An on-line framework for monitoring nonlinear processes with multiple operating modes

Ruomu Tan, Tian Cong, James R. Ottewill, Jerzy Baranowski, Nina F. Thornhill

A Centre for Process Systems Engineering, Imperial College London, London, UK
b Department of Automatic Control and Robotics, AGH University of Science and Technology, Kraków, Poland
c ABB Power Grids Research, Kraków, Poland

A R T I C L E   I N F O

Article history:
Received 29 August 2019
Revised 10 February 2020
Accepted 11 March 2020

Keywords:
Fault detection
Unsupervised learning
Process monitoring
Multimode process
Kernel method

A B S T R A C T

A multivariate statistical process monitoring scheme should be able to describe multimodal data. Multimodality typically arises in process data due to varying production regimes. Moreover, multimodality may influence how easy it is for process operators to interpret the monitoring results. To address these challenges, this paper proposes an on-line monitoring framework for anomaly detection where an anomaly may either indicate a fault occurring and developing in the process or the process moving to a new operating mode. The framework incorporates the Dirichlet process, which is an unsupervised clustering method, and kernel principal component analysis with a new kernel specialized for multimode data. A monitoring model is trained using the data obtained from several healthy operating modes. When online, if a new healthy operating mode is confirmed by an operator, the monitoring model is updated using data collected in the new mode. Implementation issues of this framework, including the parameter tuning for the kernel and the selection of anomaly indicators, are also discussed. A bivariate numerical simulation is used to demonstrate the performance of anomaly detection of the monitoring model. The ability of this framework in model updating and anomaly detection in new operating modes is shown on data from an industrial-scale process using the PRONTO benchmark dataset. The examples will also demonstrate the industrial applicability of the proposed framework.

© 2020 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY license. (http://creativecommons.org/licenses/by/4.0/)

1. Introduction

Changes in production regimes and market demands may result in a process running in several different healthy operating modes. The data collected in such processes are multimodal [1]. An important industrial requirement when monitoring such processes is that it must be possible to detect when a process is in one of several healthy operating modes and when it is developing a fault. Key to this is an anomaly indicator that detects anomalies in the data with respect to any of the known healthy operating modes.

To meet the requirement, the monitoring model should be able to account for all of the known healthy modes. However, new operating modes may emerge as the production regimes change. Therefore, whilst a detected anomaly might be symptomatic of a developing fault, it may also indicate that the process has moved into a healthy operating mode which is not adequately described by the monitoring model. It is desirable to include an on-line model update step so that new data collected from a new healthy operating mode can be incorporated into the monitoring model [2] in order to minimize future false and missed alarms.

This paper aims to establish a monitoring framework with an unsupervised off-line training step that can obtain a single monitoring model for multimode data and an on-line model update step that can improve the performance of monitoring multimode processes. The number of modes in the training data is determined automatically and the training samples are labelled accordingly. Moreover, the anomaly indicator developed in this paper is equally sensitive and robust for all modes of operation. It detects the extent of an anomalous behaviour with respect to the known healthy operating modes regardless of which mode the process is running in.

Additionally, data collected from new healthy operating modes can be incorporated allowing the monitoring model to be updated on-line. This prevents false alarms being triggered in the future by the new healthy mode, and further allows faults to be detected...
when the process deviates away from any of the healthy modes, including the newly considered one.

The paper proceeds as follows. Section 2 gives an overview of relevant literature and describes the industrial requirements for a method of anomaly detection for multimode processes. Section 4 describes the methods used for training and tuning the monitoring model and discusses the choice of anomaly indicators and the tuning of parameters. Section 5 presents a flowchart summarizing the framework, showing the steps involved and the online model updating. A numerical example with multimode and nonlinear data is presented in Section 6 showing the performance of monitoring model training. In the same section the monitoring framework with anomaly detection and model update is tested using the PRONTO experimental benchmark dataset. The dataset is used to simulate a case where the process moves to a previously unseen operating mode and when a fault develops. Future directions for improving this framework are discussed in Section 7. The paper ends with the conclusions in Section 8.

2. Background

Faults are undesirable operating conditions which may develop into process failures. The development of a fault often leads to process variables deviating from their equivalent healthy values, resulting in anomalies in process data. Multivariate Statistical Process Monitoring (MSPM) approaches attempt to detect these anomalies by building monitoring models based on data recorded during periods of healthy operation.

2.1. Fault detection in multimodal processes

According to a recent review paper [3], data-driven multimode process monitoring approaches are evaluated according to the false and missed alarms and other metrics. However, the interpretability of the monitoring results obtained by these approaches for multimode processes have been afforded less attention. For example, approaches based on building multiple monitoring models to describe the different operating modes [3] typically output several anomaly indicator values. Even if, for a given sample, the anomaly indicators of two or more modes have similar values, the associated extent of the anomalous behaviour indicated in each of the modes can vary significantly. An additional decision-level fusion step, which combines the multiple anomaly indicators to determine the occurrence of a fault, has been proposed to account for this issue [4-6].

A multimode process may not be linear, and its multimode nature means there are ranges of values of the variables that are not encountered in normal operations. Hence nonlinear MSPM approaches [7-11], especially Kernel Principal Component Analysis (KPCA) [8,10], have been used to build single monitoring models for multimode processes. Nevertheless, nonlinear algorithms are not guaranteed to work well on multimodal data. For example, the Radial Basis Function (RBF), which is the most widely used kernel function in KPCA, can handle nonlinear data. However, a previous investigation [12] showed that RBF kernels perform badly when applied to multimodal data. Therefore, the kernel should be specified properly so as to make KPCA a good candidate for building a single monitoring model. One example is the Non-stationary Discrete Convolution (NSDC) kernel [12], which achieves a better monitoring model for multimode data than the RBF kernel.

2.2. Anomaly indicators

Another task in building a single monitoring model is to enhance the interpretability of the monitoring result by achieving an anomaly indicator that is useful for process operators. Operators need to detect the presence of faults with acceptable levels of false and missed alarms. This requires an anomaly indicator that has the same interpretation regardless of the mode in which the process is operating.

