Computers and Chemical Engineering, Volume 123, April 2019, pp. 143-154
DOI:10.1016/j.compchemeng.2018.12.027

Mixed Kernel Canonical Variate Dissimilarity Analysis for Incipient Fault Monitoring in Nonlinear Dynamic Processes

Karl Ezra S. Pilario[a,b,*], Yi Cao[c], Mahmood Shafiee[a]

[a] *Department of Energy and Power, Cranfield University, Bedfordshire MK43 0AL, United Kingdom*
[b] *Department of Chemical Engineering, University of the Philippines Diliman, Republic of the Philippines*
[c] *College of Chemical and Biological Engineering, Zhejiang University, People's Republic of China*

Abstract

Incipient fault monitoring is becoming very important in large industrial plants, as the early detection of incipient faults can help avoid major plant failures. Recently, Canonical Variate Dissimilarity Analysis (CVDA) has been shown to be an efficient technique for incipient fault detection, especially under dynamic process conditions. CVDA can be extended to nonlinear processes by introducing kernel-based learning. Incipient fault monitoring requires kernels with both good interpolation and extrapolation abilities. However, conventional single kernels only exhibit one ability or the other, but not both. To overcome this drawback, this study presents a Mixed Kernel CVDA method for incipient fault monitoring in nonlinear dynamic processes. Due to the use of mixed kernels, both enhanced detection sensitivity and a better depiction of the growing fault severity in the monitoring charts are achieved. Looking ahead, this work takes a step towards understanding the impact of kernel behavior in process monitoring performance.

Keywords: Fault detection, canonical variate analysis, global kernel, local kernel, kernel density estimation

*Corresponding author
Email address: k.pilario@cranfield.ac.uk (Karl Ezra S. Pilario)

October 18, 2018

Published by Elsevier. This is the Author Accepted Manuscript issued with: Creative Commons Attribution Non-Commercial No Derivatives License (CC:BY:NC:ND 4.0).
The final published version (version of record) is available online at DOI:10.1016/j.compchemeng.2018.12.027. Please refer to any applicable publisher terms of use.

1. Introduction

Industrial plants are becoming increasingly complex with more and more inter-dependent subsystems, control units, and machines (Shafiee, 2016). Hence, the necessary task of process health monitoring becomes more challenging (Chiang et al., 2005). Fortunately, with the rise of new technologies in automation and data acquisition, large data sets from these plants are readily available (Yin et al., 2015). By taking advantage of this, Multivariate Statistical Process Monitoring (MSPM) methods are deemed most favorable for monitoring complex industrial processes (Zhang and Zhang, 2010). Since process variables are highly correlated, MSPM methods are usually dimensionality reduction tools (Chiang et al., 2005) such as principal components analysis (PCA), partial least squares (PLS), independent component analysis (ICA), canonical correlation analysis (CCA), and canonical variate analysis (CVA). Data-driven methods are attractive because their use avoids the costly and time-consuming process of first-principles modelling for distinguishing between normal and faulty process operating conditions (Yin et al., 2015; Ge et al., 2013).

The key issues in MSPM are outlined by Ge et al. (2013). Plant data was described to be nonlinear, non-Gaussian, and dynamic in nature. Hence, through the decades, the MSPM methods are continuously being enhanced for *nonlinear dynamic process* monitoring. In addition, due to the large scale of process industries, distributed modelling frameworks were also devised in recent literature (Ge, 2017; Ge and Chen, 2016).

Aside from these, incipient fault monitoring is still a fundamental issue and is recently gaining research attention. As opposed to abrupt faults, incipient faults are characterized as those that: (i) have a small magnitude at the initial stage; and, (ii) slowly drift (or increase in magnitude) as the process degrades in time (Isermann, 2005; Pilario and Cao, 2017). If not detected early, these faults can lead to an emergency situation or catastrophic failure (Vachtsevanos et al., 2006). Yet early detection is difficult, especially in closed-loop systems where the fault is initially masked by process control, and by noise or disturbances (Zhang et al., 2002).

To address this, nonlinear dynamic MSPM methods with enhanced sensitivity were recently proposed. Shang et al. (2018) used an augmented kernel Mahalanobis distance metric for improved fault detection, which avoids space partitioning in PCA. This produced a more sensitive detection index than CVA and PCA variants when tested in the Tennessee Eastman Plant. Mean-

while, Rato and Reis (2014) proposed sensitivity enhancing transformations, which also used augmented data for accounting dynamics and nonlinearities. Cheng et al. (2010) and Ji et al. (2018) used the multivariate exponentially weighted moving average for capturing small mean shifts in the process. Recently, Pilario and Cao (2018) proposed Canonical Variate Dissimilarity Analysis (CVDA) to detect incipient faults even at dynamically varying process operating conditions. However, the nonlinear issue needs to be handled more efficiently in CVDA, since processes are inherently nonlinear in practice. One way to address this is to represent the system in a set of multiple local linear models, such as the recent application of locality preserving projections (LPP) to CVA by Lu et al. (2018) and the mixture variational Bayesian CCA by Liu et al. (2018b). Alternatively, kernel-based learning can be introduced in CVDA for nonlinear pattern discovery.

Kernel methods are currently being used to handle the nonlinear issue with promising results. In kernel methods, the idea is to project the data onto a high-dimensional space using kernel functions, so that linear MSPM can be applied to the transformed data. Ever since Schölkopf et al. (1998) laid the foundations of kernel PCA, several other kernel MSPM methods have been reported in the literature. Recent works include the kernel dynamic PCA by Fezai et al. (2018) and Jaffel et al. (2016), the enhanced kernel PCA by Nguyen and Golinval (2010), the kernel PLS based generalized likelihood ratio test by Botre et al. (2016), the kernel dynamic ICA by Fan and Wang (2014), the weighted kernel ICA for non-Gaussian data by Cai et al. (2017), the kernel CVA by Samuel and Cao (2015), and the dynamic concurrent kernel CCA by Liu et al. (2018a). Fault diagnosis using kernels applied to support vector machines (SVM) was also explored in numerous works, as surveyed by Yin and Hou (2016). For example, Zhang (2009) used kernel PCA and kernel ICA features as input to SVM for classifying faults. The most widely used kernel function in these studies, e.g. Cheng et al. (2010); Nguyen and Golinval (2010); Fan and Wang (2014); Samuel and Cao (2015); Bernal-de Lázaro et al. (2016); Liu et al. (2018a), is the Gaussian radial basis function, which we refer to as the RBF kernel for the rest of this paper. Other choices include the polynomial and sigmoid kernels, to name a few.

Indeed, the impact of the choice of kernel to process monitoring performance is still not clear (Zhang, 2009; Bernal-de Lázaro et al., 2016). A first step towards addressing this issue is to explore the behavior of typical kernel functions individually. Some existing works such as Jia et al. (2012) and Shao et al. (2009) have provided results towards understanding this issue for

kernel PCA based monitoring. Our current work is motivated by this same premise but in the context of the dynamic process monitoring of incipient faults.

In this paper, we first highlight some drawbacks in using the RBF kernel or any single kernel on their own for the predictive monitoring of incipient faults. Also, since CVDA is recognized as a dynamic MSPM method that is sensitive to incipient faults, we extend its applicability to nonlinear processes using kernel methods. As a result, a new kernel MSPM method is presented that is called Mixed Kernel Canonical Variate Dissimilarity Analysis (MK-CVDA). The overall method consists of a kernel PCA (KPCA) followed by CVDA. Cross-validation via the grid search method is also suggested as a practical way to find optimal kernel parameters. In MK-CVDA, the same statistical indices from CVDA, namely the T^2, Q, and D, are adopted. The non-Gaussianity issue is handled by using kernel density estimation for computing the detection limits of these indices. The new method is intended for monitoring slowly developing faults in nonlinear dynamic processes under varying operating conditions.

The structure of the paper is as follows. KPCA is first revisited in Section 2. Afterwards, mixed kernels are introduced in Section 3. Section 4 discusses how MK-CVDA proceeds by performing KPCA followed by CVDA. The MK-CVDA method is evaluated in Section 5. Finally, the work is concluded in Section 6 along with some future perspectives.

2. Kernel PCA Revisited

In general, kernel dynamic MSPM methods consist of: (i) data projection to the kernel space; (ii) augmentation of lagged variables to treat dynamics; and (iii) dimensionality reduction for partitioning the data space into the *state* and *residual* subspaces. For MK-CVDA, step (i) is done using kernel PCA (KPCA) and steps (ii)-(iii) are performed using CVDA. In this section, KPCA is revisited as follows.

Let $\mathbf{x}_k = \begin{bmatrix}\mathbf{u}_k^T & \mathbf{y}_k^T\end{bmatrix}^T \in \mathbb{R}^m, k = 1, \ldots, N$ denote a data set of N observations of m variables, where $\mathbf{u}$ and $\mathbf{y}$ represent the process inputs and outputs, respectively. $\mathbf{x}_k$ is normalized to zero mean and unit variance as $\widehat{\mathbf{x}}_k$.

In PCA, features are extracted only in a linear space. Thus, some nonlinear map $\mathbf{\Phi}(\cdot)$ must first be used to project the data from the nonlinear input space onto a linear feature space F, i.e. $\mathbf{\Phi} : \mathbb{R}^m \rightarrow F$. Assuming that $\sum_{k=1}^{N} \mathbf{\Phi}(\widehat{\mathbf{x}}_k) = 0$, PCA seeks to solve an eigenvalue problem on the sample

covariance in F, as follows:

$$\mathbf{C}^F = \frac{1}{N}\sum_{k=1}^{N} \mathbf{\Phi}(\widehat{\mathbf{x}}_k)\mathbf{\Phi}(\widehat{\mathbf{x}}_k)^T, \tag{1}$$

$$\mathbf{C}^F\mathbf{w} = \lambda\mathbf{w}, \tag{2}$$

where $\mathbf{C}^F$ is the sample covariance in F, $\mathbf{w}$ is an eigenvector, and λ is an eigenvalue.

