An efficient multimodal attentional principal component analysis for continual learning-based dynamic process monitoring

Article

Accepted Version

An efficient multimodal attentional principal component analysis for continual learning-based dynamic process monitoring

Jingxin Zhang[a,*], Haikun Wei[a], Kanjian Zhang[a], James Xiao[b] and Xia Hong[c]

[a]School of Automation, Southeast University, Nanjing 210096, China
[b]Independent researcher, Vancouver, BC, V6K 2R1, Canada.
[c]Department of Computer Science, School of Mathematical, Physical and Computational Sciences, University of Reading, RG6 6AY, U.K.

ARTICLE INFO

Keywords:
Multimode dynamic process monitoring
multimodal attentional principal component analysis
replay continual learning
attention mechanism

ABSTRACT

Traditional multimode process monitoring methods extract features from time series data. Due to the catastrophic forgetting effect, data-driven multimode dynamic process monitoring is challenging based on a single monitoring model paradigm, i.e. the learned knowledge from previous modes may diminish as operating conditions undergo changes between modes, yet it is impractical to access all past data to retrain the model. In this work, a novel efficient method of multimodal attentional principal component analysis (M-APCA) with continual learning ability is introduced. Under the assumption that data from successive modes are received sequentially, dynamic process data are modeled using an attention mechanism to capture the relationship between data and the latent space, whereby meaningful information is concentrated as dynamic features which are extracted via a vector autoregressive model. In order to overcome the catastrophic forgetting problem, the idea of replay continual learning is employed. Specifically, past modes' data which are significant to reflect the operating conditions, are selected and stored. These are repeatedly used in tandem with sequential data as replay data. Two types of attention mechanisms are considered and analyzed, each of which is specifically designed to learn from data in an unsupervised manner, so the overall algorithm is efficient both in time and storage costs. The proposed attentional principal component analysis and M-APCA are analyzed against several state-of-the-art methods to highlight the virtues of the proposed method. Compared with multimode monitoring methods, the effectiveness is demonstrated through case studies of: a continuous stirred tank heater, the Tennessee Eastman process and a practical coal pulverizing system.

1. Introduction

Industrial processes often operate under multiple modes owing to materials, product specifications, maintenance, etc. [1, 2, 3, 4]. Moreover, the systems are naturally dynamic in each mode, with the internal variables being time-correlated [5]. Multimode dynamic process monitoring methods have been actively studied and have been divided into two groups [6], namely, single-model methods and multiple-model ones. Single-model methods generally transform multiple distributed data into a uniform distribution [7] or update the parameters adaptively based on the forthcoming data [8], which are difficult to track normal variations between diverse modes.

Over the past several decades, multiple-model approaches have become a central branch of research in multimode monitoring. These approaches generally divide training data into several clusters offline and build local models correspondingly [9]. Then, the mode is identified online according to a decision function [10], or a global model is constructed by a weighted sum of local monitoring results based on Bayesian theory [11]. For instance, Wen *et al.* proposed the mixture of canonical variate analysis (MCVA) to monitor multimode dynamic processes [9], where data were divided into several clusters via Gaussian mixture models and a local canonical variate analysis model was built for each mode. Yao *et al.* presented a parallel semi-supervised Gaussian mixture model for multimode hierarchical quality monitoring [11], in which a quality regression model was built to deal with multimode big data and a global monitoring model was established based on Bayesian fusion. The aforementioned methods required that data from all potential modes have already been received for training, so the resulting model may have to be retrained from scratch when a new mode arises. Similar modes may be misidentified [12], which may degrade the monitoring performance. Moreover, normal data from all potential modes are required to be available for future learning, which may lead to high computational and storage costs. However, new modes appear continuously and thus it is impractical to store complete data in practical industrial systems. Therefore, it is desirable to investigate effective methods which are capable of monitoring successive modes with limited computing and storage resources.

Continual learning has become increasingly popular, particularly in the field of image processing [13, 14]. The features are continually extracted from a stream of data and the previously learned knowledge is accumulated for future learning. One consistent challenge is catastrophic forgetting, specifically when training a model using new features would interfere with the learned knowledge [13]. As summarized and discussed in [13], recent progresses in continual learning can be categorized into three families based on how the previous data are used: regularization-based approaches, replay approaches and parameter isolation approaches. Parameter

*Corresponding author
zjx18@tsinghua.org.cn (J. Zhang); x.hong@reading.ac.uk (X. Hong)
ORCID(s):

isolation approaches establish different model parameters to each mode to avoid catastrophic forgetting issue. The previous mode parameters would be frozen [15] or a mode copy may be dedicated to each mode [16]. To our best knowledge, parameter isolation methods are appropriate to overcome the forgetting of network-based methods, for instance, deep neural network [17], autoencoder [16] and so on. However, they have not been applied to multimode process monitoring.