A common observation from the anomaly indicators achieved by various nonlinear monitoring techniques is that, for nonlinear processes with a single operating mode, the same magnitude of the anomaly indicator represents the same level of anomaly in the process and the control limit does not change over time [13-16]. Such behaviour is desired in practice because it is easy for operators to determine if the process has anomalies by inspecting the anomaly indicator. This may not be guaranteed in multimodal process monitoring, however, because the multiple operating modes often result in significantly different steady states of the process variables and, if the process is also nonlinear, different static process models in each mode. For example in [17,18], the control limit changes as the operating mode changes in the process and the same magnitude of the anomaly indicator does not indicate the same level of anomaly if the mode has changed. The NSDC kernel is designed to describe multimodal data accurately whilst achieving the behaviour that the same magnitude of the anomaly indicator represents the same level of anomaly in the process [12]. This paper aims to achieve a monitoring approach that gives an anomaly indicator that has consistent behaviour regardless of changes in steady states or the static models caused by varying healthy operating modes.

2.3. Data clustering and on-line updating

The NSDC-KPCA approach requires the training data to be clustered a priori. When clustering the historical data used for training, the number of operating modes existing in the data may not be known. Data-driven clustering methods such as K-means [19] and expectation maximization method [20], typically need to know the number of clusters a-priori [21]. In contrast, the Dirichlet Process (DP), which is a non-parametric Bayesian method [22], can automatically determine the number of clusters when clustering the data. After applying DP, the data are organized to be used in NSDC-KPCA.

The off-line training step of the proposed framework assumes that all training data are from healthy operating modes. This step uses DP to automatically determine the number of modes in the training data and to label the training samples accordingly. This step is unsupervised, in that it does not require the information about which mode each data sample belongs to. The training step then employs NSDC-KPCA in order to reduce false and missed alarms. Unlike the multiple model approaches such as [23,24] and [17], the NSDC-KPCA approach does not require the classification of a test sample or the fusion of the monitoring results. Moreover, the magnitude of an anomaly indicator achieved by NSDC-KPCA reflects the level of anomalies in the data with respect to the known healthy operating modes regardless of which mode the process is running in. In the on-line update step, new data collected from new healthy operating modes can be incorporated via the DP-NSDC-KPCA approach allowing the monitoring model to be updated on-line. Preliminary results of such a framework were presented in [25]. This paper extends [25] in the following aspects:

1. Demonstrating the performance of monitoring modelling and anomaly detection of the proposed framework on a simulated multimode dataset with a nonlinear mode;
2. Considering the influence of the kernel width of the NSDC kernel on the monitoring model built by NSDC-KPCA and proposing a tuning strategy for the kernel width;
3. Proposing an anomaly indicator that suits the NSDC-KPCA;
4. Applying the framework with properly tuned kernel widths and the new anomaly indicator to both the numerical simulation and the PRONTO dataset.

3. Glossary of terms

Acronyms used in this paper are summarized in Table 1.

4. Training of the monitoring model

This section will introduce the DP-GMM approach for clustering unlabelled data obtained from multiple healthy operating modes. The section will also describe the NSDC-KPCA approach for building monitoring models. The following bivariate algebraic example with four modes is used to illustrate the behaviour of the DP-GMM and NSDC-KPCA approaches:

Mode 1:

\[ x_1 = e_{11} + 8 \]
\[ x_2 = -0.2x_1 + 5 + e_{12} \]

where \( e_{11} \sim \mathcal{N}(0, 2.25) \) and \( e_{12} \sim \mathcal{N}(0, 0.25) \).

Mode 2:

\[ x_1 = e_{21} \]
\[ x_2 = 1.5x_1 + e_{22} \]

where \( e_{21} \sim \mathcal{N}(0, 1) \) and \( e_{22} \sim \mathcal{N}(0, 0.9) \).

Mode 3:

\[ x_1 = e_{31} + 15 \]
\[ x_2 = (x_1 - 15)^2 + 5 + e_{32} \]

where \( e_{31} \sim \mathcal{U}[-2, 2] \) and \( e_{32} \sim \mathcal{N}(0, 0.09) \).

Mode 4:

\[ x_1 = e_{41} + 9 \]
\[ x_2 = \frac{1}{3} x_1 - 4 + e_{42} \]

where \( e_{41} \sim \mathcal{N}(0, 1) \) and \( e_{42} \sim \mathcal{N}(0, 0.25) \). \( \mathcal{N} \) denotes Gaussian distributions and \( \mathcal{U} \) denotes uniform distributions. The model of \( x_1 \) and \( x_2 \) is nonlinear in Mode 3. The data generated from this example are visualized in Fig. 1.

4.1. Dirichlet process Gaussian mixture models for clustering

The Dirichlet Process Gaussian Mixture Model (DP-GMM) assumes that the data to be clustered are samples drawn from Gaussian Mixture Models (GMMs) [26], and applies DP for inferring a GMM for the data.

4.1.1. Gaussian mixture models

An \( m \)-dimensional random variable \( \mathbf{x} \) following a Gaussian mixture model with \( J \) components can be described as:

\[ \mathbf{x} \sim \mathcal{N}(\mu_j, \Sigma_j) \] with probability \( \pi_j \)

\[ \text{s.t.} \quad \pi_j > 0 \quad \forall j, \quad \sum_{j=1}^{J} \pi_j = 1 \]

(5)

where the \( j \)-th component is an \( m \)-dimensional multivariate Gaussian distribution with mean \( \mu_j \) and covariance \( \Sigma_j \), \( \mathbf{x} \) is assumed to be drawn for the \( j \)-th component with a mixture probability \( \pi_j \).

4.1.2. DP-GMMs

A multimodal training dataset \( \mathbf{X} \) is assumed to have samples of \( \mathbf{x} \) which follow the GMM model in Eq. (5). The objective of DP-GMM clustering is to use \( \mathbf{X} \) to determine the number of the components \( J \), the mean \( \mu_j \) and the covariance \( \Sigma_j \) associated with each component and from which component a sample in \( \mathbf{X} \) is drawn. Such inference is done in an iterative manner with two steps: the inference of means \( \mu_j \), and covariances \( \Sigma_j \) of the Gaussian components and the inference of the labels of the training samples indicating from which component a sample is drawn. All the parameters are assumed to follow certain distributions and these distributions are updated using observations, by applying the Dirichlet distribution and the Dirichlet Process.