Many kinds of nonlinear relationships must be accounted to design a $\mathbf{\Phi}(\cdot)$ that models the process accurately, but it may inevitably result in a large dimensionality in F. So to avoid specifying $\mathbf{\Phi}(\cdot)$ explicitly, Schölkopf et al. (1998) suggested to represent dot products in F using kernel functions K for $(i, j) = 1, \ldots, N$ as follows:

$$K(\mathbf{x}_i, \mathbf{x}_j) \triangleq K_{ij} = \langle \mathbf{\Phi}(\mathbf{x}_i), \mathbf{\Phi}(\mathbf{x}_j) \rangle, \tag{3}$$

where $\langle \cdot, \cdot \rangle$ denotes dot product. Similarly, Eq. (2) is replaced by the following set of equations for $k = 1, \ldots, N$:

$$\langle \mathbf{\Phi}(\widehat{\mathbf{x}}_k), \mathbf{C}^F\mathbf{w} \rangle = \lambda \langle \mathbf{\Phi}(\widehat{\mathbf{x}}_k), \mathbf{w} \rangle. \tag{4}$$

Noting that there exists some $\mathbf{v}$ such that $\mathbf{w} = \langle \mathbf{v}, \mathbf{\Phi}(\widehat{\mathbf{x}}_k) \rangle$, the expression in Eq. (4) is then expanded, where all instances of $\langle \mathbf{\Phi}(\mathbf{x}_i), \mathbf{\Phi}(\mathbf{x}_j) \rangle$ are replaced with the kernels in Eq. (3), yielding a different eigenvalue problem:

$$\widehat{\mathbf{K}}\mathbf{v} = N\lambda\mathbf{v}, \tag{5}$$

where $\mathbf{v}$ is an eigenvector, λ is an eigenvalue, $\mathbf{K} \equiv [K_{ij}]$ is an $N \times N$ symmetric *kernel matrix*, and $\widehat{\mathbf{K}}$ is matrix $\mathbf{K}$ mean-centered in F using:

$$\widehat{\mathbf{K}} = \mathbf{K} - \mathbf{1}_N\mathbf{K} - \mathbf{K}\mathbf{1}_N + \mathbf{1}_N\mathbf{K}\mathbf{1}_N, \tag{6}$$

where $\mathbf{1}_N \in \mathbb{R}^{N\times N}$ and $(\mathbf{1}_N)_{ij} = 1/N$.

A form of nonlinear PCA now involves solving Eq. (5) instead of Eq. (2). Thus, the need to specify $\mathbf{\Phi}(\cdot)$ is eliminated since the nonlinear mapping is implicitly achieved by a so-called *kernel trick*. However, as it will be discussed in Section 3, not all functions can be used as kernels.

KPCA proceeds by forming the kernel matrix $\mathbf{K}$ from $\widehat{\mathbf{x}}_k$ using Eq. (3) and centering $\mathbf{K}$ to $\widehat{\mathbf{K}}$ using Eq. (6). Due to Eq. (5), $\widehat{\mathbf{K}}$ is then diagonalized as

$$\widehat{\mathbf{K}}/N = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^T, \tag{7}$$

where $\mathbf{S} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N] \in \mathbb{R}^{N \times N}$ represents N eigenvectors and $\mathbf{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_N) \in \mathbb{R}^{N \times N}$ are eigenvalues where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$. In order to keep only the relevant information, only the top r number of principal components (PCs) that explain 99% of the total variance are retained. Denoting $\mathbf{S}_r \in \mathbb{R}^{N \times r}$ as the first r columns of $\mathbf{S}$, the PCs $\mathbf{t}_k$ are then obtained by using the following projection:

$$\mathbf{T} \equiv [\mathbf{t}_k] = \mathbf{S}_r^T \widehat{\mathbf{K}} \in \mathbb{R}^{r \times N}. \tag{8}$$

Any test data $\mathbf{x}_k^{\text{test}} \in \mathbb{R}^m$ at the kth sampling instant is then normalized using the mean and standard deviation of the training set, which yields $\widehat{\mathbf{x}}_k^{\text{test}}$. The $\widehat{\mathbf{x}}_k^{\text{test}}$ is projected to the feature space F using:

$$\mathbf{k}_k^{\text{test}} = K(\widehat{\mathbf{x}}_k^{\text{test}}, \widehat{\mathbf{x}}_j) \in \mathbb{R}^{1 \times N}, \tag{9}$$

where $\widehat{\mathbf{x}}_j$ represents all training samples $j = 1, \ldots, N$. $\mathbf{k}_k^{\text{test}}$ is then centered as

$$\widehat{\mathbf{k}}_k^{\text{test}} = \mathbf{k}_k^{\text{test}} - \mathbf{1}_N^{\text{test}}\mathbf{K} - \mathbf{k}_k^{\text{test}}\mathbf{1}_N + \mathbf{1}_N^{\text{test}}\mathbf{K}\mathbf{1}_N, \tag{10}$$

where $\mathbf{1}_N^{\text{test}} \in \mathbb{R}^{1 \times N}$ and $(\mathbf{1}_N^{\text{test}})_{ij} = 1/N$. Finally, the kernel PCs of the test data at the kth sampling instant are obtained as

$$\mathbf{t}_k^{\text{test}} = \mathbf{S}_r^T (\widehat{\mathbf{k}}_k^{\text{test}})^T \in \mathbb{R}^r. \tag{11}$$

In KPCA monitoring, the widely used statistical indices are computed as

$$T^2 = (\mathbf{t}^{\text{test}})^T \mathbf{\Lambda}_r^{-1} \mathbf{t}^{\text{test}}, \tag{12}$$

$$Q = \|\mathbf{S}^T (\widehat{\mathbf{k}}^{\text{test}})^T\|^2 - (\mathbf{t}^{\text{test}})^T \mathbf{t}^{\text{test}}, \tag{13}$$

where $\mathbf{\Lambda}_r \in \mathbb{R}^{r \times r}$ is a diagonal matrix of the first r eigenvalues in Eq. (7) (Choi et al., 2005). The T^2 and Q indices monitor the principal subspace $\mathbf{T}$ and the residual subspace, respectively. However, dynamics in the data are not handled by KPCA alone. Hence, the MK-CVDA method takes the PCs $\mathbf{t}_k$ as input features to CVDA. So aside from data projection to the kernel space, KPCA effectively serves as a data whitening step (as did Fan and Wang (2014)), as well as a way to avoid singular matrices in CVDA afterwards (as did Samuel and Cao (2015)).

The overall effect of KPCA followed by CVA is equivalent to the implementation of a direct kernel CVA, as noted by Zhu et al. (2012) and Samuel and Cao (2015). In the kernel CCA utilized by Liu et al. (2018a),

the approach to ensure invertible matrices is by introducing a regularization parameter to the kernel matrix. In our work, the number of retained kernel principal components, r ($< N$), has the same role of a regularization parameter (Zhu et al., 2012).

In summary, KPCA involves the transformation of training data $\mathbf{x}_k \in \Re^m$ into $\mathbf{t}_k \in \mathbb{R}^r$ by nonlinear projection to a feature space F, and further onto a subspace of F so as to perform whitening and regularization in MK-CVDA. In the following section, the choice of kernel in Eq. (3) is discussed.

3. Choice of Kernel

3.1. Local and Global Kernel Behavior

In functional analysis, Mercer's theorem gives conditions for kernel functions that can act as a dot product in a possibly ∞-dimensional space, formally known as a Hilbert space (Cristianini and Shawe-Taylor, 2014). In loose terms, admissible kernels are said to be those that produce a positive semi-definite kernel matrix, $\mathbf{K}$. Although many different functions satisfy this requirement, Jordaan (2002) noted that there are two main types of kernels: *local* and *global*. A typical example of a local kernel is the widely used Gaussian radial basis function (RBF), that is given by:

$$K_g(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{c}\right), \tag{14}$$

where c is the kernel width. It satisfies the Mercer condition for $c > 0$ (Cristianini and Shawe-Taylor, 2014). It also corresponds to an ∞-dimensional space F, because the exponential can be viewed as a polynomial of *infinite* degree, when expressed as a power series. On the other hand, a typical example of a global kernel is the polynomial kernel, given by:

$$K_p(\mathbf{x}, \mathbf{x}') = (\langle \mathbf{x}, \mathbf{x}' \rangle + 1)^d, \tag{15}$$

where d is the kernel parameter that denotes the degree of the polynomial. This kernel satisfies the Mercer condition for $d \in \mathbb{N}$ (Smola et al., 2000). Others have found polynomial kernels more suitable than the RBF kernel for certain applications, e.g. the penicillin process (Jia et al., 2012; Lee et al., 2004).

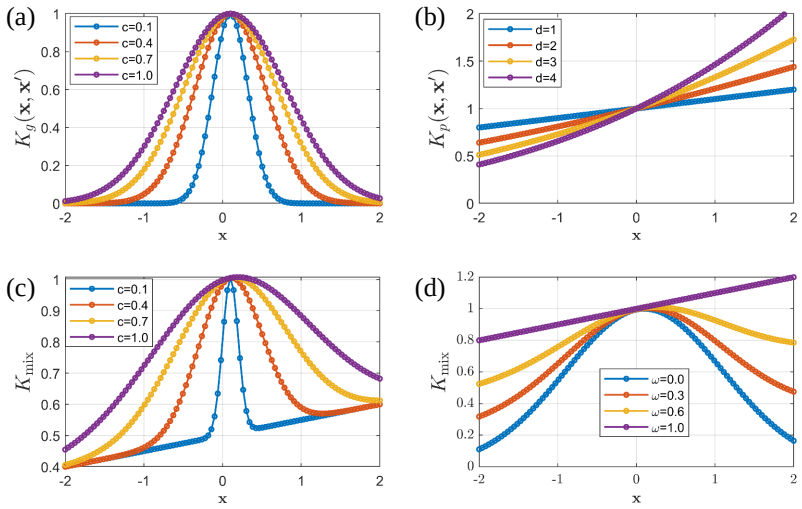

Figure 1: Sample plots of kernel values, where $\mathbf{x}' = 0.1$, for: (a) a local kernel (RBF kernel); (b) a global kernel (polynomial kernel); (c) mixed kernel at fixed $d = 1, \omega = 0.5$; and (d) mixed kernel at fixed $d = 1, c = 2$ (Zhu et al., 2012; Jordaan, 2002).

Sample plots of local and global kernel values are shown in Fig. 1(a,b), where $\mathbf{x}'$ represents a training sample from the normal process and $\mathbf{x}$ represents any unseen test sample to be mapped using K_g or K_p, as in Eq. (14)-(15). Using these plots, the differences and limitations of each type of kernel are discussed empirically as follows.

