Fault Classification in Dynamic Processes Using Multiclass Relevance Vector Machine and Slow Feature Analysis

JIAN HUANG1,2, XU YANG1, YURI A. W. SHARDT3, AND XUEFENG YAN2

1Key Laboratory of Knowledge Automation for Industrial Processes, Ministry of Education, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China
2Key Laboratory of Advanced Control and Optimization for Chemical Processes, Ministry of Education, East China University of Science and Technology, Shanghai 200237, China
3Department of Automation Engineering, Technical University of Ilmenau, 98684 Ilmenau, Germany

Corresponding author: Xu Yang (yangxu@ustb.edu.cn)

This research was supported in part by the National Natural Science Foundation of China under Grant 61903026 and Grant 61673053, in part by the China Postdoctoral Science Foundation under Grant 2019M660462, in part by the Fundamental Research Funds for the Central Universities under Grant 222201917006, and in part by the National Key Research and Development Program of China under Grant 2017YFB0306403.

ABSTRACT This paper proposes a modified relevance vector machine with slow feature analysis fault classification for industrial processes. Traditional support vector machine classification does not work well when there are insufficient training samples. A relevance vector machine, which is a Bayesian learning-based probabilistic sparse model, is developed to determine the probabilistic prediction and sparse solutions for the fault category. This approach has the benefits of good generalization ability and robustness to small training samples. To maximize the dynamic separability between classes and reduce the computational complexity, slow feature analysis is used to extract the inner dynamic features and reduce the dimension. Experiments comparing the proposed method, relevance vector machine and support vector machine classification are performed using the Tennessee Eastman process. For all faults, relevance vector machine has a classification rate of 39%, while the proposed algorithm has an overall classification rate of 76.1%. This shows the efficiency and advantages of the proposed method.

INDEX TERMS Slow feature analysis, relevance vector machine, dynamic process, fault classification, process monitoring, statistical learning, support vector machine, feature extraction, process control.

I. INTRODUCTION

With the increase in industrial plant complexity, the ability to easily extract information has become more difficult, especially when considering product quality and process safety as critical parameters in the manufacturing process. Data-driven process monitoring techniques have been applied in modern industry to ensure the safety and stability of processes [1]–[8]. In recent years, many multivariate statistical process monitoring algorithms have been proposed for large-scale industrial processes. He et al. proposed a transition identification and process monitoring method for multimode processes using a dynamic mutual information similarity analysis [9]. A supervised non-Gaussian latent structure was introduced by He et al. to model the relationship between predictor and quality variables [10]. He and Zeng proposed double layer distributed process monitoring based on hierarchical multiblock decomposition for large-scale distributed processes [11]. However, the multivariable statistical methods are not designed for fault classification. Fault detection is used to identify whether the process is under normal condition, but cannot provide any information about the fault type. The main idea of fault classification is to build a classifier on the basis of some known fault categories. Fault classification can provide the connection between the detected fault occurrence and some known faults. Recognizing the type of fault is an important step for process monitoring.

A. CLASSIFICATION METHODS

Numerous machine learning methods have been used for fault classification. Ge et al. proposed a kernel-driven semisupervised Fisher discriminant analysis (FDA) model for nonlinear fault classification, which incorporates the advantages of
FDA for fault classification [12]. However, Chiang et al., who investigated the advantages of FDA and support vector machine (SVM), showed that nonlinear FDA outperforms FDA [13]. Jing and Hou studied the multiclass classification problem of SVM and principal component analysis (PCA) [14], while Gao and Hou improved the multiclass SVM using a PCA approach and applied it to the Tennessee Eastman (TE) process [15]. An SVM classifier is the most commonly used classification technique. However, SVM has the following three disadvantages [16], [17]. First, the number of support vectors grows as the size of the training set increases, which also causes the computational complexity to rapidly increase. Secondly, the output from the SVM does not use a probabilistic approach for estimating the conditional distribution, which implies that the prediction cannot capture the uncertainty. This means that the accuracy of SVM is sensitive to the training set. Thus, the SVM training set should include a variety of training samples. Thirdly, the classification performance of SVM is sensitive to the parameters. Therefore, a probabilistic and sparse classification algorithm needs to be developed.