First, the vector of mixture probabilities \( \pi = \{\pi_1, \ldots, \pi_J\} \) is assumed to follow the Dirichlet distribution [27]. The authors of [28,29] have shown that the means \( \mu_j \) and covariance matrices \( \Sigma_j \) follow a discrete distribution \( G(\cdot) \) generated from a Dirichlet Process specified by the base distribution \( G_0 \) and the concentration parameter \( \alpha \). A DP-GMM model for \( \mu_j, \Sigma_j \) and \( \pi_j \) is written as

\[ G(\cdot) \sim DP(\alpha, G_0) \]
\[ \mu_j, \Sigma_j \sim G(\cdot) \quad \text{for} \quad j = 1, \ldots, J \]
\[ \pi \sim Dir(\frac{\alpha}{J}, \ldots, \frac{\alpha}{J}) \]

(6)

where \( DP \) and \( Dir \) represent the Dirichlet Process and the Dirichlet Distribution, respectively.

The inference step for \( \mu_j \) and \( \Sigma_j \) is to find the updated distribution \( G(\cdot) \) using the data from the \( j \)-th Gaussian component. From a Bayesian perspective, the base distribution \( G_0 \) is the prior distribution of \( \mu_j \) and \( \Sigma_j \) while \( G(\cdot) \) is the posterior distribution to be updated given the samples in \( \mathbf{X} \). The following Normal Inverse

![Fig. 1. Visualization of the numerical example.](image-url)
Wishart (NIW) distribution \[30\] is selected as \(G_0\) because it is a conjugate prior of the multivariate Gaussian distribution:

\[
\Sigma_j \sim IW(\nu_0, \Lambda_0)
\]

\[
\mu_j | \Sigma_j \sim N \left( \mu_0, \Sigma_j / \kappa_0 \right)
\]

(\(\mu_j, \Sigma_j\) \(\sim\) \(IW(\Theta_0)\)) \(7\)

where \(\Sigma_j\) follows an Inverse Wishart distribution and \(\mu_j\) given \(\Sigma_j\) follows a Gaussian distribution. The parameters of this NIW distribution are \(\Theta_0 = \{\mu_0, \kappa_0, \nu_0, \Lambda_0\}\), where \(\nu_0\) and \(\kappa_0\) are positive values. The inference step for \(\mu_j\) and \(\Sigma_j\) involves calculation of the posterior distribution \(G(\cdot)\) and sampling \(\mu_j\) and \(\Sigma_j\) from the updated \(G(\cdot)\). The updated \(G(\cdot)\) also has the NIW formulation in Eq. (7), but the parameter vector \(\Theta\) is updated using \(X\). For the dataset \(X = \{x_1, \ldots, x_n\} \in \mathbb{R}^{n \times N}\) to be clustered, the inference step for the labels of samples uses a vector \(I = \{i_1, \ldots, i_n\} \in \mathbb{R}^{1 \times N}\), which denotes the cluster indices for each sample in \(X\). \(i_j\) takes values from 1 to \(J\) and indicates that the \(i\)-th sample \(x_i\) is drawn from the \(i_j\)-th Gaussian component.

4.1.3. Inference

According to Theorem 2 of \[29\], the distribution of the GMM given the training samples is the stationary distribution of the Markov chain, and convergence to the stationary distribution is independent of initial assumptions. The inference steps below are based on Gibbs sampling following \[31\]. Gibbs sampling infers \(I\) and \((\mu_j, \Sigma_j)\) by iteratively drawing samples of \(I\) and \((\mu_j, \Sigma_j)\) from the joint distribution of \(I\) and \((\mu_j, \Sigma_j)\). The joint distribution is updated when the new inference of \(I\) or \((\mu_j, \Sigma_j)\) is available.

a. Initialization: \(\Theta_0\) and \(\kappa_0\) are assigned and \(\mu_j, \Sigma_j\) are sampled from \(G_0\) to generate an initial group of clusters.

b. Updating the parameters of each Gaussian component \((\mu_j, \Sigma_j)\) for \(j = 1, \ldots, J\): The parameters \(\mu_j\) and \(\Sigma_j\) are sampled from the updated posterior NIW distribution \(IW(\Theta(j))\).

c. Updating the cluster indices vector \(I\): This step assigns the samples into existing clusters or creates new clusters. For a sample \(x^* \in X\), it is clustered to the \(j\)-th cluster with the probability

\[
p(I^* = j | I, \mu_1, \ldots, \mu_J) = \frac{n_j p(x^* | \mu_j, \Sigma_j)}{\alpha G_0(x^*) + \sum_{j=1}^J n_j p(x^* | \mu_j, \Sigma_j)}
\]

where \(I^*\) is the cluster indices vector for the dataset with \(x^*\) excluded and \(n_j\) is the number of samples in the \(j\)-th cluster. A new cluster is created for \(x^*\) with the probability

\[
p(I^* \neq \{1, \ldots, J\} | I, \Theta_0, \alpha) = \frac{\alpha G_0(x^*)}{\alpha G_0(x^*) + \sum_{j=1}^J n_j p(x^* | \mu_j, \Sigma_j)}
\]

The concentration parameter \(\alpha\), which influences the likelihood of generating a new cluster, is kept constant in this paper. After the assignment is updated for every sample in \(X\), empty clusters are removed. This step enables DP to generate new clusters and remove empty clusters so that the number of clusters can be determined automatically.