For the RBF kernel in Fig. 1(a), the behavior of an exponential function is expected: K_g tends to one as the difference between $\mathbf{x}$ and $\mathbf{x}'$ approaches zero, and tends to zero when their difference becomes large. However, the fact that the K_g mapping "vanishes" to $K_g = 0$ as test points move farther from the training data is undesirable in process monitoring. The theoretical implications of this fact are further discussed in Section 3.3. In most studies, much larger kernel widths c are chosen to increase the spread of K_g, e.g. in the Tennessee Eastman Plant, Fan and Wang (2014) used $c = 500m$ (where m is the number of variables) and Samuel and Cao (2015) used a constant $c = 1720$. However, these mappings still vanish beyond a certain distance from the training data. Kernels must be able to learn an effective mapping in the vicinity of the training data (indicating good interpolation ability), and also influence a mapping over the *entire* data space (indicating good ex-

trapolation ability). However, the RBF kernel only exhibits the former (Zhu et al., 2012). Thus, whenever c is said to be chosen arbitrarily, it is actually chosen too large, hoping that the RBF kernel would extrapolate well. As a local kernel, however, the RBF kernel loses its interpolation ability at large values of c. The extent of this occurrence depends on the case study at hand. Hence, local kernels alone cannot exhibit both good interpolation and extrapolation abilities at the same time (Zhu et al., 2012).

On the other hand, Fig. 1(b) shows that the polynomial kernel extrapolates well because it creates a mapping on the entire data space, regardless of where it was trained. However, it only interpolates well at large values of d. Thus, when global kernels are used alone, good extrapolation and interpolation abilities cannot be achieved at the same time either (Zhu et al., 2012).

3.2. Mixed Kernel

In practice, a kernel that has both good interpolation and extrapolation abilities, i.e. good generalization, is desired. In a particular development of soft sensors, Jordaan (2002) proposed the use of a mixture of local and global kernels, which was also proven to satisfy Mercer's condition. In that work, a convex combination of kernels was formed as given by:

$$K_{\text{mix}} = \omega K_p + (1 - \omega) K_g, \tag{16}$$

where $\omega \in [0, 1]$ is the mixture weight. At $d = 1$ in K_p, the effects of varying the kernel width c and weight ω in K_{mix} is shown in Fig. 1(c)-(d). Note that the mixed kernel reduces to the polynomial and RBF kernels at $\omega = 1$ and $\omega = 0$, respectively. Since the work of Jordaan (2002), more studies that used mixed kernels have also been published, for example, see Lian et al. (2013); Yang et al. (2013); Xu et al. (2015); Zhong and Carr (2016); Cheng et al. (2017); Zhong et al. (2018); Chen et al. (2018). One notable work related to CVA is the mixed kernel canonical correlation analysis (MKCCA) by Zhu et al. (2012) which improved dimensionality reduction in various learning tasks. Moreover, the properties of various kernel combinations were studied by Duvenaud (2014) for Gaussian process models. In this paper, the mixed kernel in Eq. (16) is used for nonlinear process monitoring of incipient faults. Having both interpolation and extrapolation abilities, mixed kernels are able to handle both the nonlinear and predictive issues in process monitoring. Owing to the local behavior of the mixed kernel, earlier detection of small-magnitude faults can be achieved. Meanwhile, owing to the global behavior

of the mixed kernel, incipient fault growth can be depicted properly in the monitoring charts as the process degrades in time. Further analysis is given in the next subsection.

According to Jordaan (2002), the weighted sum of a linear ($d = 1$) and RBF kernel is recommended to balance good interpolation and extrapolation abilities. Hence, $d = 1$ is adopted in mixed kernels for the rest of this paper. After the KPCA step in Section 2, the MK-CVDA algorithm description is continued in Section 4, including a discussion on how to choose other parameters in Eq. (16).

3.3. Theoretical Basis for Process Monitoring

The RBF kernel is by far the most popular choice of kernel in process monitoring literature. An important implication of choosing the RBF kernel in KPCA-based process monitoring is given in the following theorem.

Theorem 1. *Let $\widehat{\mathbf{x}}_j \in \mathbb{R}^m$ denote a training data set for $j = 1, 2, \ldots, N$, $\widehat{\mathbf{x}}^* \in \mathbb{R}^m$ denote a fault-free test sample, and $\Delta\mathbf{x} \in \mathbb{R}^m$ denote the amount of shift caused by a fault such that the observed test sample becomes $\widehat{\mathbf{x}}^{\text{test}} = \widehat{\mathbf{x}}^* + \Delta\mathbf{x}$. Upon training KPCA on any fixed data set $\widehat{\mathbf{x}}$ and choosing the RBF kernel with a fixed value of c, the statistical indices, T^2 and Q, approach constant values as $||\Delta\mathbf{x}|| \to \infty$.*

Proof. In Section 3.1, it has been noted that if $\Delta\mathbf{x}$ is sufficiently large, then the observed test sample $\widehat{\mathbf{x}}^{\text{test}}$ will be projected to zero kernel value when the RBF kernel is used in Eq. (9). Mathematically, this can be expressed as:

$$\lim_{||\Delta\mathbf{x}||\to\infty} \mathbf{k}^{\text{test}} = \lim_{||\Delta\mathbf{x}||\to\infty} K_g(\widehat{\mathbf{x}}^* + \Delta\mathbf{x}, \widehat{\mathbf{x}}_j) = \mathbf{0} \quad \forall\, j \tag{17}$$

where $K_g(\cdot\,,\cdot)$ is given in Eq. (14) and $\widehat{\mathbf{x}}_j$ is the jth training sample. Equation (17) is a straightforward consequence of the exponential function, at a fixed value of c. By substituting Eq. (17) into Eq. (10)-(11), we have:

$$\begin{aligned}
\lim_{||\Delta\mathbf{x}||\to\infty} \widehat{\mathbf{k}}^{\text{test}} &= -\mathbf{1}_N^{\text{test}}\mathbf{K} + \mathbf{1}_N^{\text{test}}\mathbf{K}\mathbf{1}_N \\
&= \mathbf{1}_N^{\text{test}}\mathbf{K}(\mathbf{1}_N - \mathbf{I}) \in \mathbb{R}^{1\times N}, \qquad (18)\\
\lim_{||\Delta\mathbf{x}||\to\infty} \mathbf{t}^{\text{test}} &= \mathbf{S}_r^T(\mathbf{1}_N^{\text{test}}\mathbf{K}(\mathbf{1}_N - \mathbf{I}))^T \in \mathbb{R}^r. \qquad (19)
\end{aligned}$$

The statistical indices in KPCA-based monitoring are given in Eq. (12)-(13). Since Eq. (18)-(19) indicate that both $\widehat{\mathbf{k}}^{\text{test}}$ and $\mathbf{t}^{\text{test}}$ approach a vector

of constants as $||\Delta\mathbf{x}|| \to \infty$, then the exact limits of T^2 and Q are also constants, whose values depend on the fixed training data set that enters matrix $\mathbf{K}$ and the fixed value of the RBF kernel width c. ■

Theorem 1 highlights the main drawback of using RBF kernels for nonlinear process monitoring, especially for incipient faults. Since an incipient fault worsens in magnitude with time, its effect at two different points in time, i.e. $(\Delta\mathbf{x})_1$ and $(\Delta\mathbf{x})_2$, may differ significantly. However, if they are both sufficiently large, they may be reflected as equal in the T^2 and Q monitoring charts. Hence, the notion of incipient fault growth cannot be depicted accurately when the RBF kernel is used in KPCA.

Moreover, there is no guarantee that the final values of T^2 and Q as $||\Delta\mathbf{x}|| \to \infty$ would remain above the computed upper control limits, T^2_{UCL} and Q_{UCL}, even as the process continues to degrade under faulty conditions. This occurrence is investigated in Section 5.1.

In contrast to the RBF kernel, the polynomial kernel does not suffer from such an undesirable effect, since it can be shown that:

$$\lim_{||\Delta\mathbf{x}||\to\infty} K_p(\widehat{\mathbf{x}}^* + \Delta\mathbf{x}, \widehat{\mathbf{x}}_j) = \pm\infty \quad \forall\, j \tag{20}$$

where $K_p(\cdot\,,\cdot)$ is given in Eq. (15) and the sign of the limit depends on the polynomial degree d and the direction of $\Delta\mathbf{x}$. Thus, the rising fault magnitude can be depicted more accurately in the T^2 and Q monitoring charts upon choosing the polynomial kernel for KPCA.

However, the polynomial kernel only has limited flexibility to approximate the nonlinearities in the process, as dictated by the degree d (Jordaan, 2002). These drawbacks can be resolved by using mixed kernels for KPCA, which combine the benefits from the RBF and polynomial kernels. Although this approach introduces more kernel parameters to tune in the training phase, the parameters can remain fixed during the online monitoring phase. Thus, the computational load for mixed kernel based KPCA during online monitoring remains the same as that when single kernels are used.

4. Mixed Kernel CVDA

CVDA is a framework based on canonical variate analysis (CVA), which is an effective dynamic MSPM method (Odiowei and Cao, 2010). CVDA aims to enhance CVA for incipient fault detection (Pilario and Cao, 2018). In this

paper, the proposed MK-CVDA consists of KPCA followed by CVDA. After performing KPCA to handle process nonlinearities in Section 2, together with mixed kernels in Section 3, we proceed with CVDA for handling process dynamics as follows.

4.1. CVDA Methodology

In CVDA, data are first arranged into past and future matrix blocks. However, only the process output variables must appear in the future data, considering that future inputs are independent from the past data. Thus, KPCA must be performed for $\mathbf{x}_k$ only, and another KPCA for $\mathbf{y}_k$ only.