Relevance vector machine (RVM) [16], [17] is a classification method that does not have any of the above limitations. Given its probabilistic Bayesian framework, RVM can be applied in cases where there are limited training samples. However, no research exists on the application of RVMs to the data-driven fault classification of chemical processes. Considering the sensitivity of the RVM prediction to the input dimension of the training set in industrial data, a sound method should be used to transform the training set into a lower-dimensional feature space, which maximizes the separability between classes.

B. DYNAMIC DIMENSIONAL REDUCTION METHODS

PCA, a widely used multivariate statistical algorithm, is used to reduce the dimensions. However, PCA is a static dimension reduction method. Sometimes it is necessary to consider the presence of time delay in dynamic systems. In recent years, slow feature analysis (SFA) [18] has been a popular method for extracting dynamic features by focusing on learning the time features that have a slow frequency. Shang et al. pointed out that the time dynamics in the extracted slow features (SFs) is an indicator of process changes [19]. Modified SFA methods have also been proposed [20], [21]. SFA can extract the dynamic representations of the original data from different levels of dynamics. The above studies have shown that SFA can easily extract dynamic features. However, in previous studies, the SFA algorithm was only used for fault detection, which does not provide information about the fault type. In fact, the SFA algorithm has not been applied to fault classification.

C. MOTIVATION FOR THIS PAPER

In modern chemical processes, sufficient training samples for fault classifiers are usually not available, which worsens the performance of traditional classifiers. RVM has the advantages of sparsity and probabilistic prediction. It can be applied when there are insufficient training samples. Furthermore, many chemical processes are large dimensional and have dynamic characteristics. Thus, such processes require that a dynamic dimensional reduction method be used to transform the training set into a lower-dimensional feature space. Since the SFA algorithm shows the inner dynamic characteristics of the faulty samples, it will indirectly increase the classification performance of RVM. Therefore, it seems plausible to combine SFA and RVM.

The contributions of this paper are the development of a robust, sparse RVM fault classification method that can be used when there are insufficient training samples and an SFA dimension reduction method that when combined with the RVM classifier can extract the distinct features, and hence, enhance the classification results. This paper proposes a modified multiclass RVM with SFA fault classification strategy for use in industrial processes. First, an SFA dynamic feature extraction model is built on the benchmark data. Then the important features are selected in order to reduce the dimension. After reducing the dimension, the high-dimensional training set is transformed into a low-dimensional dynamic feature space, which decreases the computational complexity of the RVM algorithm. Afterwards, some typical fault samples are collected and used for RVM training. RVM classifies the type of fault, providing a connection between the detected faulty samples and the known faults.

II. THEORETICAL BASIS

In this section, the SFA, SVM and RVM algorithms are reviewed. As well, the training results of the SVM and RVM algorithms are compared using a numerical example.

A. SLOW FEATURE ANALYSIS

SFA [22]–[24] extracts the slowly varying features. The input signal is expressed as \( x(t) = [x_1(t), x_2(t), \ldots, x_m(t)]^T \). The optimization objective of SFA is to find a transformation
function $g(x)$ such that the feature $s(t) = g(x(t))$ varies as slowly as possible [22], [23]. The function $g(x) = [g_1(x), g_2(x), \ldots, g_m(x)]^T$ is the transformation function, while $s(t) = [s_1(t), s_2(t), \ldots, s_m(t)]^T$ is the slow feature. The optimization problem for SFA is

$$\min_{g(\cdot)} \left\{ \frac{1}{2} \sum_{t} \langle s^2 \rangle_t \right\}$$

with the constraints

$$\langle s_1 \rangle_t = 0 \quad \text{(zero mean)},$$

$$\langle s^2 \rangle_t = 1 \quad \text{(unit variance)},$$

$$\forall i \neq j: \langle s_i s_j \rangle_t = 0 \quad \text{(uncorrelated)},$$

The SFA algorithm is formalized as

$$s = Wx$$

where $W$ is the weight matrix which is expressed as $W = [w_1, w_2, \ldots, w_m]^T$. The whitening step is carried out by the singular value decomposition (SVD) of $R = \{x(t)x(t)^T\}_t$, which is given as

$$R = U \Lambda U^T$$

The whitening matrix is $Q = \Lambda^{-1/2} U^T$. Then, the whitening step is written as

$$z = \Lambda^{-1/2} U^T x = Q x$$

Based on (5) and (7), the SF vector is given as

$$s = Wx = WQ^{-1} z = Pz$$

where $P = WQ^{-1}$. Clearly, $\langle zz^T \rangle_t = Q \{xx^T\} Q^T = I$ and $\langle z \rangle_t = 0$. Constraints (3) and (4) imply that