d. Updating the hyperparameter \(\Theta\) of the NIW distribution: the following Bayesian formulation estimates the posterior distribution of the parameters \(\mu_j\) and \(\Sigma_j\):

\[
p(\mu_j, \Sigma_j | X_j, \Theta_0) \propto p(X_j | \mu_j, \Sigma_j) p(\mu_j, \Sigma_j | \Theta_0)
\]

\[
= p(\mu_j, \Sigma_j | \Theta(j))
\]

where \(X_j\) denotes the samples clustered into the \(j\)-th cluster. \(\Theta_0\) denotes the parameters for the prior NIW distribution \(u_0, \kappa_0, \nu_0\) and \(\Lambda_0\) and \(\Theta(j)\) for the posterior distribution, i.e. \(u(j), \kappa(j), \nu(j)\) and \(\Lambda(j)\). Since the NIW distribution is a conjugate prior for the multivariate Gaussian distribution with unknown means and covariances, \(\Theta(j)\) can be obtained by comparing the posterior distribution in Eq. (10) with the standard NIW formulation, leading to Eq. (11):

\[
u(j) = \nu_0 + n_j
\]

\[
\kappa(j) = \kappa_0 + n_j
\]

\[
\nu(j) = \nu_0 + n_j
\]

\[
\Lambda(j) = \Lambda_0 + \sum_{i=1}^n (x_i(j) - \bar{x}_j) (x_i(j) - \bar{x}_j)^T + \frac{\kappa_0 n_j}{\kappa_0 + n_j} (\bar{x}_j - u_0) (\bar{x}_j - u_0)^T
\]

where \(\bar{x}_j = \frac{1}{n_j} \sum_{i=1}^n x_i(j) / n_j\) and \(n_j\) is the number of samples in \(X_j\).

These steps, except the initialization step, are carried out iteratively until the GMM estimated converges to the true distribution of the data [29]. One sign of convergence is that the number of clusters does not change after several iterations. In practice, usually a maximum number of iterations is set and the iteration stops after reaching this maximum number.

4.1.4. Parameter estimation for Gaussian components

The inference procedure assigns the data to a certain number of clusters and obtains posterior estimates of the mean vectors and covariance matrices of these clusters. However, in order to reduce the influence of the initialization of the NIW distribution, the mean \(\mu_j\) and the covariance matrix \(\Sigma_j\) of the \(j\)-th component are re-estimated from the clustered data [32]:

\[
\hat{\mu}_j = \bar{x}_j
\]

\[
\hat{\Sigma}_j = \frac{1}{n_j-1} \sum_{i=1}^n (x_i(j) - \bar{x}_j) (x_i(j) - \bar{x}_j)^T.
\]

In summary, DP-GMM determines the number of Gaussian distributed clusters from data, estimates the means and covariance matrices of these clusters, and labels the data. The information obtained by DP-GMM will be used in KPCA-based modelling to train the monitoring model. Fig. 2 and Table 2 present the data clusters and the parameter estimation for the numerical example, respectively, obtained by DP-GMM.
Table 2

| Cluster | Parameter estimations. |
|---------|------------------------|
|         | Mean                   | Covariance |
| 1       | -1.4718                | 0.03299, 0.08990 |
|         | -0.5606                | 0.08909, 0.9146 |
| 2       | 0.0361                 | 0.06460, -0.05680 |
|         | 0.3584                 | -0.01686, 0.02530 |
| 3       | 1.2727                 | 0.04899, -0.01318 |
|         | 1.2532                 | -0.01318, 0.09790 |
| 4       | 0.1822                 | 0.04199, 0.02000 |
|         | 1.0216                 | 0.02000, 0.02710 |

4.2. KPCA Using the non-stationary discrete convolution kernel

Kernel Principal Component Analysis (KPCA) projects the data in the original variable space to a kernel space where the nonlinearity in the data can be captured. Assuming the training dataset with N samples is \( X = \{ x_1, \ldots, x_N \} \in \mathbb{R}^{m \times N} \), the kernel function \( k \) of two samples \( x_i \) and \( x_j \) calculates the \( i, j \)-th entry of the kernel matrix \( K \), where \( K \in \mathbb{R}^{N \times N} \) is assumed to be the covariance matrix of the projected kernel features in the kernel space:

\[ K_{ij} = k(x_i, x_j) \quad \text{for} \quad i, j = 1, \ldots, N. \]

(13)

\( K \) is then centered as

\[ \tilde{K}_{ij} = K_{ij} - \bar{K}_{row} - \bar{K}_{col} + \bar{K} \]

(14)

where \( \bar{K}_{row} \) is the mean of the i-th row, \( \bar{K}_{col} \) is the mean of the j-th column, and \( \bar{K} \) is the mean of the entire kernel matrix \( K \). Similarly to the standard PCA formulation, eigenvalue decomposition is applied to \( \tilde{K} \) to obtain the eigenvalues \( \lambda_1, \lambda_2, \ldots, \lambda_N \) and the eigenvectors \( v_1, \ldots, v_N \in \mathbb{R}^{n \times 1} \).

\[ N \lambda v = \tilde{K} v \]

(15)

Assuming that the first \( N_i \) kernel Principal Components (PCs) are retained after KPCA, the kernel principal vector of the i-th sample \( (y_i) \) is calculated as:

\[ y_i = [v_i^T \tilde{K}_{col}, \ldots, v_N^T \tilde{K}_{col}] \in \mathbb{R}^{N_i \times 1} \]

(16)

where \( \tilde{K}_{col} \) is the i-th column of the centered kernel matrix \( \tilde{K} \).

More details of the KPCA formulation can be found in [33]. It can be seen that the kernel PCs \( y \) and the KPCA model depends on the kernel function \( k \) and, as a result, \( k \) should be selected such that it can adequately describe the multimodal data. The proposed framework uses the KPCA with the novel NSDC kernel to build a single monitoring model for multimode process data.

4.2.1. Non-stationary discrete convolution kernel

This section briefly reviews the formulation of the NSDC kernel [12] which aims to build a monitoring model which can account for any multimodality in a dataset. Both the RBF kernel and the NSDC kernel assume the following nonlinear regression problem with basis functions:

\[ z = \sum_{h=1}^{H} w_h f_h(x; \theta_h) \]

(17)

where \( x \in \mathbb{R}^{m \times 1} \) is the vector with original measured variables and \( z \) is the output in a nonlinear feature space. There are \( H \) basis functions, of which \( f_h \) is the \( h \)-th basis function with parameters \( \theta_h \) and a weight \( w_h \). Assuming two samples \( x \) and \( x^* \) have the outputs \( z \) and \( z^* \), respectively, the covariance of \( z \) and \( z^* \) is:

\[ \text{cov}(z, z^*) = k(x, x^*) = \sigma_w^2 \sum_{h=1}^{H} [f_h(x; \theta_h) f_h(x^*; \theta_h)] \]

(18)

given the weights \( w_h \) are independently and identically Gaussian distributed with zero mean and variance \( \sigma_w^2 \).