Let $\mathbf{t}_k^{(1)} \in \mathbb{R}^{r_1}$ denote the PCs from $\mathbf{x}_k$, and $\mathbf{t}_k^{(2)} \in \mathbb{R}^{r_2}$ denote the PCs from $\mathbf{y}_k$. Although r_1 and r_2 are each chosen using the same cutoff criteria, they are not necessarily equal. Lagged variables are formed in Hankel matrices as in Eq. (21)-(22):

$$\mathbf{Y}_p = \begin{bmatrix} \mathbf{t}_p^{(1)} & \mathbf{t}_{p+1}^{(1)} & \mathbf{t}_{p+2}^{(1)} & \cdots & \mathbf{t}_{p+M-1}^{(1)} \\ \mathbf{t}_{p-1}^{(1)} & \mathbf{t}_p^{(1)} & \mathbf{t}_{p+1}^{(1)} & \cdots & \mathbf{t}_{p+M-2}^{(1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{t}_1^{(1)} & \mathbf{t}_2^{(1)} & \mathbf{t}_3^{(1)} & \cdots & \mathbf{t}_M^{(1)} \end{bmatrix}, \tag{21}$$

$$\mathbf{Y}_f = \begin{bmatrix} \mathbf{t}_{p+1}^{(2)} & \mathbf{t}_{p+2}^{(2)} & \mathbf{t}_{p+3}^{(2)} & \cdots & \mathbf{t}_{p+M}^{(2)} \\ \mathbf{t}_{p+2}^{(2)} & \mathbf{t}_{p+3}^{(2)} & \mathbf{t}_{p+4}^{(2)} & \cdots & \mathbf{t}_{p+M+1}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{t}_{p+f}^{(2)} & \mathbf{t}_{p+f+1}^{(2)} & \mathbf{t}_{p+f+2}^{(2)} & \cdots & \mathbf{t}_N^{(2)} \end{bmatrix}, \tag{22}$$

where $\mathbf{Y}_p \in \mathbb{R}^{r_1 p \times M}$ and $\mathbf{Y}_f \in \mathbb{R}^{r_2 f \times M}$ are respectively the past and future data matrices, p and f are respectively the number of lags in the past and future, and $M = N - p - f + 1$. Here, the amount of lags are chosen using autocorrelation analysis (Odiowei and Cao, 2010), but it must be ensured that $r_1 p < M$ and $r_2 f < M$ for results to make sense (Samuel and Cao, 2015). The Hankel matrices are normalized using the mean and standard deviation of each row, giving us $\widehat{\mathbf{Y}}_p$ and $\widehat{\mathbf{Y}}_f$. CVA proceeds by finding projections that maximize the correlations between $\widehat{\mathbf{Y}}_p$ and $\widehat{\mathbf{Y}}_f$. But since they may still be

rank-deficient after KPCA, they must first be factored by QR decomposition (Samuel, 2016):

$$\widehat{\mathbf{Y}}_p^T = \mathbf{Q}_p \mathbf{R}_p \mathbf{\Pi}_p^T, \tag{23}$$

$$\widehat{\mathbf{Y}}_f^T = \mathbf{Q}_f \mathbf{R}_f \mathbf{\Pi}_f^T, \tag{24}$$

where $\mathbf{Q}_p, \mathbf{Q}_f \in \mathbb{R}^{M \times M}$ are column orthogonal matrices, $\mathbf{R}_p \in \mathbb{R}^{M \times r_1 p}$ and $\mathbf{R}_f \in \mathbb{R}^{M \times r_2 f}$ are upper triangular matrices, and $\mathbf{\Pi}_p \in \mathbb{R}^{r_1 p \times r_1 p}$ and $\mathbf{\Pi}_f \in \mathbb{R}^{r_2 f \times r_2 f}$ are permutation matrices. The latter are used to permute the rows of $\mathbf{R}$ to have non-increasing absolute value of diagonal elements.

Let $\mathbf{Q}'_p$ and $\mathbf{Q}'_f$ denote the first ρ_1 columns of $\mathbf{Q}_p$ and the first ρ_2 columns of $\mathbf{Q}_f$, respectively, where $\rho_1 = \text{rank}(\widehat{\mathbf{Y}}_p)$ and $\rho_2 = \text{rank}(\widehat{\mathbf{Y}}_f)$. A numerically stable CVA involves the singular value decomposition (SVD) of the sample correlation matrix $\mathbf{H}$ as follows:

$$\mathbf{H} = (\mathbf{Q}'_f)^T \mathbf{Q}'_p = \boldsymbol{\mathcal{U}} \mathbf{\Sigma} \boldsymbol{\mathcal{V}}^T, \tag{25}$$

where $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{V}}$ consist of the left and right singular columns of $\mathbf{H}$, respectively, and $\mathbf{\Sigma}$ is a diagonal matrix of sorted singular values, $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_\rho$ with $\rho = \min(\rho_1, \rho_2)$. The singular vectors are rescaled and reordered by:

$$\mathbf{U} = \begin{bmatrix} (\mathbf{R}'_f)^{-1} \boldsymbol{\mathcal{U}} \sqrt{M-1} \\ \mathbf{0} \end{bmatrix} \mathbf{\Pi}_f^T \in \mathbb{R}^{r_2 f \times r_2 f}, \tag{26}$$

$$\mathbf{V} = \begin{bmatrix} (\mathbf{R}'_p)^{-1} \boldsymbol{\mathcal{V}} \sqrt{M-1} \\ \mathbf{0} \end{bmatrix} \mathbf{\Pi}_p^T \in \mathbb{R}^{r_1 p \times r_1 p}, \tag{27}$$

where $\mathbf{R}'_p$ and $\mathbf{R}'_f$ are the top-left $\rho_1 \times \rho_1$ submatrix of $\mathbf{R}_p$ and top-left $\rho_2 \times \rho_2$ submatrix of $\mathbf{R}_f$, respectively. The zero rows in (26)-(27) would appear only if dependent columns exist in $\widehat{\mathbf{Y}}_p$ or $\widehat{\mathbf{Y}}_f$.

Since only n (with $n < \rho$) dominant singular values explain the system dynamics (Odiowei and Cao, 2010; Pilario and Cao, 2018), only the first n columns of $\mathbf{U}$ and $\mathbf{V}$ are collected and denoted as $\mathbf{U}_n$ and $\mathbf{V}_n$, respectively. Projection matrices $\mathbf{J}$, $\mathbf{L}$ and $\mathbf{F}$ are formed as:

$$\mathbf{J} = \mathbf{V}_n^T \in \mathbb{R}^{n \times r_1 p}, \tag{28}$$

$$\mathbf{L} = \mathbf{U}_n^T \in \mathbb{R}^{n \times r_2 f}, \tag{29}$$

$$\mathbf{F} = \mathbf{I} - \mathbf{V}_n \mathbf{V}_n^T \in \mathbb{R}^{r_1 p \times r_1 p}, \tag{30}$$

which are used to reveal the state $\mathbf{Z}$ and residual $\mathbf{E}$ subspaces, as follows:

$$\mathbf{Z} \equiv [\mathbf{z}_k] = \mathbf{J}\widehat{\mathbf{Y}}_p \in \mathbb{R}^{n \times M}, \tag{31}$$

$$\mathbf{E} \equiv [\mathbf{e}_k] = \mathbf{F}\widehat{\mathbf{Y}}_p \in \mathbb{R}^{r_1 p \times M}, \tag{32}$$

where $\mathbf{z}_k$ are the state variables and $\mathbf{e}_k$ are the residual variables for $k = 1, \ldots, M$. Lastly, dissimilarity features $\mathbf{D}$ are computed as:

$$\mathbf{D} \equiv [\mathbf{d}_k] = \mathbf{L}\widehat{\mathbf{Y}}_f - \mathbf{\Sigma}_n \mathbf{J}\widehat{\mathbf{Y}}_p \in \mathbb{R}^{n \times M}, \tag{33}$$

where $\mathbf{\Sigma}_n = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ from the SVD in Eq. (25).

4.2. MK-CVDA based process monitoring

For process monitoring, the same statistical indices from the CVDA framework are adopted for MK-CVDA, defined as follows:

$$T_k^2 = \mathbf{z}_k^T \mathbf{z}_k, \tag{34}$$

$$Q_k = \mathbf{e}_k^T \mathbf{e}_k, \tag{35}$$

$$D_k = \mathbf{d}_k^T (\mathbf{I} - \mathbf{\Sigma}_n^2)^{-1} \mathbf{d}_k, \tag{36}$$

for the kth sampling instant.

Upper control limits (UCL), denoted by T^2_{UCL}, Q_{UCL}, and D_{UCL}, are computed using kernel density estimation (KDE) as explained in Odiowei and Cao (2010) and Pilario and Cao (2018). In KDE, the distributions of the indices are estimated, which may not necessarily be Gaussian. Given a significance level, α, the UCLs are solved such that $P(J < J_{\mathrm{UCL}}) = \alpha$ where $J \in \{T^2, Q, D\}$. The kth sample will be considered faulty if either of T_k^2, Q_k, or D_k exceeds T^2_{UCL}, Q_{UCL}, or D_{UCL}, respectively.

In summary, the overall algorithm of MK-CVDA is outlined in Fig. 2. As shown, KPCA is first used to project the original data onto a kernel feature space to handle nonlinearities, and further onto a kernel principal subspace to filter noise. In the CVDA step, past and future windows of kernel PCs are collected and projected onto the state and residual subspaces using a numerically stable CVA. The T^2, Q, and D statistical indices are finally used to detect and monitor faults in the features $\mathbf{z}$, $\mathbf{e}$, and $\mathbf{d}$, respectively.

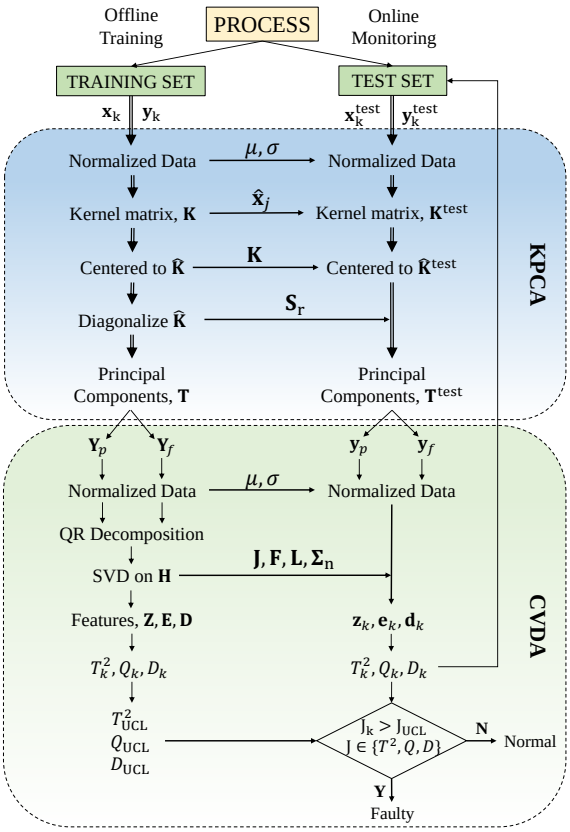

Figure 2: MK-CVDA algorithm for nonlinear process monitoring.