Recently, regularization-based continual learning has been applied to multimode process monitoring [5, 18, 19], where a regularized penalty was added to make parameters change less between modes. One key requirement was the need to evaluate the importance of specific parameters accurately. For instance, a modified principal component analysis (PCA) with continual learning ability was first investigated to monitor successive modes in [19], and the importance of the PCA model parameters was evaluated by elastic weight consolidation (EWC) [20]. This method is abbreviated to PCA–EWC and is suitable for multimode stationary processes. Subsequently, a modified sparse dynamic inner PCA (SDiPCA) was proposed for multimode dynamic processes [5], and the aforementioned importance was measured by modified synaptic intelligence (MSI). This method was denoted as SDiPCA–MSI. However, aforementioned methods require that data from different modes share similar features [5, 19], for the previously learned knowledge to be effective for future modes. Briefly speaking, the regularization monitoring mechanism, which is applied to short-term tasks, may catastrophically degrade future performance due to unfamiliar forthcoming modes [21]. Therefore, regularization-based continual learning methods are greatly limited in practical applications as future diverse modes appear constantly. To alleviate this constraint, a multimode nonlinear SDiPCA (MNSDiPCA) was presented based on replay continual learning [22], where multimode features were extracted from raw data, and intended to be applied for long-term monitoring tasks.

Against this background, this work investigates an efficient multimode dynamic process monitoring method with continual learning ability, where data from multiple modes are collected sequentially. Specifically, attentional PCA (APCA) is proposed to characterize the relationship between dynamic variables, in which two attention mechanisms are investigated and adopted to model the dynamic latent variables, focusing on the high-value information from massive data using limited computing resources [23, 24]. It is proposed that replay data, sufficient to reflect the operating conditions, are selected at the end of each mode based on cosine similarity and stored for future learning. When a new mode arrives, inspired by replay continual learning [13], the current mode data are integrated with replay data and utilized to learn the model parameters, providing outstanding performance for all existing modes. This multimodal APCA method is abbreviated to M-APCA.

The contributions of this paper are outlined below:

a) A novel APCA is presented for dynamic processes, whereby two attention mechanisms are adopted to focus on both local and global significant information, while dynamic features are extracted via a vector autoregressive model. Two unsupervised algorithms are introduced to pre-train the keys in APCA with analysis while respecting the motivations and computational complexity of the overall algorithm.

b) An efficient novel multimodal APCA with continual learning ability is proposed for successive dynamic modes, where data from multiple modes are collected sequentially. Compared with traditional multimode monitoring approaches [3, 1], only a small amount of historical data are stored and replayed for future learning, which allows it to consume a few storage and computing resources.

c) Different from PCA–EWC [19] and SDiPCA–MSI [5], since multimodal features are extracted from data in a raw format, M-APCA is free from the constraint of mode similarity and can be applied to long-term monitoring of diverse modes. In addition, it is robust to noise and may provide enhanced interpretability.

The remainder of this paper is organized below. Section 2 introduces APCA for a single mode dynamic process. Section 3 elaborates on the technical core of M-APCA including the replay data selection, the learning algorithm of two different attention mechanisms and the training and monitoring procedures of M-APCA. The proposed APCA and M-APCA are compared with several state-of-the-art methods to highlight its superiority in Section 4. The comparative methodology is designed in Section 5, and the effectiveness of M-APCA is demonstrated by its application for a continuous stirred tank heater (CSTH) case, the Tennessee Eastman process (TEP), and a practical coal pulverizing system. Section 6 is devoted to conclusions.

2. System model

2.1. Attentional PCA for single mode dynamic process

An attention function [24] consists of a query, key-value pairs and weightings, which has been widely utilized in natural language processing (NLP). The output is calculated by a weighted combination of the values, and the weight corresponding to each value is calculated by a compatibility function of the query with the corresponding key. Here, a novel APCA model is proposed in which a set of dynamical latent *attention* variables are constructed, followed by a vector autoregressive (VAR) model to characterize the dynamical relationship of *attention* variables.

Let $\boldsymbol{X} = \{\boldsymbol{x}_k\}$, $k = 1, \ldots, N$ as a time instance. N is the number of samples and $\boldsymbol{x} \in R^m$ is a sample query vector variable. Consider an attention function $\mathcal{F}$: $\boldsymbol{x} \longmapsto \phi(\boldsymbol{x})$, and $\phi(\boldsymbol{x}) = \{\phi_i(\boldsymbol{x})\} \in R^q$ given by

$$\phi_i(\boldsymbol{x}) = \texttt{Similarity}(\boldsymbol{x}, \boldsymbol{c}_i), i = 1, \ldots, q$$

where $\boldsymbol{C} = \{\boldsymbol{c}_i\}$, $i = 1, \ldots, q$ are a set of q keys. $\texttt{Similarity}(\boldsymbol{x}, \boldsymbol{c}_i)$ is a predetermined similarity metric

between a query and a set of keys in data space of $\boldsymbol{X}$. In this work we used two types of similarity $\phi_i(\boldsymbol{x})$. One of the most common similarity functions is used [23] given by

$$\phi_i(\boldsymbol{x}) = \frac{\boldsymbol{x}^T \boldsymbol{c}_i}{d} \tag{1}$$

where $d > 0$ is a scaling hyper-parameter. Alternatively, the negative Euclidean distance given by

$$\phi_i(\boldsymbol{x}) = -\frac{\|\boldsymbol{x} - \boldsymbol{c}_i\|^2}{d} \tag{2}$$

is also used. Attention is the mapping [25]

$$\texttt{Attention}(\boldsymbol{x}, \boldsymbol{C}, \boldsymbol{w}) = \sum_{i=1}^{q} \texttt{softmax}(\boldsymbol{x}, \boldsymbol{C})_i w_i \tag{3}$$

in which

$$\texttt{softmax}(\boldsymbol{x}, \boldsymbol{C})_i = \frac{\exp(\phi_i(\boldsymbol{x}))}{\sum_{i=1}^{M} \exp(\phi_i(\boldsymbol{x}))} \tag{4}$$