$$\langle ss^T \rangle_t = P \langle zz^T \rangle_t P^T = PP^T = I$$

The optimization problem for SFA is to find an orthogonal matrix $P$ to minimize $\langle s^2 \rangle_t$, which is given as

$$\langle s^2 \rangle_t = P^T \langle zz^T \rangle_t P$$

The optimal solution is obtained by taking the SVD of the covariance matrix $\langle zz^T \rangle_t$, that is,

$$\langle zz^T \rangle_t = P^T \Omega P$$

where $P$ is the eigenvector matrix and $\Omega = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m)$ is the eigenvalue matrix, where the eigenvalues $\lambda_i$ are placed in ascending order. Thus, $\langle s^2 \rangle_t = P^T \langle zz^T \rangle_t P = \lambda_i$. The weight matrix $W$ is then

$$W = PQ = P \Lambda^{-1/2} U^T$$

B. SUPPORT VECTOR MACHINE

The main idea of the SVM classifier is to find support vectors that define the bounding hyperplanes, so as to maximize the margin between both planes. Let the training set be $(x_i, t_i)$ for $i = 1, 2, \ldots, n$ and $t_i \in \{-1, 1\}$, where $x_i$ is the $i$th input sample and $t_i$ is the label corresponding to sample $x_i$. The SVM optimization problem requires solving the optimization problem

$$\min_{w, \xi} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \xi_i$$

subject to

$$t_i (w^T x_i + b) - 1 + \xi_i \geq 0 \quad \text{for } i = 1, 2, \ldots, n$$

where the slack variable $\xi_i$ represents the violation for training sample $x_i$. A penalty parameter $C$ is introduced to control the total violation while maximizing the margin. This introduces a trade-off between maximizing the margin and minimizing the violation. As well, a nonlinear transformation $z = \Phi(x)$ is used to project the training data onto a high-dimensional linear space. Based on functional theory, a kernel function should satisfy the Mercer condition.

The Lagrange dual problem is

$$\min_{\alpha} \frac{1}{2} \alpha^T (\mathbf{d} \cdot \mathbf{d}^T) \alpha - e^T \alpha$$

subject to $0 \leq \alpha \leq C$.

where $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_n]^T$ is the Lagrange multiplier vector with $\alpha_i \geq 0$, $t = [t_1, t_2, \ldots, t_n]^T$, $\mathbf{d} = [t_1 z_1, t_2 z_2, \ldots, t_n z_n]^T$, and $e$ is a unit column vector, that is, a column vector where each entry is 1. The input sample $x_i$ for which $\alpha_i \neq 0$ is called a support vector (SV).

The discriminant function $f(x)$ for a test sample $x$ can be obtained from

$$f(x) = \text{sign} \left( \sum_{i \in SV} \alpha_i t_i \Phi (x \cdot x_i) + b \right)$$

C. RELEVANCE VECTOR MACHINE

RVM seeks to maximize the possible sparsity, by discarding a large number of extremely small weights and thus, training samples that have no effect on the classification function. Originally, RVM [16], [17], [25], [26] was derived for binary classification. RVM is designed to predict the posterior probability of the membership of the input $x$. The training set for RVM is defined as $(x_i, t_i)$ for $i = 1, 2, \ldots, n$. The learning function is

$$y(x; w) = \sum_{i=1}^{n} \alpha_i K(x, x_i) + \alpha_0$$

where $K(x, x_i)$ is a kernel transformation and $\alpha_i$ is the weight. Generally, a Gaussian kernel function is used. Bayes’ rule gives the posterior probability of $w$, that is,

$$p(w|t, \alpha) = \frac{p(t|w, \alpha)p(w|\alpha)}{p(t|\alpha)}$$
where \( p(t|w, \alpha) \) is the posterior probability, \( p(w|\alpha) \) is the conditional prior likelihood for the hyperparameters \( \alpha = [\alpha_1, \ldots, \alpha_n]^T \), and \( p(t|\alpha) \) is the probability of the evidence. The logistic sigmoid function \( \sigma(y) = 1/(1 + e^{-y}) \) and Bernoulli distribution \( p(t|x) \) are used to generalize the linear model. Thus, (17) can be rewritten as