4.3. Parameter Selection

The parameters that must be set prior to the application of MK-CVDA include: the mixed kernel parameters c and ω, and the number of states (dominant singular values), n, in CVDA. Because these parameters are difficult to determine automatically, they must be subjected to an optimization procedure, where the objective might be to select values of $[c, \omega, n]$ that best distinguishes normal from faulty process conditions (as did Bernal-de Lázaro et al. (2016)). However, it is assumed that no prior fault information is available for checking this criteria. So to choose $[c, \omega, n]$, two different data sets are taken from the normal operation of the process: SET 1 (the training set) is used to train an MK-CVDA model and SET 2 (the validation set) is used to evaluate the model trained from SET 1. The optimal $[c, \omega, n]$ is defined as

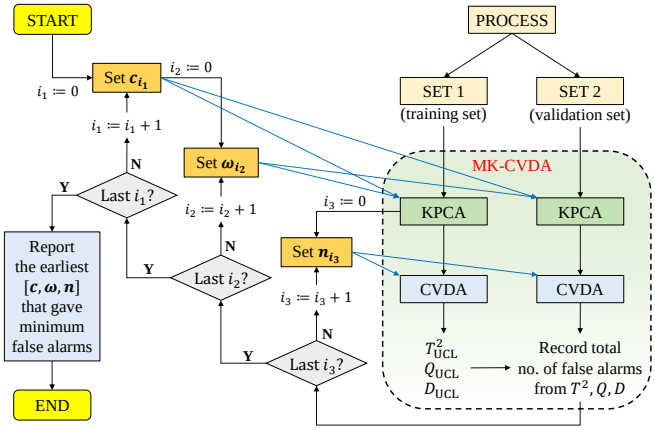

Figure 3: MK-CVDA offline training with grid search algorithm.

that which minimizes the combined false alarms incurred by the T^2, Q, and D indices in monitoring SET 2.

Several approaches exist for optimizing kernel parameters. Bernal-de Lázaro et al. (2016) used differential evolution (DE) and particle swarm optimization (PSO), and Jia et al. (2012) used genetic algorithms (GAs), to name a few. Although the use of these metaheuristics is attractive, the precision of results is not worth the computational effort. In reality, only a small range of $[c, \omega, n]$ needs to be explored, since under- or overitting may occur outside these ranges (Zhu et al., 2012). For example, too small c makes the RBF kernel sensitive to noise, while too large c creates a smooth mapping that may behave as linear (Bernal-de Lázaro et al., 2016). A similar case for choosing n is discussed by Ruiz-Cárcel et al. (2015). Hence, the grid search method is adopted as a practical way to find optimal parameters (see Zhu et al. (2012)). In grid search, combinations from only a finite set of values of $[c, \omega, n]$ are explored. The set of values are pre-defined manually depending on the problem.

In summary, the grid search method is used to decide kernel parameters c and ω, and the number of states, n, in MK-CVDA, by way of minimizing false alarms in a validation data set. After defining the sets of $[c, \omega, n]$ to explore, grid search is performed as represented in Fig. 3.

5. Case Studies

In this section, the benefits of using mixed kernels for KPCA monitoring are first illustrated in a numerical example. Next, the overall proposed MK-CVDA is evaluated using a closed-loop continuous stirred-tank reactor (CSTR) case study, described in Pilario and Cao (2018).

5.1. Numerical Example

A modified nonlinear example from Dong and McAvoy (1996) is considered as follows:

$$\begin{aligned} x_1 &= t + e_1, \\ x_2 &= t^2 - 3t + e_2, \\ x_3 &= -t^3 + 3t^2 + e_3, \\ y_1 &= -x_2(1 + x_1), \\ y_2 &= x_3 - \sin(1.5\pi x_2), \end{aligned} \tag{37}$$

where y_1 and y_2 are the only observable variables. A training data set was generated for time $t \in [0.01, 2]$ with $e_{1,2,3} \sim \mathcal{N}(0, 0.001)$. A test data set was also generated in the same period of time, where a slow linear drift fault occurred in y_2 starting at the 100th sample onwards. Each data set contains 300 samples. A plot of the training (blue) and test (red) data samples in the space of y_1-y_2 is shown in Fig. 4(a). The nonlinear behavior of the system manifests as a tortuous path taken by the output data from start to end of normal simulation. On the other hand, the incipient fault condition results in a gradual departure from the normal path. In this example, the goal is to distinguish the faulty from the normal data as early as possible. All UCLs in this example are calculated using kernel density estimation with 99% significance level.

PCA is a standard first-choice technique in process monitoring which finds only linear projections of the original data at directions of maximum variance. Using PCA, the widely used Hotelling's T^2 statistical index for every point in the y_1-y_2 data space is calculated and shown as a contour map in Fig. 4(b). The upper control limit, T^2_{UCL}, is depicted as an elliptical envelope around the training data whose axes lie at the principal component directions. Any point inside the envelope is deemed as under normal condition, while outside the envelope is under faulty condition. As shown, the path of the test data

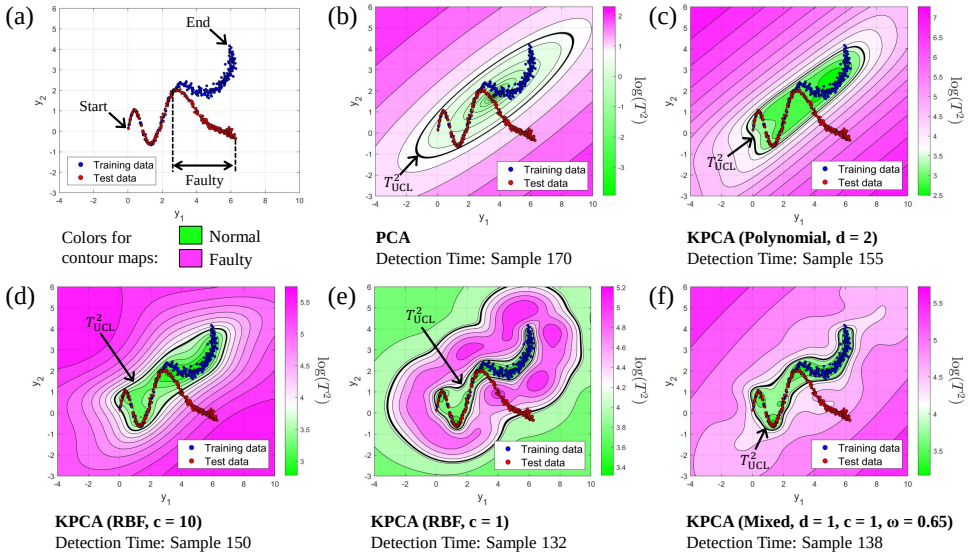

Figure 4: (a) Plot of training (blue) and test (red) data from the numerical example; contour maps of the T^2 statistical index (log-scale) from: (b) PCA; (c) KPCA with polynomial kernel, Eq. (15); (d,e) KPCA with RBF kernel, Eq. (14); (f) KPCA with mixed kernel, Eq. (16). The 99% significance T^2_{UCL} envelope is shown as a thick contour line.

leaves the normal region only at the 170th sample, which gives a detection delay of 70 samples for PCA-based monitoring.

By finding nonlinear projections of the original data, KPCA can be used to improve the detection performance. KPCA leads to tighter UCL bounds around the data at normal conditions because the nonlinear behavior of the process is captured more accurately (Schölkopf et al., 1998). The results in Fig. 4(c) to 4(f) indeed reflect this improvement. For a fair comparison, the same cutoff criteria was applied in choosing the number of top principal components in KPCA for all generated T^2 contour maps (see Section 2).

In Fig. 4(c), a quadratic kernel is used, i.e. Eq. (15) with $d = 2$. Although the detection time was improved by 15 samples against PCA, the underlying nonlinear behavior of the process is still not captured by the kernel. The T^2_{UCL} envelope in Fig. 4(c) is tighter than that in Fig. 4(b), but it still does not follow the path of training data closely. This limitation is due to the poor interpolation ability of the quadratic kernel. Nonetheless, good extrapolation behavior of the quadratic kernel is exhibited by the increasing T^2 contours radially outward from the normal region, which reflects an increasing fault

magnitude as the test samples move farther from the normal region.

In Fig. 4(d), an RBF kernel with $c = 10$ (see Eq. (14)) is used. An improvement in detection time by 20 samples against PCA is realized due to an even tighter T^2_{UCL} envelope than that in Fig. 4(c). Lowering the value of c increases the fit to the training data, which gives even tighter envelopes. For $c = 1$ in Fig. 4(e), the KPCA detection performance is seen to have achieved a large improvement in detection time, which is 38 samples earlier than that in PCA. This behavior exemplifies the good interpolation ability of the RBF kernel. However, as mentioned in Section 3.1, the extrapolation ability of the RBF kernel is compromised when low values of c are chosen. This local kernel behavior is confirmed in Fig. 4(e), where the latest faulty test samples are undesirably perceived to be normal. Using the analysis in Theorem 1 for the scenario in Fig. 4(e), the final T^2 value as the fault effect tends to infinity is computed to be $T^2 = 8.77 \times 10^3$. Since this value is below the computed $T^2_{\text{UCL}} = 1.68 \times 10^4$, then $c = 1$ is considered to be a poor choice of kernel width. With a small c, the problem of rising and then falling T^2 has already been observed by Choi et al. (2005), and this is seen as a drawback of local kernels. In any case, it is difficult to tune the kernel width alone to strive for earlier detection while ensuring that T^2 is always increasing outward beyond the T^2_{UCL} envelope.

The T^2 contour map for KPCA with a mixed kernel is shown in Fig. 4(f). In the mixed kernel, the low RBF kernel width of $c = 1$ provided a more accurate capture of nonlinear behavior than that in Fig. 4(d), while simultaneously ensuring that the T^2 is increasing beyond the T^2_{UCL} envelope due to the contribution of the polynomial kernel. Hence, the use of mixed kernels improves incipient fault monitoring performance by having both good interpolation and extrapolation abilities. At the chosen parameters, detection is achieved 38 samples after the introduction of the fault in the numerical example.

Even though the mixed kernel has been shown to have greater flexibility in modelling the process, kernel parameters must still be chosen carefully in order to maximize the capabilities of KPCA. Hence, this paper suggests grid search for parameter selection (see Section 4.3). After demonstrating the importance of mixed kernels, the performance of the overall MK-CVDA is evaluated in a nonlinear dynamic case study.

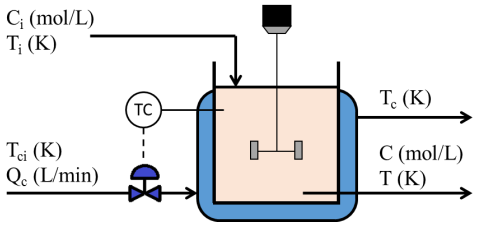

Figure 5: Schematic of the CSTR case study.