For convenience, $\texttt{Attention}(\boldsymbol{x}, \boldsymbol{C}, \boldsymbol{w})$ is denoted as t and the function of $\texttt{softmax}(\cdot)$ is denoted as $\boldsymbol{x}_\phi$. Similar to DiPCA [26], the proposed APCA aims to extract the most predictable information by a VAR model to characterize the dynamic relationship. The latent *attention* variables defined at time instant k as,

$$t_k = \boldsymbol{x}_{\phi,k}^T \boldsymbol{w} \tag{5}$$

where $\boldsymbol{w} = [w_1, \ldots, w_q] \in R^q$ is the weight vector with $\|\boldsymbol{w}\|_2 = 1$. Over a data set $\boldsymbol{X}$, the mapped data are denoted as $\boldsymbol{X}_\phi \in R^{N \times q}$ by using the above attention mechanism and the kth sample is denoted as $\boldsymbol{x}_{\phi,k}$ correspondingly.

Similar to DiPCA [26], the current latent *attention* variable is represented by the past ones, namely,

$$t_k = \sum_{j=1}^{s} \beta_i t_{k-j} + r_k \tag{6}$$

where r_k is the Gaussian white noise at kth instant and s is the order of the VAR model. According to (5) and (6), the prediction of the dynamic latent *attention* variables is described by [25]:

$$\begin{aligned} \hat{t}_k &= \sum_{j=1}^{s} \boldsymbol{x}_{\phi,k-j}^T \boldsymbol{w} \beta_j \\ &= \begin{bmatrix} \boldsymbol{x}_{\phi,k-1}^T & \cdots & \boldsymbol{x}_{\phi,k-s}^T \end{bmatrix} (\boldsymbol{\beta} \otimes \boldsymbol{w}) \end{aligned}$$

where $\otimes$ denotes the Kronecker product, $\boldsymbol{\beta} = \begin{bmatrix} \beta_1 & \cdots & \beta_s \end{bmatrix}^T$ and $\|\boldsymbol{\beta}\|_2 = 1$.

In the original work of [23, 24] for NLP, there is a need to treat triplets {query, key, value} as learnable variables, and the attention in a large transformer network. In this work, we used attention in a more simplistic form, since in dynamic system models, the observed data can be used directly as the query, rather than being transformed into an embedding as in a neural language model.

The proposed APCA method extracts the dynamic latent *attention* variables by maximizing the covariance between t_k and $\hat{t}_k$. Our learnable attention variables are $\boldsymbol{C}, \boldsymbol{w}$ and $\boldsymbol{\beta}$, with the objective of APCA being designed as

$$\begin{aligned} &\min \ \ J(\boldsymbol{w}, \boldsymbol{\beta}) = -\boldsymbol{w}^T \left(\boldsymbol{X}_\phi^{(s+1)} \right)^T \boldsymbol{Z} (\boldsymbol{\beta} \otimes \boldsymbol{w}) + \lambda_1 \boldsymbol{\beta}^T \boldsymbol{D} \boldsymbol{\beta} \\ &s.t. \ \ \|\boldsymbol{w}\|_2 = 1, \ \ \|\boldsymbol{\beta}\|_2 = 1 \end{aligned} \tag{7}$$

where $\boldsymbol{D}$ is a weighting matrix to make $\boldsymbol{\beta}$ sparse, and λ_1 is a predefined regularization coefficient. Sparse representation is utilized to avoid potential overfitting and further mitigate catastrophic forgetting [21]. $\boldsymbol{X}_\phi^{(s+1)}$ and $\boldsymbol{Z}$ are constructed by [25]

$$\boldsymbol{X}_\phi^{(j)} = \begin{bmatrix} \boldsymbol{x}_{\phi,j} & \boldsymbol{x}_{\phi,j+1} & \cdots & \boldsymbol{x}_{\phi,N-s+j-1} \end{bmatrix}^T, \ \ j = 1, \ldots, s+1 \tag{8}$$

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{X}_\phi^{(1)} & \boldsymbol{X}_\phi^{(2)} & \cdots & \boldsymbol{X}_\phi^{(s)} \end{bmatrix} \tag{9}$$

We observe that if the data set $\boldsymbol{X}$ only represents a single mode, (7) can be solved efficiently, provided $\boldsymbol{C}$ is fixed. This suggests that a hybrid algorithm can be used to obtain $\boldsymbol{C}$, followed by (7). However, if the data set $\boldsymbol{X}$ represents multiple modes, it is necessary to introduce the problem statement and outline the objective.

2.2. An outline of M-APCA problem statement

Consider the task of monitoring multimode dynamic processes, with each of the modes being denoted as $\mathcal{M}_K$, $K = 1, 2, \ldots$, and a respective typical data set $\boldsymbol{X}_K^0$. In contrast to the common approach of building local models for each mode then combining them as a global model, a single adaptive model is obtained such that after the model is updated by any new $\mathcal{M}_K$, all previous modes up to $\mathcal{M}_{K-1}$ can still be represented by the model. The objective is to build a single model for monitoring multimode dynamic processes based on APCA, with good performance for all previous modes and within acceptable costs for replay data storage.