\[
p(t|w, \alpha) = \prod_{i=1}^{n} \sigma(y_i; w)^{t_i} \left[ 1 - \sigma(y_i; w) \right]^{1-t_i} \tag{18}
\]

where \( t = [t_1, t_2, \ldots, t_n]^T, t_i \in \{0, 1\}, \) and \( w = [\omega_0, \omega_1, \ldots, \omega_n]^T \). In (17), \( p(w|\alpha) \) is the Gaussian kernel function, where the weights \( w_i \) are obtained from the parameter \( \alpha_i \) using

\[
p(w|\alpha) = \prod_{i=1}^{n} \sqrt{\alpha_i} \exp \left( -\frac{\alpha_i w_i^2}{2} \right) \tag{19}
\]

The approximation procedure is based on Laplace’s method [27], which consists of the following three steps:

1. Given a fixed value of \( \alpha \), find the maximum posterior weights \( w_{MP} \) to provide the location of the mode of the posterior distribution. To solve the problem, the penalized logistic log-likelihood function is introduced. Since \( p(w|t, \alpha) \propto P(t|w)p(w|\alpha) \), the weights \( w_{MP} \) can be obtained from

\[
\log p(t|w)p(w|\alpha) = \sum_{i=1}^{n} \left[ t_i \log(y_i) + (1 - t_i) \log(1 - y_i) \right] - \frac{1}{2} w^T A w \tag{20}
\]

where \( y_i = \sigma(y_i; w) \) and \( A = \text{diag}(\alpha_0, \alpha_1, \ldots, \alpha_n) \).

2. Laplace’s method is used to find a quadratic approximation of the log-posterior based on its mode. Equation (20) is solved using the second-order Newton optimization method. The Hessian is

\[
\nabla w \nabla w \log p(w|t, \alpha)\big|_{w_{MP}} = -(\Phi^T B \Phi + A) \tag{21}
\]

where \( \Phi = [\phi(x_1), \phi(x_2), \ldots, \phi(x_n)]^T \) is a structure matrix, in which \( \phi(x_i)=[1, K(x_i, x_1), K(x_i, x_2), \ldots, K(x_i, x_n)]^T \); and \( B = \text{diag}(\beta_1, \beta_2, \ldots, \beta_n) \) is a diagonal matrix with \( \beta_i = \sigma(y_i) [1 - \sigma(y_i)] \). The Hessian matrix is converse and inverted to give \( \Sigma = (\Phi^T B \Phi + A)^{-1} \) for the Gaussian approximation to the posterior function at \( w_{MP} \). Given that \( \nabla w \log p(w|t, \alpha)\big|_{w_{MP}} = 0 \), \( w_{MP} \) can be written as \( w_{MP} = \Sigma \Phi B \Phi \).

3. The hyperparameters \( \alpha \) are updated using \( \Sigma \) and \( w_{MP} \)

\[
\alpha_{i}^{\text{new}} = \frac{\gamma_i}{\mu_i} \tag{22}
\]

where \( \gamma_i = 1 - \alpha_i \Sigma_{ii} \), \( \Sigma_{ii} \) is the \((i, i)\) diagonal entry of the matrix \( \Sigma \), and \( \mu = [\mu_1, \ldots, \mu_n]^T \) is the posterior mean weight vector.

This optimization procedure forces most of the \( \alpha_i \) to infinity, which implies that, based on (19), the corresponding weights \( w_i \) are discarded. The remaining vectors, which are called the relevance vectors (RVs), give a sparse solution.

The discriminant function \( f(x) \) can be calculated as

\[
f(x) = \sum_{i=1}^{r} w_i K(x, x_i) + w_0 \tag{23}
\]

where \( r \) is the number of RVs, \( x_i \) is the RV, and \( w_i \) represents the remaining weights.

Fault classification usually involves more than 2 classes, whereas RVM and SVM are intrinsically binary classifiers. To convert binary classifiers into multiclass classifiers, the general strategy is to combine an ensemble of binary classifiers based on some decision rules. In this study, the one-versus-one strategy is used [17]. In the one-versus-one approach, an ensemble of models, whose training set is made of any two classes is designed. Therefore, the total number of classifiers is \( M(M - 1)/2 \). When testing an unknown sample, all the discriminant functions need to be calculated. A voting mechanism is used to count the score of each category. The category that obtains the most votes determines the category of the test sample.