5.2. CSTR Case Study

A schematic of the CSTR is given in Fig. 5. The data from this process are simulated by the following nonlinear state-space model:

$$\frac{dC}{dt} = \frac{Q}{V}(C_i - C) - akC + \nu_1, \tag{38}$$

$$\frac{dT}{dt} = \frac{Q}{V}(T_i - T) - a\frac{(\Delta H_r)kC}{\rho C_p} - b\frac{UA}{\rho C_p V}(T - T_c) + \nu_2, \tag{39}$$

$$\frac{dT_c}{dt} = \frac{Q_c}{V_c}(T_{ci} - T_c) + b\frac{UA}{\rho_c C_{pc} V_c}(T - T_c) + \nu_3, \tag{40}$$

where $\mathbf{u} = \begin{bmatrix} C_i & T_i & T_{ci} \end{bmatrix}^T$ and $\mathbf{y} = \begin{bmatrix} C & T & T_c & Q_c \end{bmatrix}^T$ are the respective inputs and outputs, and $k = k_0 \exp\left(\frac{-E}{RT}\right)$. Here, the same controller settings and parameter values in Eq. (38)-(40) were used as those in Pilario and Cao (2018). Simulations of normal and faulty data were carried out under varying operating conditions every 60 min. The incipient faults listed in Table 1 are investigated. Fault 1 is a drift in the readings of reactor temperature, which is the controlled variable. This fault produces oscillations to the coolant flow rate, as the controller becomes saturated. Fault 2 is a slow decay in catalyst activity, introduced by decreasing the value of a in Eq. (38) and Eq. (39) to zero. Lastly, Fault 3 is a fouling fault in the cooling jacket, introduced by decreasing the value of b in Eq. (39) and Eq. (40) to zero. In all fault scenarios, the incipient fault is introduced only after 200 min of normal operation.

In the following, results for MK-CVDA offline training and online monitoring are presented.

5.2.1. MK-CVDA offline training

To proceed with offline training, a training set and a validation set is generated from the CSTR, each consisting of 1200 samples of the 7 variables.

Table 1: Incipient fault scenarios in the CSTR

Fault	Name	Description†	δ	Nominal Values
1	Sensor Drift	$T = T_0 + \delta t$	0.005	$T_0 = 430\,\mathrm{K}$
2	Catalyst Decay	$a = a_0 \exp(-\delta t)$	0.0005	$a_0 = 1$
3	Fouling	$b = b_0 \exp(-\delta t)$	0.001	$b_0 = 1$

† All t in minutes.

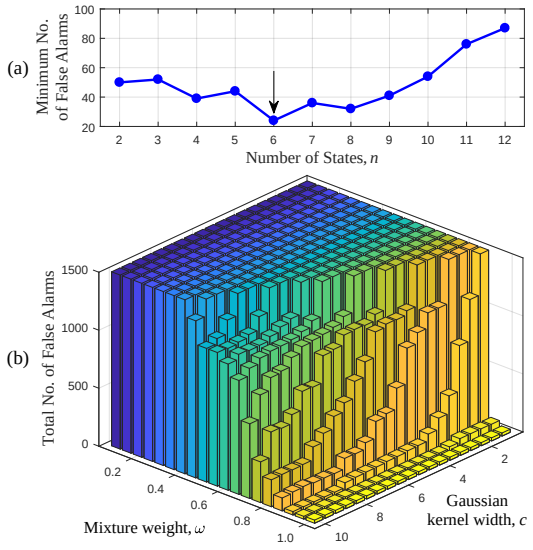

Figure 6: Grid search results for the CSTR: (a) no. of false alarms against n; (b) no. of false alarms against $[c, \omega]$ at $n = 6$. Note: The upper portion of the bar graph is tapered at 1500 in the z-axis.

The sampling interval is 1 min. Random seeds for input disturbances and noise are different between the training and validation sets. Grid search (Fig. 3) was then used to find MK-CVDA parameters, $[c, \omega, n]$, such that false alarms in the validation set are minimized. For this case, the following sets were considered: $c \in \{1, 0.5, 2, \ldots, 10\}$, $\omega \in \{0, 0.05, 0.1, \ldots, 1\}$, and $n \in \{2, 3, \ldots, 12\}$. Besides, if smaller increments are used in the c and ω sets, the number of false alarms may be indistinct between adjacent choices.

In Fig. 6(a), the minimum number of false alarms ever recorded for each n (at any $[c, \omega]$) are plotted, indicating that the CSTR must have $n = 6$ states. Further in Fig. 6(b), a bar graph of the number of false alarms against choices of $[c, \omega]$ for $n = 6$ is shown. These results agree with those from Jordaan (2002) and Zhu et al. (2012) in that the choice of mixture weight is desirable

at high values of ω, i.e. only a "pinch" of the RBF kernel needs to be added to improve the interpolation ability of a low-order polynomial kernel. In other words, the influence of each kernel type in the mixture may not necessarily be balanced, i.e. at $\omega = 0.5$, for optimal performance. At low values of both c and ω, Fig. 6(b) shows that the resulting MK-CVDA models are not suitable since the statistical indices are found above the detection limits most of the time. In the end, the grid search found $[c, \omega, n] = [4.5, 0.95, 6]$ as the optimal parameters for MK-CVDA monitoring of the CSTR.

5.2.2. MK-CVDA online monitoring

In this subsection, the proposed MK-CVDA method is compared with linear CVDA and KCVDA (which uses RBF kernels only). For both KCVDA and linear CVDA, the same number of states as MK-CVDA was adopted, i.e. $n = 6$. For KCVDA, the same kernel width of $c = 4.5$ as with MK-CVDA was tried. However, this setting has produced the same scenario as in Fig. 4(e), leading to poor KCVDA monitoring results. Instead, $c = 3300$ is adopted for KCVDA, which is the setting among $\{100, 200, \ldots, 5000\}$ that produced the minimum false alarms when the validation set was monitored.

The performance of any process monitoring method can be evaluated using detection delays (DD), false alarm rates (FAR), and missed detection rates (MDR). In this case study, detection time is defined as the first time when 10 consecutive alarms are raised from the start of operation. Hence, DD is the period between the start of fault and the detection time. Also, standard FAR and MDR definitions are given for statistical index J as:

$$\text{FAR} = \frac{\text{no. of samples } (J > J_{\text{UCL}} | \text{fault-free})}{\text{total samples (fault-free)}} \times 100\%, \tag{41}$$

$$\text{MDR} = \frac{\text{no. of samples } (J < J_{\text{UCL}} | \text{fault})}{\text{total samples (fault)}} \times 100\%. \tag{42}$$

For a robust comparison, 15 test data sets were generated for each fault scenario in Table 1, while varying the random seeds for disturbances and noise. In each test data set, false alarms were recorded whenever $J > J_{\text{UCL}}$ in the first 200 min of normal operation, while missed detections were recorded whenever $J < J_{\text{UCL}}$ in the faulty operation from 201-1200 min, where $J \in \{T^2, Q, D\}$. After monitoring all test data, performance results are summarized in Fig. 7 as box plots. Each row of box plots correspond to a fault scenario, while each column of box plots correspond to DD, FAR, and MDR

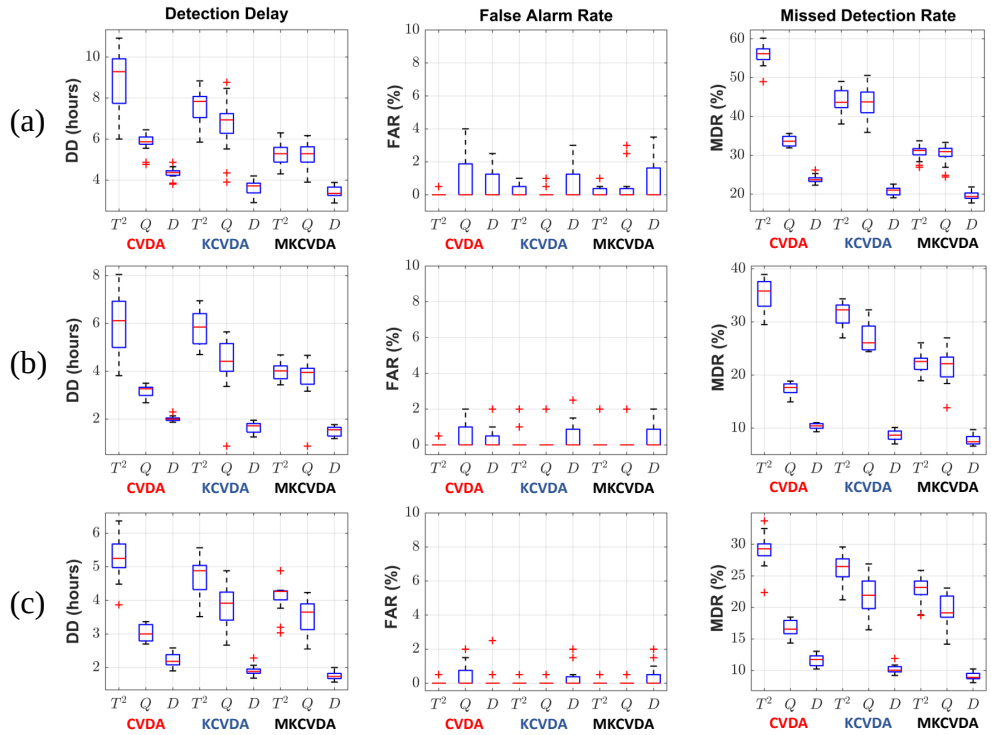

Figure 7: Monitoring performance for fault conditions: (a) Fault 1; (b) Fault 2; and, (c) Fault 3, by comparing CVDA (no kernel), KCVDA (RBF kernel, $c = 3300$), and MK-CVDA (mixed kernel, $d = 1, c = 4.5, \omega = 0.95$) using DD (in hours), FAR (in %), and MDR (in %). The boxplots summarize the outcome of 15 trial data sets. All UCLs are computed at 99.9% significance level.

results. Within a box plot, the statistical indices from CVDA, KCVDA, and MK-CVDA are compared with each other. For ease of comparison, the medians of all boxplots are tabulated in Table 2. In general, a good detection index must have low DD, FAR, and MDR, and must also depict the severity of the fault properly above the detection limit. Using these criteria, the statistical indices were evaluated as follows.

Due to having a more accurate nonlinear model at optimal mixed kernel parameters, the T^2_{MKCVDA} and D_{MKCVDA} indices obtained earlier detection times and less MDR against their counterparts in CVDA, for all fault scenarios. The margin of improvement in detection time for T^2_{MKCVDA} even reached as much as 4 hours against T^2_{CVDA}, also with less variability. Both

Table 2: Boxplot medians† from the results in Fig. 7

Fault	CVDA			KCVDA			MK-CVDA		
	T^2	Q	D	T^2	Q	D	T^2	Q	D
Detection Delay (DD, hours)									
1	9.28	5.87	4.37	7.83	6.93	3.72	5.28	5.28	**3.35**
2	6.12	3.27	2.00	5.85	4.42	1.72	4.02	3.95	**1.55**
3	5.25	3.00	2.18	4.88	3.92	1.88	4.27	3.65	**1.73**
False Alarm Rate (FAR, %)									
	All medians are zero.								
Missed Detection Rate (MDR, %)									
1	56.17	33.60	23.67	43.64	43.74	20.99	31.23	30.92	**19.34**
2	35.81	17.65	10.43	32.26	26.06	8.69	22.54	22.13	**7.45**
3	29.29	16.55	11.74	26.47	21.92	10.03	23.16	19.13	**8.89**

† The best values in each row are boldfaced.