In order to achieve tractability, computational efficiency and algorithmic simplicity, a hybrid learning algorithm is designed in stages, so that the problem is decomposed into sequential tractable problems. The proposed M-APCA is as shown in Figure 1. Given $\mathcal{M}_K$, $K = 1, ...$, the training data contains $\boldsymbol{X}_K^0$ from $\mathcal{M}_K$, as well as replay data $\boldsymbol{\mathcal{D}}_K = \{\tilde{\boldsymbol{X}}_1, \ldots, \tilde{\boldsymbol{X}}_{K-1}\}$ from previous $K-1$ modes. Note that $\tilde{\boldsymbol{X}}_k$ $(k = 1, \ldots, K-1)$ from each previous mode are selected by cosine similarity in Section 3.1 and stored for future learning. The keys $\boldsymbol{C}^K$ of attention mechanisms are updated from a prior model parameter set $\boldsymbol{C}^{K-1}$ in Section 3.2. Then, the VAR parameters $\{\boldsymbol{w}^K, \boldsymbol{\beta}^K\}$ are optimized in Section 3.3.

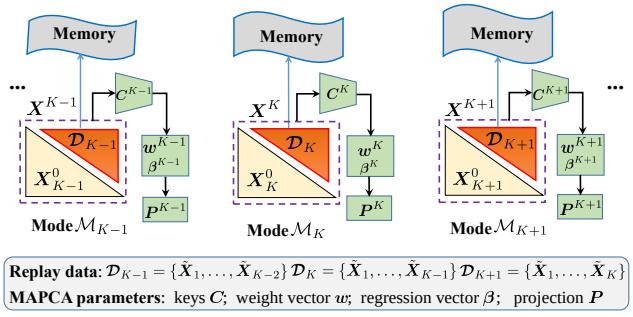


Figure 1: Illustration of M-APCA in three consecutive steps.

Table 1
Data illustration

Data	Illustration
$\boldsymbol{X}_K^0$	Sensing data solely collected for the Kth mode $\mathcal{M}_K$, $K = 1, 2, \ldots$
$\boldsymbol{X}_K$	Preprocess $\boldsymbol{X}_K^0$ with zero mean and unit variance, and get $\boldsymbol{X}_K$
$\tilde{\boldsymbol{X}}_K^0$	Replayed data selected from $\boldsymbol{X}_K^0$ based on cosine similarity
$\tilde{\boldsymbol{X}}_K$	Preprocess $\tilde{\boldsymbol{X}}_K^0$ with zero mean and unit variance, and get $\tilde{\boldsymbol{X}}_K$
$\boldsymbol{\mathcal{D}}_K$	Replay data of previous $K-1$ modes
$\boldsymbol{X}^K$	Data $\boldsymbol{X}^K = \{\boldsymbol{\mathcal{D}}_K, \boldsymbol{X}_K\}$ are constructed from the existing K modes and used for training

A projection matrix $\boldsymbol{P}^K$ is acquired to monitor the existing K modes (for details see Section 3.4).

As a novel method of applying attention mechanism to nonlinear dynamical systems, this work assumes that data from multiple modes are collected sequentially. It is also assumed that any incoming new mode needs to be notified, which is still a limitation of the proposed method shared with other multimode monitoring approached [5, 19, 22]. Nevertheless, the dynamics of future modes are assumed to be unknown which are not predetermined. In practical applications, the mode label $\mathcal{M}_K$ is defined *in situ* as it arrives. Therefore, there is no constraint on the number of multiple modes. In other words, the proposed M-APCA method allows different modes to arrive continually in future and the model is updated when a new mode appears. Meanwhile, similar to DiPCA [26], the dynamic latent features and static features are extracted simultaneously with no need of assumptions to the dynamics for multimode processes. In the case that the automatically determined multiple modes share a certain degree of similarity, they will furnish diverse dynamic characteristics to the model and enhance the monitoring performance.

3. Proposed M-APCA algorithm

In this section, the details of M-APCA are introduced to monitor successive dynamic modes. We start with outlining the technical aspects which are: replay data selection, training data preparation and followed by the learning of keys $\boldsymbol{C}^K$ in the attention mechanism in an unsupervised manner for each mode. Two algorithms have been proposed for the learning of keys based on (1) and (2) respectively, followed by the proposed M-APCA in which the objective is settled by alternating direction of method of multipliers (ADMM) [27]. Finally, the offline training and online monitoring phases are outlined.

3.1. Training data preparation

While data replay is instrumental for continual learning, the constraints of efficiency (storage and computing costs) are met by the proposed algorithm in order to minimize the replay data size. The subset of data is selected as follows: Define multiple modes as $\mathcal{M}_K$, $K = 1, 2, \ldots$, which are normalized to zero means and unit variances, to yield $\boldsymbol{X}_K$. To facilitate exposition, assume that there are N_K samples in the data matrix $\boldsymbol{X}_K^0 \in R^{N_K \times m}$ for each mode $\mathcal{M}_K$, which are collected as normal data. Replay data are selected based on cosine similarity in order to represent the operating condition with minimal redundancy [22]. Data may contain different information, and should be selected and stored for future retraining when the $(K+1)$th mode arrives.