D. COMPARISON OF SVM AND RVM USING A NUMERICAL EXAMPLE

Consider the following numerical example with the measurements \( x = [x_1, x_2] \)

\[
x_2 = x_1^2 + y \alpha + v \tag{24}
\]

where \( x_1 \) is uniformly distributed over the interval [1, 4], \( v \) is a random Gaussian noise with zero mean and variance of 0.25, \( \alpha \) is an unmeasurable variable with a uniform distribution over the interval [1, 3], and \( y \in \{-1, 1\} \) is the label for sample \( x \). The training set consists of 100 samples of which the first 50 samples are labelled 1 and the last 50 samples are labelled -1.

Figure 1 shows the results of training the SVM and RVM classifiers. The subfigure on the left shows the results for the SVM classifier, while the subfigure on the right shows the same for the RVM classifier. The circled samples represent the support (respectively, relevant) vectors. Comparing the two subfigures, it can be seen that SVM requires 40 support vectors, while RVM only requires 4 relevance vectors. Thus, RVM shows higher sparsity.
Algorithm 1 The Multiclass RVM approach

Step 0: Initialization

Define the class number \( M \). Set the training class set \( C = \{ S_{\text{train}}^{(1)}, S_{\text{train}}^{(2)}, \ldots, S_{\text{train}}^{(M)} \} \).

Step 1: Training

For \( i = 1, 2, \ldots, M \)

For \( j = i + 1, i + 2, \ldots, M \)

Choose the training set \( S_{\text{train}}^{(i)} \) and \( S_{\text{train}}^{(j)} \). Obtain the discriminant function \( f_{ij}(x) \)

End

Step 2: Testing

For \( i = 1, 2, \ldots, M \)

For \( j = i + 1, i + 2, \ldots, M \)

Test sample \( x \) using \( f_{ij}(x) \) to define \( P(x, S_{\text{train}}^{(i)}) \) and \( P(x, S_{\text{train}}^{(j)}) \). If \( P(x, S_{\text{train}}^{(i)}) > P(x, S_{\text{train}}^{(j)}) \), set \( S_i(x) = S_i(x) + 1 \); otherwise, set \( S_j(x) = S_j(x) + 1 \).

End

End

Step 3: Decision Making

Define the class assignment for \( x \) using (28).

Stop.

For the testing vector \( s_{\text{test}} \), the discriminant function set \( f_{ij}(x) \) is used to determine the fault category for each binary RVM classifier. Then, a voting mechanism is used to count the score of each fault category. The test vector belongs to the fault category that obtains the most votes.

Figure 2 shows the flowchart of the SFA-RVM algorithm.

IV. SIMULATIONS AND EXPERIMENTS USING THE TE PROCESS

A. BACKGROUND ABOUT THE TE PROCESS

The TE process was introduced by Downs and Vogel [28]. The control system for the TE process is shown in Figure 3 [29]. There are 22 measured variables and 12 manipulated variables for the TE process. There are 500 samples in the normal dataset. To further enhance the TE process, 21 different faults are simulated. For the training dataset, there are 480 faulty samples for each fault. For the testing dataset, the first 160 samples in the faulty dataset are collected under normal condition, while the remaining 800 samples are faulty data. The measurements contain random noise which is carried out using a random seed. The sampling frequency for TE process is 3 minutes. The simulation dataset of the TE is provided at http://web.mit.edu/braatzgroup/links.html. In the fault classification procedure, the training set consists of 100 randomly chosen samples from the training faulty samples for each class.

B. FEATURE EXTRACTION AND DIMENSION REDUCTION OF THE TE PROCESS DATA

The goal of feature extraction is to retain only the most significant features. The number of SFs retained for dimension
reduction is based on the $L_2$-norm method. The weight vectors are re-ordered in descending order based on their $L_2$-norm. The number of slow features $k$ is determined by

$$
\sum_{i=1}^{k} L_i / \sum_{i=1}^{m} L_i > \theta
$$

(29)

where $\theta$ is the selection threshold.

The number of SFs depends on the data for the current application. After testing a number of threshold values, the most important SFs are retained when the $\theta$ equals to 0.9 with the data used. In the TE process, the number of slow features is 21 according to (29). Figure 4 shows the visualization of the first, second, and third dimensions of the training set. The different categories of fault samples are separated using different colors. Since faults 3, 9, and 15 are commonly considered to be hard to detect, the visualization of the training set does not include faults 3, 9, and 15. The training samples are gathered in a small area and are hardly visually separated. The TE process contains 33 variables that are combinations of latent factors. The characteristic of high dimensionality makes the classification model complex. Figures 5 and 6 show the visualization for the first 6 dimensions of the SFA dimensionally reduced dataset. Compared with Figure 4, part of the faulty samples in the SFA dimensionally reduced dataset are visually separated from the other fault categories. The extracted dynamic features maximize the separation between classes, which highlights the inner dynamic characteristics of faulty samples and indirectly increases the classification performance of RVM.