Eq. (18) becomes the RBF kernel function if the following basis radial basis function is taken as \( f \), the number of basis functions \( H \rightarrow \infty \), the parameters \( \theta_h \) include the centers of the basis functions \( c \) and the kernel width \( \delta \), and \( c \) are distributed from \( -\infty \) to \( \infty \) [34]:

\[ f(x, \theta) = f(x, c, \delta) = \exp \left( -\frac{(x - c)^T \Sigma^{-1} (x - c)}{\delta^2 / 2} \right) \]

(19)

where \( \Sigma \) is the covariance matrix of \( x \) and \( \delta \) controls the performance of the kernel function.

In contrast, in the NSDC kernel only samples in the training set \( X \) are chosen as centers of the radial basis functions. Given that \( c \in X \) and \( X \) has been clustered into \( J \) clusters by DP-GMM, the kernel function becomes:

\[ k(x, x^*) = \sigma_w^2 \sum_{j=1}^{J} \left[ \exp \left( -\frac{(x - x^*)^T \Sigma_j^{-1} (x - x^*)}{\delta} \right) \right. \]

\[ \times \left. \sum_{i=1}^{N_j} \exp \left( -\frac{4}{\delta} (c^{(i)} - x^* - x)^T \Sigma_j^{-1} (c^{(i)} - x^* - x) \right) \right] \]

(20)

where \( N_j \) is the number of samples in the \( j \)-th cluster and \( c^{(i)} \in X \). The NSDC kernel uses the covariance matrix of the j-th cluster estimated by DP-GMM, \( \Sigma_j \), to incorporate the localized property of each operating mode. In NSDC-KPCA, Eq. (13) uses the kernel function defined in Eq. (20).

Although the original formulation of the NSDC kernel is supervised because the data clusters \( X \) and their covariance matrices need to be known a-priori [12], the inclusion of the DP-GMM step into the framework renders the approach unsupervised as the training data is clustered and the parameters are estimated automatically.

4.2.2. Anomaly detection

When applying KPCA to anomaly detection, the anomaly indicators are calculated using the kernel PCs. An anomaly indicator is expected to change monotonically as the level of anomaly increases. For multimodal process monitoring, the same magnitude of anomaly indicators should imply the same level of anomaly for different operating modes.

\( T^2 \) and SPE are examples of anomaly indicators which are widely used in MSPM. For a sample \( x^* \), \( T^2 \) is defined as:

\[ T^2 = y_i^T \Lambda w_i y_i \]

(21)

where \( y_i \in \mathbb{R}^{N_i \times 1} \) are the kernel PCs of \( x^* \) obtained by KPCA, \( \Lambda w_i \) is an \( N_i \times N_i \) diagonal matrix with the first \( N_i \) eigenvalues being its diagonal elements. Under the KPCA formulation the SPE is usually defined as [35]:

\[ \text{SPE}_i = \| \Phi(x^*) \|_2 - 2 \Phi^T(x^*) \cdot \Phi_0 + \| \Phi_0 \|_2 - y_i^T y_i \]

\[ = K_{xx} - 2 \tilde{K}_{xx} - \tilde{K} y_i^T y_i \]

(22)

where \( x^* \) is the centered data, \( \Phi(x^*) \) are the unknown projections of \( x^* \) to the kernel space before applying PCA-based dimension reduction, and \( \Phi_0 \) is the center of \( \Phi(x^*_k) \) for \( k = 1, \ldots, N \). Although \( \Phi(x^*_k) \) and \( \Phi_0 \) cannot be obtained directly, their second order norms can be calculated using the kernel matrix. \( K_{xx} \) is the variance of \( x^*_k \) in the kernel space and \( \tilde{K} \) is the mean of the kernel vector \( K = \{ k(x_1, x_1), \ldots, k(x_N, x_N) \} \). It should be noted that, unlike the RBF kernel, \( k(x, x^*_k) \) is not necessarily equal to 1 when using the NSDC kernel.

SPE may be superior to \( T^2 \) as an anomaly indicator when the RBF kernel is used since \( \Phi(x^*_k) \) has infinite dimensions while \( y_i \) has only finite dimension. As a result, SPE increases monotonically as \( x^* \) gradually deviates away from the training data [36]. However, when using NSDC-KPCA neither \( T^2 \) or SPE are ideal for...
anomaly detection as the indicators do not increase monotonically with the level of anomaly. More specifically, multiple samples that are anomalous to different extents may have the same value of $T^2$ or SPE. For the numerical example in Fig. 3(a), although the two samples highlighted (red asterisks) are different because one is located within the training data cluster while the other is far away from the training data and is likely to be an anomaly, they have the same $T^2$ value. The shaded region defined by the contour is not suitable for anomaly detection because false alarms may be triggered when test samples are inside the training cluster. The same behaviour may be observed when considering SPE as shown in Fig. 3(b).

The reason is that, in the NSDC formulation, the dimension of $\Phi(\tilde{x})$ is also finite ($N \times 1$) due to the number of basis functions in Eq. (18) being finite. As a result it is insufficient to only monitor the model-data mismatch in the NSDC-KPCA model. Therefore, an anomaly indicator should be able to monitor the variations that cause violation of the NSDC-KPCA model and the variations that cannot be modelled by the $N$ basis functions in the NSDC-KPCA model.