The proposed M-APCA algorithm is dependent on both the sequential current mode data (online) and the replay data in storage (offline). For clarity, the data preparation steps are presented. Recall at each mode $\mathcal{M}_K$, the original normal data set is $\boldsymbol{X}_K^0$, $K = 1, 2, \ldots$, and the normalized data are $\boldsymbol{X}_K$. Alternatively, at the end of each mode $\mathcal{M}_K$, replay data selection is carried out, and $\tilde{\boldsymbol{X}}_K^0$ is obtained based on cosine similarity, followed by normalization to zero means and unit variances, to yield $\tilde{\boldsymbol{X}}_K$. Thus, at mode $\mathcal{M}_K$, we have the normalized replay data $\boldsymbol{\mathcal{D}}_K = \{\tilde{\boldsymbol{X}}_1, \ldots, \tilde{\boldsymbol{X}}_{K-1}\}$ available for the past modes $\mathcal{M}_1, \ldots, \mathcal{M}_{K-1}$. Let $\boldsymbol{X}^K = \{\boldsymbol{\mathcal{D}}_K, \boldsymbol{X}_K\} \in R^{N^K \times m}$ be constructed, where N^K is the number of prepared training samples (combined current mode's and replay modes' data) that is ready to be employed in the training algorithms. The data have been illustrated in Table 1.

3.2. Pre-training algorithms for $\boldsymbol{C}^K$

At each mode, the proposed algorithm involves an unsupervised pre-training step to update $\boldsymbol{C}$ upon receiving new mode $\boldsymbol{X}_K^0$. We present two algorithms that are specific to Attention I and II as follows.

3.2.1. *Attention I based on (1) using a new maximum likelihood estimator*

Recall Attention I that is based on (1), and the attention mapping (3) is rewritten as

$$\texttt{Attention}(\boldsymbol{x}, \boldsymbol{C}, \boldsymbol{w}) = \sum_{i=1}^{q} \texttt{softmax}(\boldsymbol{x}, \boldsymbol{C})_i w_i \quad (10)$$

Note that $\texttt{softmax}(\boldsymbol{x}, \boldsymbol{C})_i$ can be interpreted as the probability of each key $\boldsymbol{c}_i$, with respect to its corresponding w_i (value in the attention mechanism). Intuitively, the set of keys $\boldsymbol{C}$ provides a parsimonious representation of $\boldsymbol{X}^K$, $K = 1, 2, \ldots$. We propose that this is obtained using a simple pre-training algorithm based on the maximum likelihood estimator (MLE) of joint probability of the prepared data set $\boldsymbol{X}^K$.

Algorithm 1 Updating $\boldsymbol{C}^K$ based on MLE

Require: Data $\boldsymbol{X}^K \in R^{N^K \times m}$, $\eta = 0.1$, error ϵ.

Ensure: $\boldsymbol{C}^K = \{\boldsymbol{c}_1^*, \dots, \boldsymbol{c}_q^*\}$ is obtained by maximizing $J = \sum_{k=1}^{N^K} J_k$.

1: Initialize $t = 1$, randomly select q samples from $\boldsymbol{X}^K$ to construct the initial $\boldsymbol{C}$, calculate the initial $J(0) = \sum_{k=1}^{N^K} J_k$ based on (11).
2: For each data $\boldsymbol{x}_k (k = 1, \dots, N^K)$, update $\boldsymbol{c}_i$ by (12), $i = 1, \dots, q$.
3: Calculate $J(t) = \sum_{k=1}^{N^K} J_k$, and J_k is calculated by (11).
4: Return to step 2 until $\|J(t) - J(t-1)\| < \epsilon$, let $t = t + 1$.
5: The optimal cluster centers are denoted as $\boldsymbol{C}^K = \{\boldsymbol{c}_1^*, \dots, \boldsymbol{c}_q^*\}$.

Consider the instantaneous log-likelihood function

$$\begin{aligned} J_k &= \sum_{i=1}^{q} \log \texttt{softmax}(\boldsymbol{x}_k, \boldsymbol{C})_i \\ &= \sum_{i=1}^{q} \phi_i(\boldsymbol{x}_k) - q \log\{\sum_{i=1}^{q} \exp(\phi_i(\boldsymbol{x}_k))\} \end{aligned} \tag{11}$$

We randomly initialize $\boldsymbol{C}$, then over the data samples index k, J_k is maximized by adjusting $\boldsymbol{C}$ jointly, using the gradient ascent algorithm subject to the constraint $\sum_i \|\boldsymbol{c}_i\| = q$, which is necessary to avoid the magnitude of $\boldsymbol{c}_i$ growing to infinity. We then have

$$\begin{aligned} \boldsymbol{c}_i^{new} &= \boldsymbol{c}_i^{old} + \eta \delta \boldsymbol{c}_i \\ \boldsymbol{c}_i^{new} &= \boldsymbol{c}_i^{new} / \sum_i \|\boldsymbol{c}_i^{new}\|, \end{aligned} \tag{12}$$

with $\delta \boldsymbol{c}_i = \frac{\partial}{\partial \boldsymbol{c}_i} J_k = (\boldsymbol{x}_k - q\texttt{softmax}(\boldsymbol{x}, \boldsymbol{C})_i \boldsymbol{x}_k)/d$ for all i, where $\eta > 0$ is a small preset learning rate.

Clearly (11) is data dependent and will only perform well over the given training data set. Hence in order to ensure continual learning for multimode data sets, the joint training of current mode data together with replayed data is proposed (see data preparation of $\boldsymbol{X}^K$ in Section 3.1). At each mode, the algorithm of updating $\boldsymbol{C}^K = \{\boldsymbol{c}_1, \dots, \boldsymbol{c}_q\}$ in Attention I is summarized in Algorithm 1. For the mode $\mathcal{M}_1$, let $\boldsymbol{X}^1 = \boldsymbol{X}_1$. Since replayed data from previous modes are used, the key parameter $\boldsymbol{C}^K$ should capture significant information of all previous modes based on the maximum likelihood criterion.