T^2 and D are statistical indices for the state subspace, while the Q index is that for the residual subspace. Since CVDA extracted only a linear model for the nonlinear process, the Q_{CVDA} index can readily detect departures from this linear model. For this reason, the Q_{CVDA} index detected parametric Faults 2 and 3 earlier than Q_{MKCVDA}. Nonetheless, Q_{MKCVDA} still performed better than the Q_{CVDA} for sensor Fault 1. These improvements in performance are attributed to the good interpolation ability of the kernel in MK-CVDA, which leads to an accurate capture of nonlinear process behavior. Also, majority of the test data sets incurred no false alarms during normal operation. Hence, the median of FAR on all the indices is reported as zero (see Table 2).

As a nonlinear process monitoring method, KCVDA is also expected to extract a more accurate process model than linear CVDA. However, without using mixed kernels, the results are not as improved as that of MK-CVDA in terms of DD and MDR (see Table 2). In what follows, the effect of kernel width c to KCVDA performance is further investigated.

Similar to what was done in the previous case study, i.e. going from Fig. 4(d) to Fig. 4(e), lowering the value of c in the RBF kernel can create tighter bounds around the normal data and improve the detection time of KCVDA. To proceed with this experiment, a new test data set under Fault 2 conditions (catalyst decay) was generated and monitored. Sample profiles

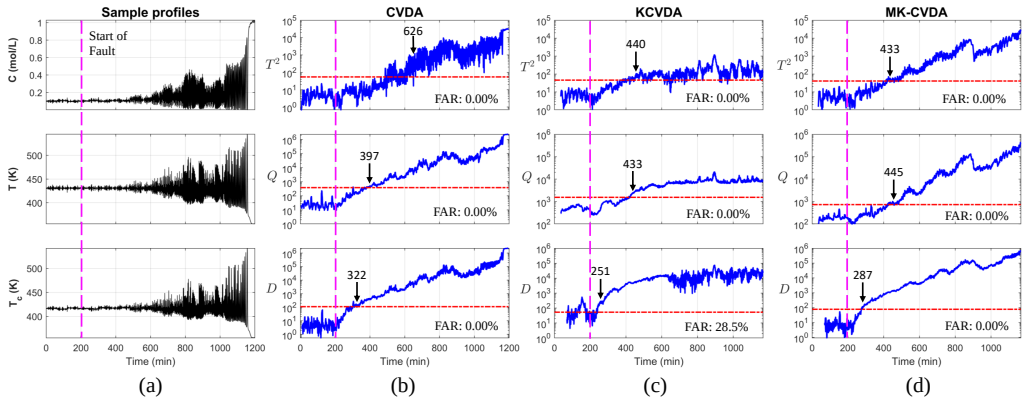

Figure 8: (a) Plots of sample C, T, T_c data set at Fault 2 conditions; monitoring charts (with detection times) using: (b) CVDA; (c) KCVDA (RBF kernel, $c = 100$); and, (d) MK-CVDA (mixed kernel, $d = 1, c = 4.5, \omega = 0.95$) on the data set in (a). Legend: Dashdot - UCL; Solid - statistical index; Dash - start of fault.

from this data set are shown in Fig. 8(a), where the incipient fault is seen to remain elusive in the first few hours after its introduction. Yet, the same fault leads to a process failure within 20 hours. Monitoring charts for CVDA, KCVDA and MK-CVDA are shown in Fig. 8(b)-(d), respectively, where all settings are retained except that $c = 100$ in the RBF kernel in KCVDA.

Indeed, at $c = 100$, KCVDA has improved by detecting the catalyst decay fault earlier than CVDA and MK-CVDA via the D_{KCVDA} index. However, the large increase in FAR for D_{KCVDA} is a sign of overfitting to the training data for the chosen value of c. Prevalent false alarms is undesirable as this makes the detection method unreliable in practice. Moreover, the fault severity is not properly reflected in any KCVDA index above their respective UCLs. Note that the incipient fault continues to degrade the process (as seen in Fig. 8(a)), especially at 600-1200 min of operation. Yet, the KCVDA indices remain levelled during these times (cf. Theorem 1). This demonstrates the limitation of KCVDA which uses the RBF kernel alone, i.e. at low values of kernel width c, interpolation ability improves but at the expense of extrapolation ability. On the other hand, in all the MK-CVDA indices, fault severity is reflected properly and this demonstrates good extrapolation ability of the kernel. In addition, T^2_{MKCVDA} and D_{MKCVDA} detection times and MDRs have improved significantly against T^2_{CVDA} and D_{CVDA}, owing to the good interpolation ability of the kernel. Hence, MK-CVDA is a better

route for *kernelizing* CVDA rather than KCVDA in incipient fault monitoring. Due to these results, it is recommended to consider mixed kernels in any kernel MSPM method for better performance in monitoring incipient faults in nonlinear dynamic processes.

In monitoring incipient faults, any improvement in the detection time means additional lead time before failure eventually occurs in the process. Within this lead time, activities such as fault prognosis and condition-based maintenance can take effect in light of the incipient fault condition. Hence, major plant failures can be avoided with better process monitoring methods.

In summary, the benefits of MK-CVDA are stated as follows: (i) enhanced sensitivity to incipient faults (low DD) due to the dissimilarity index and the use of mixed kernels; (ii) increased detection reliability (less MDR and FAR) due to handling of nonlinearities, dynamics, and non-Gaussianity by kernels, a numerically stable CVA, and KDE based threshold setting, respectively; and, (iii) better depiction of incipient fault growth in the monitoring charts due to mixed kernels with both interpolation and extrapolation abilities. While grid search has also been shown to be applicable for parameter selection in MK-CVDA, more efficient techniques on this aspect needs further research.

6. Conclusion and Future Perspectives

In this paper, the drawbacks of using a single RBF kernel or polynomial kernel for KPCA-based incipient fault monitoring were first examined empirically and theoretically. To address these drawbacks, mixed kernels were adopted to combine the benefits from single kernels. The CVDA method for incipient fault detection was then extended by preceding it with KPCA and mixed kernels. Hence, a new Mixed Kernel Canonical Variate Dissimilarity Analysis (MK-CVDA) method was proposed for incipient fault monitoring.

To decide parameters for MK-CVDA, cross-validation by grid search is suggested. Two case studies were used to demonstrate the improved performance of using mixed kernels and also MK-CVDA over linear CVDA in terms of detection delay, false alarm rates, and missed detection rates. More importantly, predictive monitoring performance is improved because the growth of a fault across time is better depicted by the MK-CVDA indices beyond their detection limits.

The MK-CVDA method in this study can be extended in the future for fault diagnosis and prognosis. Also, the effect of other kernel choices in

KPCA must be investigated, similar to that in Section 5.1, as well as the development of more efficient methods for parameter selection in these kernels. Fundamentally, the idea of combining kernels for enhancing generalization capabilities offers new directions for nonlinear process monitoring research, not only in KPCA, but also for other kernel methods such as Gaussian process models (Duvenaud, 2014; Ge, 2018) and extreme learning machines (Chen et al., 2018; Wang and Han, 2014; Lian et al., 2013).

MK-CVDA can also participate in a distributed framework to cope with the size of larger process industries. For example, Bayesian networks can be used to organize the plant into blocks that each cite an instance of MK-CVDA, similar to the work of Zhu et al. (2018). For the challenging task of analyzing big data, i.e. millions of samples, some approaches such as those by Yao and Ge (2018) and Zhu et al. (2017) can also be utilized, where MK-CVDA takes the place of PCA.

Acknowledgement

The authors gratefully acknowledge the support from the DOST-ERDT Faculty Development Fund of the Republic of the Philippines.

References

Botre, C., Mansouri, M., Nounou, M., Nounou, H., Karim, M.N., 2016. Kernel PLS-based GLRT method for fault detection of chemical processes. J. Loss Prev. Process Ind. 43, 212–224. doi:10.1016/j.jlp.2016.05.023.

Cai, L., Tian, X., Chen, S., 2017. Monitoring nonlinear and non-Gaussian processes using Gaussian mixture model-based weighted kernel independent component analysis. IEEE Trans. Neural Networks Learn. Syst. 28, 122–135. doi:10.1109/TNNLS.2015.2505086.

Chen, Y., Kloft, M., Yang, Y., Li, C., Li, L., 2018. Mixed kernel based extreme learning machine for electric load forecasting. Neurocomputing 312, 90–106. doi:10.1016/j.neucom.2018.05.068.

Cheng, C.Y., Hsu, C.C., Chen, M.C., 2010. Adaptive kernel principal component analysis (KPCA) for monitoring small disturbances of nonlinear processes. Ind. Eng. Chem. Res. 49, 2254–2262. doi:10.1021/ie900521b.

Cheng, K., Lu, Z., Wei, Y., Shi, Y., Zhou, Y., 2017. Mixed kernel function support vector regression for global sensitivity analysis. Mech. Syst. Signal Process. 96, 201–214. doi:10.1016/j.ymssp.2017.04.014.

Chiang, L.H., Russell, E.L., Braatz, R.D., 2005. Fault Detection and Diagnosis in Industrial Systems. Springer-Verlag, London.

Choi, S.W., Lee, C., Lee, J.M., Park, J.H., Lee, I.B., 2005. Fault detection and identification of nonlinear processes based on kernel PCA. Chemom. Intell. Lab. Syst. 75, 55–67. doi:10.1016/j.chemolab.2004.05.001.

Cristianini, N., Shawe-Taylor, J., 2014. Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press.

Dong, D., McAvoy, T., 1996. Nonlinear principal component analysis based on principal curves and neural networks. Comput. Chem. Eng. 20, 65–78. doi:10.1016/0098-1354(95)00003-K.

Duvenaud, D.K., 2014. Automatic Model Construction with Gaussian Processes. Ph.D. thesis. Cambridge University.

Fan, J., Wang, Y., 2014. Fault detection and diagnosis of non-linear non-Gaussian dynamic processes using kernel dynamic independent component analysis. Inf. Sci. (Ny). 259, 369–379. doi:10.1016/j.ins.2013.06.021.

Fezai, R., Mansouri, M., Taouali, O., Harkat, M.F., Bouguila, N., 2018. Online reduced kernel principal component analysis for process monitoring. J. Process Control 61, 1–11. doi:10.1016/j.jprocont.2017.10.010.