A revised SPE is proposed to address this issue. The revised SPE is defined as the sum of the SPE defined in Eq. (22), which quantifies the violation of the NSDC-KPCA model, and an additional term $\|\Phi_{N+1:\infty}(\tilde{x}^*)\|_2$ that includes the variations which are not modelled by NSDC-KPCA:

$$SPE_\ast = \|\Phi_{1:N}(\tilde{x}^*)\|_2 - 2\Phi_{1:N}^\top(\tilde{x}^*) \cdot \Phi_0 + \|\Phi_0\|_2$$

$$= 1 - 2\Phi^\top(\tilde{x}^*) \cdot \Phi_0 + \|\Phi_0\|_2 - y^\top\cdot y^\ast$$

$$= 1 - 2\tilde{K} + \delta - y^\top\cdot y^\ast$$

(23)

given that in the RBF kernel

$$k_{RBF}^{\ast}(x^\ast, x^\ast) = \|\Phi_{1:1}(\tilde{x}^\ast)\|_2^2 = \|\Phi_{1:1}(\tilde{x}^\ast)\|_2^2 + \|\Phi_{N+1:\infty}(\tilde{x}^\ast)\|_2 = 1.$$  

(24)

$x^\ast$ is detected as an anomaly if the following detection criterion holds:

$$SPE_\ast > SPE_{UCL}$$

(25)

where $SPE_{UCL}$ is the upper control limit of the revised SPE estimated from the training data. The distribution of SPE in KPCA does not have an analytic form because of the kernel transformation. In practice, the $SPE_{UCL}$ can be defined by the use of the empirical reference distribution of SPE values for healthy training data [37]. This paper will use the $(100 - \eta)\%$ percentile of the SPEs obtained in the training data as the $SPE_{UCL}$ with confidence level $\eta\%$.

4.2.3. Tuning the NSDC kernel

The kernel width $\delta$ of the NSDC kernel influences the performance of the NSDC-KPCA approach. One of the common choices for the kernel width of RBF kernels is $\delta = rm$ [38,39] where $m$ is the dimension of the input variable space and $r$ is a scaling factor. However, [36] showed that the proper kernel width selection for RBF kernels may also be related to the distances between samples in the training dataset.

This paper applies the tuning strategy proposed in [36]. The strategy examines several values of the kernel width $\delta$ to train the monitoring model. The false alarm rates are calculated by applying the trained monitoring models to a cross-validation dataset generated using the model presented at the beginning of Section 4. Fig. 4 plots the false alarm rates achieved for various $\delta$ values. The optimal $\delta$ value is considered as 5.4 because $\delta = 5.4$ can achieve a reasonably low false alarm rate that is close to 1%. Details of this strategy are discussed in [36].

Fig. 5 compares the monitoring models built by NSDC-KPCA with various values of the kernel width by visualizing the detection contours. The detection contours for the numerical example are achieved by connecting the samples in the variable space with SPE values equal to $SPE_{UCL}$. Any sample located outside the detection contour will be detected as an anomaly. The contour achieved when $\delta = 0.01$ in Fig. 5(a) is over-fitted and will lead to increased false alarms. The contour achieved when $\delta = 1000$ in Fig. 5(b) is under-fitted and cannot describe the dataset sufficiently, leading to increased missed alarms. The optimal kernel width $\delta = 5.4$ results in the contour in Fig. 5(c), which suits the multimode data well.
5. The monitoring framework

Fig. 6 summarizes this framework. In the off-line training step, the DP-GMM approach clusters the unlabelled training data from multiple healthy operating modes. The labelled data, the number of clusters and the means and covariances of each cluster are used for tuning the kernel width for the NSDC kernel and then for training the KPCA model. The NSDC kernel in Eq. (20) calculates the kernel matrix from the training data, where the labelled data, the tuned kernel width $\delta$, the number of the clusters, and the covariances of each cluster are needed. After the PCA dimension reduction in the kernel space, the monitoring model and the upper control limit of SPE are obtained.

In the on-line monitoring step, the monitoring model is applied to the on-line data to obtain the SPE without clustering. If the SPE does not exceed its control limit, the process is considered to operate in a healthy mode. If the SPE exceeds the control limit, an anomaly is detected, meaning that the process has moved to an operating mode that has not been seen in the training data. One needs to refer to expert knowledge, obtained for example by consulting with the process operator, to determine if this mode is a new healthy operating mode of the process. If a new mode has been confirmed, the data from this new mode will be included in the training data. As shown in Fig. 6, the mode update step incorporates the new data which are confirmed to be from the new operating mode. The DP-NSDC-KPCA approach will again cluster the data, identify the clusters in the data, and train the monitoring model in the model update step. Otherwise if there is no evidence that the process has moved to a new healthy mode, the framework will conclude that there is a fault in the process.

A question arises about the simultaneous occurrence of a fault and a mode change. The assumption in the model update step is that it is unlikely for operators to switch the process to a new healthy mode if there is already a fault detected in the process. Therefore, if the expert knowledge confirms the new operating mode, the data collected from the initial period of this mode are assumed to be healthy and can be incorporated in the monitoring model.

6. Examples

6.1. Numerical simulation

6.1.1. Simulation model

The illustrative numerical example presented in Section 4 is used for validating the performance of DP-NSDC-KPCA. In addition to the training data, Fig. 7 also visualizes the two test sequences used for testing the anomaly detection performance. Test sequence 1 (marked with yellow “*” in Fig. 7) has data samples with noise moving from Mode 1 to Mode 4. Test sequence 2 (marked with purple “×” in Fig. 7) has data samples moving from Mode 1 via Mode 2 to Mode 3, which is a nonlinear mode. For clarity, in Fig. 7 every 5-th sample of Test sequences 1–2 are plotted. The monitoring models are trained using the RBF-KPCA and the DP-NSDC-KPCA approach, respectively, and are then used for anomaly detection, following the procedure presented in Fig. 6. For each monitoring model, the 95% percentile of the SPEs of the training data is selected as the upper control limit $UCL$. Since there is no new mode assumed in the test sequences, the model update step is not tested.
6.1.2. Performance

Fig. 8 presents the monitoring contours achieved by the RBF-KPCA and DP-NSDC-KPCA with various values of the kernel width $\delta$. The monitoring contour is obtained by connecting the points in the variable space which have SPE values equal to $\text{SPE}_{\text{UCL}}$. Therefore, samples located outside the contour will be detected as anomalous and samples inside the contour will be labelled as healthy.

Two samples, $S_1$ and $S_2$, are used to illustrate the performance of the monitoring contours. $S_1$ is located among the samples in Mode 1, indicating that it is likely to be a healthy sample, while $S_2$ is located far away from any known modes indicating that it is likely to be an anomaly. Fig. 8(a) shows that the monitoring contours achieved by RBF-KPCA are over-fitted when $\delta = \{0.2, 0.5, 1\}$, especially for Mode 1. In those cases, $S_1$ is always outside the monitoring contours and is incorrectly labelled as anomalous. When $\delta = 5$, the model is under-fitted and the potentially anomalous sample $S_2$ is labelled as healthy. The result shows that RBF-KPCA may not be able to adequately describe the multimodal data regardless of the tuning of its kernel width. On the other hand, the DP-NSDC-KPCA method performs better than RBF-KPCA because $S_1$ is inside the monitoring contours and $S_2$ is outside the contours when $\delta = \{2, 5, 8\}$ in Fig. 8(b). This indicates that the DP-NSDC-KPCA method can achieve monitoring models which are sensitive to anomalies whilst being robust to the small variations in healthy samples.