3.2.2. Attention II based on (2) using k-means clustering algorithm

Whilst the maximum likelihood estimator provides a general method for any attention mechanism. Here, a simple method is proposed based on heuristics specific to the model in the form of Attention II. Recall Attention II based on (2), and attention mapping turns out to be,

$$\texttt{Attention}(\boldsymbol{x}, \boldsymbol{C}, \boldsymbol{w}) = \sum_{i=1}^{q} \frac{\exp(\frac{-\|\boldsymbol{x}-\boldsymbol{c}_i\|^2}{d})}{\sum_{i=1}^{q} \exp(\frac{-\|\boldsymbol{x}-\boldsymbol{c}_i\|^2}{d})} w_i \tag{13}$$

We propose to adopt the well known online k-means algorithm [28] to train $\boldsymbol{C}^K$ [25]. The objective of k-means algorithm can be described as

$$L = \min_{i=1}^{q} \sum_{k=1, \boldsymbol{x}_k \in S_i}^{N_K} \left\|\boldsymbol{x}_k - \boldsymbol{c}_i\right\|^2 \tag{14}$$

where S_i, $i = 1, \dots, q$ divides data into q disjoint clusters. When the Kth mode is encountered, the clustering centers are updated based on the current mode data $\boldsymbol{X}_K$ and the key $\boldsymbol{C}^{K-1}$ of last mode. The procedure of online k-means algorithm can refer to [28], where the initial cluster centers are $\boldsymbol{C}^{K-1} = \{\boldsymbol{c}_1, \dots, \boldsymbol{c}_q\}$ and the optimal clustering centers are denoted as $\boldsymbol{C}^K = \{\boldsymbol{c}_1^*, \dots, \boldsymbol{c}_q^*\}$. For the first mode, $\boldsymbol{C}^0$ is selected randomly from $\boldsymbol{X}_1$.

The rationale that k-means clustering algorithm can be used for Attention II is explained as follows: Although (14) is not a probabilistic measure, it can achieve the similar goal of identifying keys (centers). In fact, for any data $\boldsymbol{x}$, the resulting probability if Attention II is used, $\texttt{softmax}(\boldsymbol{x}, \boldsymbol{C})_{\texttt{i}}$, can be sorted in the same order of the Euclidean distance between $\boldsymbol{x}$ and $\boldsymbol{c}_i$, $i = 1, \dots, q$. Since the closed form solution of (14) is the mean of the data points in the cluster, it is expected that these are well suited for the learning keys in Attention II. While the replay data are not used in Attention II, the continual learning ability is maintained via initialization between successive modes.

Remarks

- *Motivation of two methods*. The two attention models arise from different similarity metrics between query and key in the attention models. For example, Attention I adopts cross product to measure directional similarity, which is more useful than Attention II for very high-dimensional correlated data sets. Other Attention model forms can be extended from this MLE framework, potentially leading to metric learning. For Attention II, Euclidean distance is utilized to measure similarity, which may occur the curse of dimensionality in high-dimensional space. Then, k-means clustering algorithm is utilized to update the keys, which is simple and converges quickly. Other online clustering algorithms can also be adopted to update the clustering centers.

- *The differences of two aforementioned methods*. The online k-means clustering algorithm is used in Attention II, in which the key $\boldsymbol{C}^K$ is updated based on data $\boldsymbol{X}_K$ and $\boldsymbol{C}^{K-1}$, but without using data replay. The reason of not using the replay data is that the k-means clustering algorithm typically does not to forget about previous modes due to the fact that at each iteration, only the clusters closest to the new mode data are updated, indicating some centers that have been learned in previous modes will basically remain unchanged in the case of novel modes (due to initialization using previous centers). Thus, the overall cluster centers can reflect the information of previous modes as well as new mode as K increases.

- *Justification of the pre-training algorithms*. One of the novelties of the proposed methods is the model structure where the latent dynamic variables are in the form of an attention mechanism. This attention mechanism can also be interpreted as projecting data to nonlinear space to capture the underlying nonlinearities from the viewpoint of approximation theory. Moreover, according to the relationship between a query and a set of keys in the attention mechanism, the proposed algorithm to learn $\boldsymbol{C}^K$ in each mode is justified here: (i) Since a single model is built for all modes, the key $\boldsymbol{C}^K$ should be updated adaptively, but also continually so that it represents the compressed information of all modes; (ii) Without $\boldsymbol{C}^K$ being fixed appropriately according to the data distribution, the solution to (7) can become intractable for streaming data applications. Other methods such as stochastic gradient descent algorithm that are typically used in attention mechanism estimation [23] will be slower to converge in real time industrial process monitoring applications.

- *The keys and queries between APCA and NLP.* In NLP, a text embedding is a piece of text projected into a high-dimensional latent space, attention mechanism key and query are based on latent space, rather than original data. In our APCA approach the data is in real space, no embedding is used. For Attention I, the keys are estimated by maximizing (11) and the final results are iterated by (12). In a certain sense, the keys are a nonlinear transformation of original data. With regard to Attention II, the keys are clustering centers of k-means algorithm, which are essentially the linear transformation of original data. Generally, q is set to be large to leave space for future modes.