C. SIMULATION RESULTS

To test the online fault classification performance of the proposed method, regular RVM and SVM are used for comparison. The optimal parameters of the SVM are obtained using cross-validation and the grid search method. Table 1 gives the correct fault classification rate (CFCR) and precision for each fault for RVM, SVM, and SFA-RVM, while Table 2 shows the fusion matrix of the classification results using the RVM, SVM, and SFA-RVM methods for each fault type. Each row gives the fault classification results for a particular fault. The bold numbers on the diagonal show the number of samples that are properly classified. The CFCR is calculated using

$$
\text{CFCR} = \frac{\text{Number of samples correctly classified}}{\text{Total number of testing samples}}
$$

(30)

The precision for each fault is calculated by

$$
\text{Precision} = \frac{\text{Number of samples correctly classified as this fault}}{\text{Total number of samples classified as this fault}}
$$

(31)

In Tables 1 and 2, it can be seen that all three methods have very high correct fault classification rates for Faults 1, 2, and 7. The fault magnitudes for Faults 1, 2, and 7 are large,
and their distinct fault features make it easy to classify the fault. However, for all methods, the correct fault classification rates for Faults 3, 9, and 15 are very poor. This can be attributed to the fact that these 3 faults have little extractable fault information, which then confuses the fault detection algorithms. The fault classification results for faults without Faults 3, 9, and 15 are also given in Table 1. Furthermore, RVM has a low correct fault classification rate (< 30%) for Faults 4, 5, 10, 11, 16, 19, and 20. These faults are conflated with each other, that is, RVM cannot distinguish these faults. Compared with RVM, SVM has better fault classification results for these faults. However, SVM has poor correct fault classification rates for Faults 6 and 18, where the accuracy is only 10%. The fusion matrix in Table 2 shows that most test samples for Faults 6 and 18 are incorrectly classified as Fault 21. The proposed SFA-RVM algorithm has improved the correct fault classification rates. SFA-RVM achieves the best classification rates for Faults 5, 6, 10, 11, 12, 16, 19, and 20. Of note, the classification rates for SFA-RVM are over 70% for faults excluding Faults 3, 9, and 15 (the last 3 columns in Table 1). Furthermore, the classification rates for Faults 1, 2, 5, 6, and 7 are over 90%. Thus, it has been shown that the proposed SFA-RVM is more robust than RVM and SVM.

Fault 10 is a random variation in the C feed temperature. RVM can hardly recognize fault 10 with low fault classification rates (approximately 10%). SVM shows better fault classification results. However, the rates are lower than 50%. In the previous fault detection studies, the fault detection rates of some modified process monitoring method for Fault 10 are approximately 0.85 [7], [30], [31]. Some traditional methods such as PCA and DPCA can only detect 50% of the fault samples. The reason why Fault 10 is hard to recognize is that Fault 10 is a random fault, in which the fault amplitude varies considerably. Amplitudes of some fault samples are extremely low, which makes the fault samples similar to the normal samples. Fault detection is a binary classification problem, which is much easier than fault classification, especially for the TE process. The fault classification rate of SFA-RVM (excluding Faults 3, 9, and 15) reaches 0.83, which is close to the best fault detection rate.

Fault 11 is a random change in the reactor cooling inlet temperature. SVM has poor correct fault classification rates for faults without Faults 3, 9, and 15, and their distinct fault features make it easy to classify the fault. However, for all methods, the correct fault classification rates for Faults 3, 9, and 15 are very poor. This can be attributed to the fact that these 3 faults have little extractable fault information, which then confuses the fault detection algorithms. The fault classification results for faults without Faults 3, 9, and 15 are also given in Table 1. Furthermore, RVM has a low correct fault classification rate (< 30%) for Faults 4, 5, 10, 11, 16, 19, and 20. These faults are conflated with each other, that is, RVM cannot distinguish these faults. Compared with RVM, SVM has better fault classification results for these faults. However, SVM has poor correct fault classification rates for Faults 6 and 18, where the accuracy is only 10%. The fusion matrix in Table 2 shows that most test samples for Faults 6 and 18 are incorrectly classified as Fault 21. The proposed SFA-RVM algorithm has improved the correct fault classification rates. SFA-RVM achieves the best classification rates for Faults 5, 6, 10, 11, 12, 16, 19, and 20. Of note, the classification rates for SFA-RVM are over 70% for faults excluding Faults 3, 9, and 15 (the last 3 columns in Table 1). Furthermore, the classification rates for Faults 1, 2, 5, 6, and 7 are over 90%. Thus, it has been shown that the proposed SFA-RVM is more robust than RVM and SVM.