The anomaly detection performance of the proposed approach for the two test sequences is shown in Figs 9 and 10. The ker-
nel width $\delta$ used for NSDC-KPCA is 5 and the confidence level of $\text{SPE}_\text{UCL}$ is 1%. In order to relate the results in the trend plots of SPE to the scatter plots of the data, several samples are marked. The detection contours in Figs 9(b) and 10(b) are obtained by the $\text{SPE}_\text{UCL}$s in Figs 9(a) and 10(a). In Fig. 9(b), the samples between $S_{11}$ and $S_{12}$ do not belong to any known modes, while the SPE exceeds its control limit in Fig. 9(a), resulting in these samples being identified as anomalous. Similar behaviour is observed for the samples after $S_{13}$. On the other hand, it can be seen that the rest of the samples in the test sequence, e.g. the samples between $S_{12}$ and $S_{13}$ belong to known modes and that the associated SPE does not exceed its control limit. For Test sequence 2, the samples between $S_{21}$ and $S_{22}$, the samples between $S_{2,3}$ and $S_{2,4}$, and the samples after $S_{2,5}$ are identified as anomalous.

When compared with RBF-KPCA, NSDC-KPCA can build better monitoring models and can generate better monitoring contours for multimodal data. The results also demonstrate that the magnitude of the SPE obtained using NSDC-KPCA is sensitive to the mode changes and the same control limit can be used for anomaly detection in several operating modes. In addition, the SPE increases monotonically as the test sample deviates from the known modes. This behaviour may also be useful for estimating the development of a fault.

6.2. Experimental case study

6.2.1. Process description

The PRONTO benchmark dataset [40] is used to emulate the online implementation environment for the proposed approach. This dataset is collected from a pilot-scale multiphase flow facility with several operating modes specified by the water and air flow rates. Artificial faults were induced in various operating modes and data-driven monitoring algorithms can be tested using the data collected in the faulty scenarios. A detailed description of the experiments and dataset is given in [41]. In order to validate the proposed framework, the data from four different operating modes are used. Table 3 presents the air and water flow rates for the four modes. Since the proposed framework is for static process monitoring without considering the temporal correlation in the data, it is possible to re-arrange the data from each mode to simulate the mode switch that may be observed in a multimode process.

In this paper 11 variables are used for monitoring, which are the same as those presented in Table 1 in [25]. Figs 11 and 12 plot the data used for training and testing, respectively. The training set only contains data from Mode 1 and Mode 2 while the test set has data from all four modes shown in Table 3 of which the modes New 1 and New 2 are not included in the off-line training step. Moreover, a developing air blockage fault was introduced by gradually reducing the valve opening of the air inlet pipeline in the mode New 1. Fig. 13 shows the sequence for the mode change and the valve opening for fault simulation of the test set. Both the training and the testing sets mainly consist of steady state data from multiple operating modes. Short transition periods may exist between the modes. It may be observed from Figs 11 and 12 that the steady states of process variables are different in the various operating modes. Therefore, the data sets are suitable for validating the proposed monitoring framework.

6.2.2. Performance

The fault detection performance of the proposed framework is compared against the performance achieved by the monitoring model trained off-line using NSDC-KPCA without model update. The off-line monitoring result shows that, although the monitoring model is sensitive to anomalies caused by mode changes ($A_1$ and $A_2$ in Fig. 14(a)) and is robust to the variations within the known modes in the training data ($R_1$ in Fig. 14(a)), there is no way to incorporate the data from the new operating mode. As a result the monitoring model is unable to distinguish between the new
healthy operating modes and a fault in one of the new modes. The proposed framework also detects anomalies when the process has moved to a new operating mode. However, the model update step includes the data collected from a new mode when the new mode is confirmed to be healthy operation ($U_1$ and $U_2$ in Fig. 14). DP-GMM can automatically determine the number of operating modes in the data, as shown in Fig. 14(b). The fault induced in the new mode can be detected as it develops ($D_1$ in Fig. 14) even if this new mode did not exist in the dataset used for off-line training. Fig. 14 also shows small periods when false alarms are triggered, such as the samples 1050 between 1090 and the samples between 1360 and 1410. For the first period, the false alarms are triggered by the oscillation in the data due to controller overshoot when the set points changed. The overshoot can be observed in Fig. 12. The second false alarm occurs because the model update step $U_2$ uses the data when the process has switched to New 2 but did not reach the steady state (sample 1224–1324 in Fig. 12). The new model does not fully apply to the period afterwards when the process has stabilized. Since transitions between modes are not considered in the framework, it is recommended to use the data collected from steady states for model update. In on-line implementation, false alarms caused by transition periods may be suppressed by alarm management techniques.

It may also be noticed that, after model update, there is no change in the SPE$_{UCL}$ (red-dashed line) in Fig. 14. Moreover, the magnitudes of SPEs obtained in multiple healthy operating modes are similar. This is an important result because the usage of NSDC-KPCA can achieve a single model for multiple modes where the same SPE value indicates the same level of anomaly for different modes. The SPE will facilitate the decision-making procedure of process operators because they can inspect the SPE and determine if there is a fault in the process without the need to specify the current operating mode. Hence this example demonstrates that the proposed framework has achieved an anomaly indicator which is useful for process operators.

7. Discussion

The DP-NSDC-KPCA approach used for building monitoring models in this framework is motivated by practical considerations
when designing and applying monitoring methods. Historical data from process operations are usually unlabelled and the operating modes are often unknown. Since DP-based clustering can automatically determine the number of clusters, the off-line training procedure is made unsupervised and the on-line model update can easily incorporate new data from on-line operations. The clustering method assumed that data samples from operating modes do not overlap significantly with each other. If they do overlap, then the clustering performance may not be precise. Moreover, the transition periods between modes may be identified by DP-based clustering as individual modes.