3.3. Objective and solutions

When the mode $\mathcal{M}_K$ arrives, motivated by replay continual learning, construct data $\boldsymbol{X}^K = \left\{\boldsymbol{D}_K, \boldsymbol{X}_K\right\}$ to build a single monitoring model for multiple modes. The key $\boldsymbol{C}^K$ is pre-trained via Algorithm 1 or online k-means clustering algorithm. Map data $\boldsymbol{X}^K$ to a high-dimensional space by (3)–(4), and then calculate the mean $\boldsymbol{\mu}_K^{\phi}$ and variance $\boldsymbol{\Sigma}_K^{\phi}$. The pre-processed data are denoted as $\boldsymbol{X}_{\phi,K}$ with zero mean and unit variance. Similar to (8) and (9), construct $\boldsymbol{X}_{\phi,K}^{(j)}$ $(1 \le j \le s+1)$ and $\boldsymbol{Z}_K = \left[\boldsymbol{X}_{\phi,K}^{(1)} \;\; \boldsymbol{X}_{\phi,K}^{(2)} \;\; \cdots \;\; \boldsymbol{X}_{\phi,K}^{(s)}\right]$. M-APCA aims to build one model for sequential modes with acceptable storage and computing costs. For all K modes, the objective function of M-APCA is designed as

$$J_K(\boldsymbol{w}, \boldsymbol{\beta}) = -\boldsymbol{w}^T \left(\boldsymbol{X}_{\phi,K}^{(s+1)}\right)^T \boldsymbol{Z}_K \left(\boldsymbol{\beta} \otimes \boldsymbol{w}\right) + \lambda_1 \boldsymbol{\beta}^T \boldsymbol{D} \boldsymbol{\beta} \tag{15}$$

with the constraint $\boldsymbol{w}^T\boldsymbol{w} = 1$, $\boldsymbol{\beta}^T\boldsymbol{\beta} = 1$.

The parameters $\boldsymbol{w}$ and $\boldsymbol{\beta}$ are optimized alternatively by ADMM[27]. The weighting matrix $\boldsymbol{D}$ is updated after each iteration [29]. Assuming that $\boldsymbol{w}^i$, $\boldsymbol{z}_w^i$, $\boldsymbol{u}_w^i$, $\boldsymbol{\beta}^i$, $\boldsymbol{z}_\beta^i$ and $\boldsymbol{u}_\beta^i$ are available after the ith iteration, the updating procedure at $(i+1)$th iteration is summarized as follows:

1) Update parameters about $\boldsymbol{w}$

$$\begin{aligned} \arg\min_{\boldsymbol{w}} \;\; & J_K(\boldsymbol{w}, \boldsymbol{\beta}^i) \\ s.t. \;\; & \boldsymbol{w}^T\boldsymbol{w} = 1 \end{aligned} \tag{16}$$

According to Chapter 9 in [27], the parameters are updated by:

$$\boldsymbol{w}^{i+1} := \arg\min_{\boldsymbol{w}} \left(J_K(\boldsymbol{w}, \boldsymbol{\beta}^i) + \rho_w \left\|\boldsymbol{w} - \boldsymbol{z}_w^i + \boldsymbol{u}_w^i\right\|_2^2\right)$$

$$\boldsymbol{z}_w^{i+1} := \frac{\boldsymbol{w}^{i+1} + \boldsymbol{u}_w^i}{\|\boldsymbol{w}^{i+1} + \boldsymbol{u}_w^i\|} \tag{17}$$

$$\boldsymbol{u}_w^{i+1} := \boldsymbol{u}_w^i + \boldsymbol{w}^{i+1} - \boldsymbol{z}_w^{i+1} \tag{18}$$

where the regularization coefficient ρ_w is predefined. Take the derivative with regard to $\boldsymbol{w}$ and let it be zero, then

$$\boldsymbol{w}^{i+1} = 2\rho_w \left(\boldsymbol{G}_{\beta,K} + \boldsymbol{G}_{\beta,K}^T - 2\rho_w \boldsymbol{I}_M\right)^{-1} \left(\boldsymbol{u}_w^i - \boldsymbol{z}_w^i\right) \tag{19}$$

where $\boldsymbol{G}_{\beta,K} = \sum_{j=1}^{s} (\boldsymbol{X}_{\phi,K}^{(s+1)})^T \boldsymbol{X}_{\phi,K}^{(j)} \beta_j$.

2) Update parameters about $\boldsymbol{\beta}$

$$\begin{aligned} \arg\min_{\boldsymbol{\beta}} \;\; & J_K(\boldsymbol{w}^{i+1}, \boldsymbol{\beta}) \\ s.t. \;\; & \boldsymbol{\beta}^T\boldsymbol{\beta} = 1 \end{aligned} \tag{20}$$

Here, ADMM has the form [27]:

$$\boldsymbol{\beta}^{i+1} := \arg\min_{\boldsymbol{\beta}} \left(J_K(\boldsymbol{w}^{i+1}, \boldsymbol{\beta}) + \rho_\beta \left\|\boldsymbol{\beta} - \boldsymbol{z}_\beta^i + \boldsymbol{u}_\beta^i\right\|_2^2\right)$$

$$\boldsymbol{z}_\beta^{i+1} := \frac{\boldsymbol{\beta}^{i+1} + \boldsymbol{u}_\beta^i}{\|\boldsymbol{\beta}^{i+1} + \boldsymbol{u}_\beta^i\|} \tag{21}$$

$$\boldsymbol{u}_\beta^{i+1} := \boldsymbol{u}_\beta^i + \boldsymbol{\beta}^{i+1} - \boldsymbol{z}_\beta^{i+1} \tag{22}$$

