, 2011, 26 (2): 99–157.

[14] H. Langseth, T. D. Nielsen, R. Rumi, et al. Inference in hybrid Bayesian networks. *Reliability Engineering & System Safety*, 2009, 94 (10): 1499–1509.

[15] B. R. Cobb, P. P. Shenoy. Inference in hybrid Bayesian networks with mixtures of truncated exponentials. *International Journal of Approximate Reasoning*, 2006, 41 (3): 257–286.

[16] P. P. Shenoy, J. C. West. Inference in hybrid Bayesian networks using mixtures of polynomials. *International Journal of Approximate Reasoning*, 2011, 52 (5): 641–657.

[17] M. A. T. Figueiredo, A. K. Jain. Unsupervised learning of finite mixture models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2002, 24 (3): 381–396.

[18] O. Pourret, P. Naim, B. Marcot. *Bayesian Networks: a practical guide to applications*. Chichester: John Wiley & Sons, 2008.

[19] Y. S. Jing, V. Pavlovic, J. M. Rehg. Boosted Bayesian network classifiers. *Machine Learning*, 2008, 73 (2): 155–184.

[20] D. A. Reynolds, T. F. Quatieri, R. B. Dunn. Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing*, 2000, 10 (1/3): 19–41.

[21] G. Brown. A new perspective for information theoretic feature selection. *Proc. of the 12th International Conference on Artificial Intelligence and Statistics*, 2009: 49–56.

[22] H. H. Yang, S. V. Vuuren, S. Sharma, et al. Relevance of time frequency features for phonetic and speaker-channel classification. *Speech Communication*, 2000, 31 (1): 35–50.

[23] S. A. Imtiaz, S. L. Shah. Treatment of missing values in process data analysis. *Canadian Journal of Chemical Engineering*, 2008, 86 (5): 838–858.

[24] J. L. Schafer, J. W. Graham. Missing data: our view of the state of the art. *Psychological Methods*, 2002, 7 (2): 147–177.

[25] L. Josifovski, M. Cooke, P. Green, et al. State based imputation of missing data for robust speech recognition and speech enhancement. *Proc. of the Eurospeech*, 1999: 2837–2840.

[26] Z. Q. Ge, Z. H. Song. A nonlinear probabilistic method for process monitoring. *Industrial & Engineering Chemistry Research*, 2010, 49(4): 1770–1778.

[27] L. H. Chiang, M. E. Kotanchek, A. K. Kordon. Fault diagno-

sis based on Fisher discriminant analysis and support vector machines. *Computers & Chemical Engineering*, 2004, 28 (8): 1389–1401.

[28] S. Verron, T. Tiplica, A. Kobi. Fault diagnosis of industrial systems by conditional Gaussian network including a distance rejection criterion. *Engineering Applications of Artificial Intelligence*, 2010, 23 (7): 1229–1235.

[29] I. Monroy, R. Benitez, G. Escudero, et al. A semi-supervised approach to fault diagnosis for chemical processes. *Computers & Chemical Engineering*, 2010, 34 (5): 631–642.

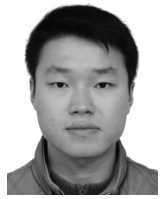
[30] K. Murphy. The Bayes net toolbox for Matlab. *Computing Science and Statistics*, 2001, 33 (2): 1024–1034.

[31] G. Wei, Z. Wang, H. Shu. Robust filtering with stochastic nonlinearities and multiple missing measurements. *Automatica*, 2009, 45(3): 836–841.

Biographies

Zhengdao Zhang was born in 1976. He received the Ph.D. degree in control theory and engineering in 2006 from Nanjing University of Aeronautics and Astronautics, China. He is currently an associate professor of the Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education) in Jiangnan University. From 2010 to 2011, he was a visiting scholar in Pennsylvania State University. He published more than 30 papers in journals and conferences. His research interests include fault diagnosis, and fault prognosis and applications.

E-mail: wxzzd.dr@gmail.com



Jinlin Zhu was born in 1988. He reveived his B.S. degree from Jiangnan University in 2010. He is now a master student of the Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education) in Jiangnan University. His research interests include Bayesian networks and fault diagnosis.

E-mail: neozzjl@gmail.com

Feng Pan was born in 1963. He is a professor and Ph.D. candidate supervisor in the School of Internet of Things at Jiangnan University. He received the M.S. degree in industrial automation and the Ph.D. degree in fermentation engineering both from Jiangnan University. He is a member of Association of Automation and IEEE. He is a holder of the Third-Grade College Young Teacher Prize (Teaching) of Fok Ying Tung Education Foundation. His research interests include process modeling, optimization and control, biochemical process intelligence control, computer distribute control system, advanced control theory & application.

E-mail: pan_feng_63@163.com