In order to guarantee the process is operating efficiently and to minimize the likelihood of faults developing into failures, the decision of process operation and maintenance should be made according to the existence of faults and the fault severity in the process. Therefore, as stated in Section 4.2.2, an anomaly indicator is desired if it exceeds the control limit and detects the fault when a fault occurs and changes monotonically as the fault severity develops. In the framework, the fault detection performance is improved by appropriate tuning of the NSDC kernel. The revised SPE proposed in Section 4.2.2 aims to fulfill the requirement of monotonicity. Additionally, the decision step will be made easier if the same magnitude of the anomaly indicator indicates the same level of fault severity in various modes. The SPE trend in Fig. 14 demonstrates that the DP-NSDC-KPCA approach, which builds a single monitoring model for multiple modes, fulfills this expectation.

Such behaviour of anomaly indicators achieved by DP-NSDC-KPCA approach will facilitate fault diagnosis and prognosis. For example, contribution plot-based methods can pin-point the location and the cause of a fault by identifying process variables that have large contributions to the anomaly indicator. The magnitude of variable contributions can only be compared if the magnitude of an anomaly indicator is the same for multiple operating modes. As for fault prognosis, the trend of the anomaly indicator can reflect fault development only if the same magnitude of the anomaly indicator indicates the same level of fault severity in various modes.

A future research direction will be to extend this framework to dynamic process monitoring. The dynamic extension requires the description of the process dynamics, to which auto-regressive models or state-space models may apply. For example, [42] provided a way of modelling nonlinear and dynamic processes. The extension also requires the dynamic adaptation of the DP-NSDC-KPCA framework with respect to the dynamic process models. The aim will be to cope with process dynamics and transition periods between modes whilst maintaining the unambiguous behaviour of the anomaly indicators achieved by the monitoring framework.

8. Conclusions

This paper presented an on-line monitoring framework for multimode and nonlinear processes. The unsupervised training step uses the DP-GMM clustering approach to cluster the training data from multiple operating modes and implements the NSDC-KPCA to build a single monitoring model. The tuning of the kernel width of the NSDC kernel has been considered and the SPE was chosen as the anomaly indicator. This framework has the following advantages. The monitoring result can be easily interpreted because the monitoring model and the anomaly indicator are robust to mode changes. At the same time the monitoring model is sensitive to anomalies that deviate from the known modes. Data from new op-
erating modes observed during process operations can be incorpo-
rated to update the monitoring model, making it possible to detect
faults in the new operating modes.

Declaration of Competing Interest

The authors declare that they have no known competing financial
interests or personal relationships that could have appeared to
influence the work reported in this paper.

CRediT authorship contribution statement

RuoMu Tan: Conceptualization, Methodology, Data curation,
Software, Visualization, Writing - original draft. Tian Cong:
Methodology, Software, Visualization, Writing - original draft. James R. Ottewill: Conceptualization, Writing - review & edit-
ing, Supervision. Jerzy Baranowski: Supervision. Nina F. Thornhill:
Conceptualization, Writing - review & editing, Supervision, Funding
acquisition.

Acknowledgments

This work was supported by the European Union’s Horizon 2020
research and innovation programme under grant agreement
No. 675215 - PRONTO - H2020-MSCA-ITN-2015.

References

[1] B. Song, H. Shi, Temporal-spatial global locality projections for multimode pro-
cess monitoring, IEEE Access 6 (2018) 9740–9749.
[2] C. Tong, A. Palazoglu, X. Yan, An adaptive multimode process monitoring strat-
ey based on mode clustering and mode unfolding, J. Process Control 23 (10)
(2013) 1497–1507.
[3] M. Quiones-Grueiro, A. Prieto-Moreno, C. Verde, O. Llanes-Santiago, Data-
driven monitoring of multimode continuous processes: a review, Chemometr.
Intell. Lab. Syst. 189 (2019) 56–71.
[4] F. Zhang, Z. Ge, Decision fusion systems for fault detection and identification
in industrial processes, J. Process Control 31 (2015) 45–54.
[5] J. Zheng, Z. Song, Linear subspace principal component regression model for
quality estimation of nonlinear and multimode industrial processes, Ind. Eng.
Chem. Res. 56 (21) (2017) 6275–6285.
[6] L. Zhou, J. Zheng, Z. Ge, Z. Song, S. Shan, Multimode process monitoring based
on switching autoregressive dynamic latent variable model, IEEE Trans. Ind.
Electron. 65 (10) (2018) 8184–8194.
[7] Z. Ge, F. Gao, Z. Song, Batch process monitoring based on support vector data
description method, J. Process Control 21 (5) (2011) 949–959.
[8] C. Zhang, X. Gao, T. Xu, Y. Li, Nearest neighbor difference rule-based kernel
principal component analysis for fault detection in semiconductor manufactur-
ing processes, J. Chemom. 31 (6) (2017) e2888.
[9] X. Peng, Y. Tang, W. Du, F. Qian, Online performance monitoring and modeling
paradigm based on just-in-time learning and extreme learning machine for a
non-Gaussian chemical process, Ind. Engineering Chemistry Research 56 (23)
(2017) 6671–6684.
[10] X. Deng, N. Zhong, L. Wang, Nonlinear multimode industrial process fault de-
tection using modified kernel principal component analysis, IEEE Trans. Ind.
Electron. 5 (2017) 23121–23132.
[11] J. Yu, J. Yao, J. Jiang, J.H. Park, S. Kim, A novel hybrid of auto-associative kernel
regression and dynamic independent component analysis for fault detection in
nonlinear multimode processes, J. Process Control 68 (2018) 129–144.
[12] R. Tan, J.R. Ottewill, N.F. Thornhill, Non-stationary discrete convolution kernel
for multimodal process monitoring, IEEE Trans. Neural Netw. Learn. Syst. 2019
1–12, doi:10.1109/TNNLS.2019.2945847.
[13] X. Deng, X. Tian, S. Chen, C.J. Harris, Nonlinear process fault diagnosis based
on serial principal component analysis, IEEE Trans. Neural Netw. Learn. Syst.
29 (3) (2016) 560–572.