Fault 10 is a random variation in the C feed temperature. RVM can hardly recognize fault 10 with low fault classification rates (approximately 10%). SVM shows better fault classification results. However, the rates are lower than 50%. In the previous fault detection studies, the fault detection rates of some modified process monitoring method for Fault 10 are approximately 0.85 [7], [30], [31]. Some traditional methods such as PCA and DPCA can only detect 50% of the fault samples. The reason why Fault 10 is hard to recognize is that Fault 10 is a random fault, in which the fault amplitude varies considerably. Amplitudes of some fault samples are extremely low, which makes the fault samples similar to the normal samples. Fault detection is a binary classification problem, which is much easier than fault classification, especially for the TE process. The fault classification rate of SFA-RVM (excluding Faults 3, 9, and 15) reaches 0.83, which is close to the best fault detection rate.

Fault 11 is a random change in the reactor cooling inlet temperature. SVM has poor correct fault classification rates for faults without Faults 3, 9, and 15, and their distinct fault features make it easy to classify the fault. However, for all methods, the correct fault classification rates for Faults 3, 9, and 15 are very poor. This can be attributed to the fact that these 3 faults have little extractable fault information, which then confuses the fault detection algorithms. The fault classification results for faults without Faults 3, 9, and 15 are also given in Table 1. Furthermore, RVM has a low correct fault classification rate (< 30%) for Faults 4, 5, 10, 11, 16, 19, and 20. These faults are conflated with each other, that is, RVM cannot distinguish these faults. Compared with RVM, SVM has better fault classification results for these faults. However, SVM has poor correct fault classification rates for Faults 6 and 18, where the accuracy is only 10%. The fusion matrix in Table 2 shows that most test samples for Faults 6 and 18 are incorrectly classified as Fault 21. The proposed SFA-RVM algorithm has improved the correct fault classification rates. SFA-RVM achieves the best classification rates for Faults 5, 6, 10, 11, 12, 16, 19, and 20. Of note, the classification rates for SFA-RVM are over 70% for faults excluding Faults 3, 9, and 15 (the last 3 columns in Table 1). Furthermore, the classification rates for Faults 1, 2, 5, 6, and 7 are over 90%. Thus, it has been shown that the proposed SFA-RVM is more robust than RVM and SVM.

Fault 10 is a random variation in the C feed temperature. RVM can hardly recognize fault 10 with low fault classification rates (approximately 10%). SVM shows better fault classification results. However, the rates are lower than 50%. In the previous fault detection studies, the fault detection rates of some modified process monitoring method for Fault 10 are approximately 0.85 [7], [30], [31]. Some traditional methods such as PCA and DPCA can only detect 50% of the fault samples. The reason why Fault 10 is hard to recognize is that Fault 10 is a random fault, in which the fault amplitude varies considerably. Amplitudes of some fault samples are extremely low, which makes the fault samples similar to the normal samples. Fault detection is a binary classification problem, which is much easier than fault classification, especially for the TE process. The fault classification rate of SFA-RVM (excluding Faults 3, 9, and 15) reaches 0.83, which is close to the best fault detection rate.

TABLE 2. The fusion matrix for fault classification (without faults 3, 9, and 15).

| NO. | 1    | 2    | 4    | 5    | 6    | 7    | 8    | 10   | 11   | 12   | 13   | 14   | 16   | 17   | 18   | 19   | 20   | 21   |
|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| RVM | 1 783| 0 0 0| 4 3 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2| 0 0 2|
| SVM | 1 795| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0|
| SFA | 1 780| 0 0 0| 0 0 0| 8 0 6| 3 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0| 0 0 0|

FIGURE 4. Visualization of the first, second, and third dimensions of the scaled training dataset.