Ge, Z., 2017. Review on data-driven modeling and monitoring for plant-wide industrial processes. Chemom. Intell. Lab. Syst. 171, 16–25. doi:10.1016/j.chemolab.2017.09.021.

Ge, Z., 2018. Distributed predictive modeling framework for prediction and diagnosis of key performance index in plant-wide processes. J. Process Control 65, 107–117. doi:10.1016/j.jprocont.2017.08.010.

Ge, Z., Chen, J., 2016. Plant-wide industrial process monitoring: a distributed modeling framework. IEEE Trans. Ind. Informatics 12, 310–321. doi:10.1109/TII.2015.2509247.

Ge, Z., Song, Z., Gao, F., 2013. Review of recent research on data-based process monitoring. Ind. Eng. Chem. Res. 52, 3543–3562.

Isermann, R., 2005. Model-based fault-detection and diagnosis - status and applications. Annu. Rev. Control 29, 71–85. doi:10.1016/j.arcontrol.2004.12.002.

Jaffel, I., Taouali, O., Harkat, M.F., Messaoud, H., 2016. Moving window KPCA with reduced complexity for nonlinear dynamic process monitoring. ISA Trans. 64, 184–192. doi:10.1016/j.isatra.2016.06.002.

Ji, H., He, X., Shang, J., Zhou, D., 2018. Exponential smoothing reconstruction approach for incipient fault isolation. Ind. Eng. Chem. Res. 57, 6353–6363. doi:10.1021/acs.iecr.8b00478.

Jia, M., Xu, H., Liu, X., Wang, N., 2012. The optimization of the kind and parameters of kernel function in KPCA for process monitoring. Comput. Chem. Eng. 46, 94–104. doi:10.1016/j.compchemeng.2012.06.023.

Jordaan, E.M., 2002. Development of Robust Inferential Sensors: Industrial Applications of Support Vector Machines for Regression. Ph.D. thesis. Technische Universiteit Eindhoven. doi:10.6100/IR561175.

Bernal-de Lázaro, J.M., Llanes-Santiago, O., Prieto-Moreno, A., Knupp, D.C., Silva-Neto, A.J., 2016. Enhanced dynamic approach to improve the detection of small-magnitude faults. Chem. Eng. Sci. 146, 166–179. doi:10.1016/j.ces.2016.02.038.

Lee, J.M., Yoo, C.K., Lee, I.B., 2004. Fault detection of batch processes using multiway kernel principal component analysis. Comput. Chem. Eng. 28, 1837–1847. doi:10.1016/j.compchemeng.2004.02.036.

Lian, C., Zeng, Z., Yao, W., Tang, H., 2013. Displacement prediction of landslide based on PSOGSA-ELM with mixed kernel, in: 2013 Sixth Int. Conf. Adv. Comput. Intell., IEEE. pp. 52–57. doi:10.1109/ICACI.2013.6748473.

Liu, Q., Zhu, Q., Qin, S.J., Chai, T., 2018a. Dynamic concurrent kernel CCA for strip-thickness relevant fault diagnosis of continuous annealing processes. J. Process Control 67, 12–22. doi:10.1016/j.jprocont.2016.11.009.

Liu, Y., Liu, B., Zhao, X., Xie, M., 2018b. A mixture of variational canonical correlation analysis for nonlinear and quality-relevant process monitoring. IEEE Trans. Ind. Electron. 65, 6478–6486. doi:10.1109/TIE.2017.2786253.

Lu, Q., Jiang, B., Gopaluni, R.B., Loewen, P.D., Braatz, R.D., 2018. Locality preserving discriminative canonical variate analysis for fault diagnosis. Comput. Chem. Eng. 117, 309–319. doi:10.1016/j.compchemeng.2018.06.017.

Nguyen, V.H., Golinval, J.C., 2010. Fault detection based on kernel principal component analysis. Eng. Struct. 32, 3683–3691. doi:10.1016/j.engstruct.2010.08.012.

Odiowei, P.E., Cao, Y., 2010. Nonlinear dynamic process monitoring using canonical variate analysis and kernel density estimations. IEEE Trans. Ind. Informatics 6, 36–45. doi:10.1109/TII.2009.2032654.

Pilario, K.E., Cao, Y., 2017. Process incipient fault detection using canonical variate analysis, in: 2017 23rd Int. Conf. Autom. Comput., IEEE. pp. 1–6. doi:10.23919/IConAC.2017.8082031.

Pilario, K.E.S., Cao, Y., 2018. Canonical variate dissimilarity analysis for process incipient fault detection. IEEE Trans. Ind. Informatics doi:10.1109/TII.2018.2810822.

Rato, T.J., Reis, M.S., 2014. Sensitivity enhancing transformations for monitoring the process correlation structure. J. Process Control 24, 905–915. doi:10.1016/j.jprocont.2014.04.006.

Ruiz-Cárcel, C., Cao, Y., Mba, D., Lao, L., Samuel, R.T., 2015. Statistical process monitoring of a multiphase flow facility. Control Eng. Pract. 42, 74–88. doi:10.1016/j.conengprac.2015.04.012.

Samuel, R.T., 2016. Nonlinear Dynamic Process Monitoring Using Kernel Methods. Ph.D. thesis. Cranfield University.

Samuel, R.T., Cao, Y., 2015. Kernel canonical variate analysis for nonlinear dynamic process monitoring. IFAC-PapersOnLine 28, 605–610. doi:10.1016/j.ifacol.2015.09.034.

Schölkopf, B., Smola, A., Müller, K.R., 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput. 10, 1299–1319. doi:10.1162/089976698300017467.

Shafiee, M., 2016. Modelling and analysis of availability for critical interdependent infrastructures. Int. J. Risk Assess. Manag. 19, 299–314. doi:10.1504/IJRAM.2016.079608.

Shang, J., Chen, M., Zhang, H., 2018. Fault detection based on augmented kernel Mahalanobis distance for nonlinear dynamic processes. Comput. Chem. Eng. 109, 311–321. doi:10.1016/j.compchemeng.2017.11.010.

Shao, J.D., Rong, G., Lee, J.M., 2009. Learning a data-dependent kernel function for KPCA-based nonlinear process monitoring. Chem. Eng. Res. Des. 87, 1471–1480. doi:10.1016/j.cherd.2009.04.011.

Smola, A.J., Ovari, Z.L., Williamson, R.C., 2000. Regularization with dot-product kernels, in: Proc. Neural Inf. Process. Syst., MIT Press. pp. 308–314.

Vachtsevanos, G., Lewis, F.L., Roemer, M., Hess, A., Wu, B., 2006. Intelligent Fault Diagnosis and Prognosis for Engineering Systems. John Wiley & Sons, Inc.

Wang, X.Y., Han, M., 2014. Multivariate time series prediction based on multiple kernel extreme learning machine. 2014 Int. Jt. Conf. Neural Networks 61, 198–201. doi:10.1109/IJCNN.2014.6889479.

Xu, L., Niu, X., Xie, J., Abel, A., Luo, B., 2015. A local-global mixed kernel with reproducing property. Neurocomputing 168, 190–199. doi:10.1016/j.neucom.2015.05.107.

Yang, X., Peng, H., Shi, M., 2013. SVM with multiple kernels based on manifold learning for breast cancer diagnosis, in: 2013 IEEE Int. Conf. Inf. Autom., IEEE. pp. 396–399. doi:10.1109/ICInfA.2013.6720330.

Yao, L., Ge, Z., 2018. Big data quality prediction in the process industry: a distributed parallel modeling framework. J. Process Control 68, 1–13. doi:10.1016/j.jprocont.2018.04.004.

Yin, S., Li, X., Gao, H., Kaynak, O., 2015. Data-based techniques focused on modern industry: an overview. IEEE Trans. Ind. Electron. 62, 657–667. doi:10.1109/TIE.2014.2308133.

Yin, Z., Hou, J., 2016. Recent advances on SVM based fault diagnosis and process monitoring in complicated industrial processes. Neurocomputing 174, 643–650. doi:10.1016/j.neucom.2015.09.081.

Zhang, X., Polycarpou, M.M., Parisini, T., 2002. A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems. IEEE Trans. Automat. Contr. 47, 576–593. doi:10.1109/9.995036.

Zhang, Y., 2009. Enhanced statistical analysis of nonlinear processes using KPCA, KICA and SVM. Chem. Eng. Sci. 64, 801–811. doi:10.1016/j.ces.2008.10.012.

Zhang, Y., Zhang, Y., 2010. Process monitoring, fault diagnosis and quality prediction methods based on the multivariate statistical techniques. IETE Tech. Rev. 27, 406. doi:10.4103/0256-4602.62226.

Zhong, Z., Carr, T.R., 2016. Application of mixed kernels function (MKF) based support vector regression model (SVR) for CO2-reservoir oil minimum miscibility pressure prediction. Fuel 184, 590–603. doi:10.1016/j.fuel.2016.07.030.

Zhong, Z., Liu, S., Kazemi, M., Carr, T.R., 2018. Dew point pressure prediction based on mixed-kernels-function support vector machine in gas-condensate reservoir. Fuel 232, 600–609. doi:10.1016/j.fuel.2018.05.168.

Zhu, J., Ge, Z., Song, Z., 2017. Distributed parallel PCA for modeling and monitoring of large-scale plant-wide processes with big data. IEEE Trans. Ind. Informatics 13, 1877–1885. doi:10.1109/TII.2017.2658732.

Zhu, J., Ge, Z., Song, Z., Zhou, L., Chen, G., 2018. Large-scale plant-wide process modeling and hierarchical monitoring: A distributed Bayesian network approach. J. Process Control 65, 91–106. doi:10.1016/j.jprocont.2017.08.011.

Zhu, X., Huang, Z., Tao Shen, H., Cheng, J., Xu, C., 2012. Dimensionality reduction by mixed kernel canonical correlation analysis. Pattern Recognit. 45, 3003–3016. doi:10.1016/j.patcog.2012.02.007.

https://dspace.org/

Mixed kernel canonical variate dissimilarity analysis for incipient fault monitoring in nonlinear dynamic processes

Pilario, Karl Ezra; Cao, Yi; Shafiee, Mahmood

2018-12-25
Elsevier
https://doi.org/10.1016/j.compchemeng.2018.12.027
Pilario KE, Cao Y, Shafiee M. Mixed kernel canonical variate dissimilarity analysis for incipient fault monitoring in nonlinear dynamic processes. Computers and Chemical Engineering, Volume 123, April 2019, pp. 143-154

Downloaded from a DSpace repository. https://dspace.org/