FIGURE 5. Visualization of the first, second, and third dimensions of the SFA dimensionally reduced training dataset.
that it also has low fault amplitudes. The fault classification rates of RVM are 0.11 and 0.13. SVM has better results with a classification rate over 50%. In the previous studies, the fault detection rates of some advanced methods are approximately 0.85 [7], [30], [31]. The fault classification rate of SFA-RVM (excluding Faults 3, 9, and 15) is 0.74. It should be noted that Faults 4 and 11 occurred in the reactor cooling inlet temperature. The only difference between Faults 4 and 11 is that Fault 4 is a step fault whereas Fault 11 is random. Some of the fault samples from Fault 11 are quite similar to samples from Fault 4, which will confuse the fault classification model. It can be seen in Table 2 that there are 74 test samples belonging to Fault 11 that are classified as belonging to Fault 4. In industrial processes, different faults may result in similar evidence because of the presence of the control system. Therefore, the definition of fault categories before model training is also important.

Table 3 shows the overall classification rates for RVM, SVM, decision tree, SFA-based decision tree (SFA-decision tree), and SFA-RVM, as well as the classification rates for different number of training samples (100 and 200 training samples). For all faults (100 training samples), RVM can only recognize 39% of the test samples, while the accuracy of SVM reaches 55.8%. Furthermore, the SFA-RVM algorithm has an overall classification rate of 76.1%, which is the best fault recognition algorithm. The overall classification rates excluding Faults 3, 9, and 15 are higher than those for all faults. The RVM approach leads to a high dimensional solution that causes the accuracy to be lower than 50%. Since SVM is not a probabilistic algorithm, a complete training set is needed to contain enough samples. If the training set is limited, the accuracy of SVM is sensitive to the size of the training set. For example, some of the TE process faults are random faults, which means that the amplitude of the faults varies. If the randomly chosen training sets differ greatly at different times, the training models of SVM may have many differences. The proposed SFA-RVM algorithm extracts the inner dynamic features to maximize the separability between classes. Dimension reduction avoids the computational complexity of the RVM training. The extracted features are good dynamic representations of the TE process. RVM has probabilistic prediction and sparsity properties, which achieve good generalization ability and robustness for small scaled training samples. The proposed method takes advantages of both SFA in extracting dynamic features and RVM in classifying.

V. CONCLUSION

In this paper, a modified RVM with the SFA algorithm is proposed for fault classification in dynamic industrial processes. SFA extracts the lower-dimensional dynamic features, which maximizes separability and reduces the computational complexity of RVM. Simultaneously, a one-versus-one multiclass RVM model is built to recognize the fault category when there are insufficient training samples. RVM classifies the type of faults, providing the connection between the detected faulty samples and known faults. RVM has the advantages of probabilistic prediction and sparsity properties for small training sets. Detailed comparative studies between the SFA-RVM method and traditional RVM and SVM were carried out using the TE benchmark process. The simulations show that the proposed SFA-RVM method has much better overall fault classification rates than either the RVM or the SVM method alone. For all faults, RVM can only recognize 39%. The SFA-RVM algorithm has an overall classification rate of 76.1%, which is a remarkable improvement.

However, the proposed method was not tested in a real process. Collecting sufficient data with faults can be an issue for real processes. Furthermore, the knowledge of fault type may be not clear, that is, the training samples for fault classifiers are not available as fast as required. Limited by the diversity and quantity of the training samples for fault classification, the dataset of a realistic experimental application is hard to obtain. Finally, even if faulty data can be obtained, it may not be possible to obtain meaningful fault classifications due to the fact that some of the faults may have a very weak signal. Thus, future work will focus on the application of this method to fault classification on a real process and improving its ability to better handle weak faults.

REFERENCES

[1] Z. Ge, Z. Song, and F. Gao, “Review of recent research on data-based process monitoring,” Ind. Eng. Chem. Res., vol. 52, no. 10, pp. 3543–3562, 2013.

[2] S. J. Qin, “Survey on data-driven industrial process monitoring and diagnosis,” Annu. Rev. Control, vol. 36, no. 2, pp. 220–234, Dec. 2012.

[3] S. Yin, X. Li, H. Gao, and O. Kaynak, “Data-based techniques focused on modern industry: An overview,” IEEE Trans. Ind. Electron., vol. 62, no. 1, pp. 657–667, Jan. 2015.