where ρ_β is a predefined coefficient. Take the derivative with regard to $\boldsymbol{\beta}$ and let it be zero, then

$$\begin{aligned} \boldsymbol{\beta}^{i+1} = & \left(\lambda_1 \boldsymbol{D}^i + \rho_\beta \boldsymbol{I}_s\right)^{-1} \\ & \left(\frac{1}{2}\left(\boldsymbol{I}_s \otimes \boldsymbol{w}^{i+1}\right)^T \boldsymbol{G}_K^T \boldsymbol{w}^{i+1} + \rho_\beta \left(\boldsymbol{z}_\beta^i - \boldsymbol{u}_\beta^i\right)\right) \end{aligned} \tag{23}$$

where $\boldsymbol{G}_K = \left(\boldsymbol{X}_{\phi,K}^{(s+1)}\right)^T \boldsymbol{Z}_K$ and $\left(\boldsymbol{I}_s \otimes \boldsymbol{w}^{i+1}\right)^T \boldsymbol{G}_K^T = \left[\left(\boldsymbol{X}_{\phi,K}^{(s+1)}\right)^T \boldsymbol{X}_{\phi,K}^{(1)} \boldsymbol{w}^{i+1} \;\; \cdots \;\; \left(\boldsymbol{X}_{\phi,K}^{(s+1)}\right)^T \boldsymbol{X}_{\phi,K}^{(s)} \boldsymbol{w}^{i+1}\right]^T$,.

Algorithm 2 The pseudocode of M-APCA

Require: Normalized data of Kth mode $\boldsymbol{X}_K$, $\boldsymbol{X}^K$, key $\boldsymbol{C}^{K-1}$, the number of dynamic latent variables l, order of VAR model s.

Ensure: Key $\boldsymbol{C}^K$, mean $\boldsymbol{\mu}_K^{\phi}$, covariance $\boldsymbol{\Sigma}_K^{\phi}$, weight matrix $\boldsymbol{W}^K$, regression coefficient $\boldsymbol{\Gamma}^K$, projection matrix $\boldsymbol{P}^K$, latent variable matrix $\boldsymbol{T}$.

1: Pre-train the key $\boldsymbol{C}^K$ based on Algorithm 1 or online k-means clustering algorithm.
2: Map data $\boldsymbol{X}^K$ to a high-dimensional feature space by key $\boldsymbol{C}^K$ and (4), and the mapped data are $\boldsymbol{X}_{\phi,K}^0$. Calculate mean $\boldsymbol{\mu}_K^{\phi}$ and covariance $\boldsymbol{\Sigma}_K^{\phi}$, and the pre-processed data are labeled as $\boldsymbol{X}_{\phi,K}$.
3: Construct $\boldsymbol{X}_{\phi,K}^{(j)}$ $(1 \le j \le s+1)$ and $\boldsymbol{Z}_K$ by (8) and (9), let $g = 1$.
4: Initialize $\boldsymbol{w}^0$ and $\boldsymbol{\beta}^0$ with unit vector, $\boldsymbol{z}_w^0 = \boldsymbol{w}^0$, $\boldsymbol{u}_w^0 = \boldsymbol{0}$, $\boldsymbol{z}_\beta^0 = \boldsymbol{\beta}^0$, $\boldsymbol{u}_\beta^0 = \boldsymbol{0}$, $i = 0$.
5: Extract the dynamic component one by one:
   a) Calculate $\boldsymbol{z}_w^{i+1}$, $\boldsymbol{u}_w^{i+1}$ and $\boldsymbol{w}^{i+1}$ by (17)–(19);
   b) Calculate $\boldsymbol{z}_\beta^{i+1}$, $\boldsymbol{u}_\beta^{i+1}$ and $\boldsymbol{\beta}^{i+1}$ by (21)–(23);
   c) Update the weighting matrix $\boldsymbol{D}$ by (24);
   d) Calculate the objective function (15). Let $i = i + 1$, return to step 5a) until convergence.
6: The optimal parameters are denominated as $\boldsymbol{w}_g$ and $\boldsymbol{\beta}_g$, let $\boldsymbol{t}_g = \boldsymbol{X}_{\phi,K}\boldsymbol{w}_g$.
7: Calculate the loading vector $\boldsymbol{p}_g = \frac{\boldsymbol{X}_{\phi,K}^T \boldsymbol{X}_{\phi,K}\boldsymbol{w}_g}{\boldsymbol{w}_g^T \boldsymbol{X}_{\phi,K}^T \boldsymbol{X}_{\phi,K}\boldsymbol{w}_g}$.
8: Deflate $\boldsymbol{X}_{\phi,K}$ as $\boldsymbol{X}_{\phi,K} = \boldsymbol{X}_{\phi,K} - \boldsymbol{X}_{\phi,K}\boldsymbol{w}_g\boldsymbol{p}_g^T$, the covariance $(\boldsymbol{X}_{\phi,K}^{(s+1)})^T \boldsymbol{X}_{\phi,K}^{(j)}$ is calculated by (25), $j = 1, \dots, s$.
9: Let $g = g + 1$, return to step 4 until extracting l dynamic components.
10: The parameters are denoted as $\boldsymbol{W}^K = \left[\boldsymbol{w}_1 \ \cdots \ \boldsymbol{w}_l\right]$, $\boldsymbol{\Gamma}^K = \left[\boldsymbol{\beta}_1 \ \cdots \ \boldsymbol{\beta}_l\right]$, $\boldsymbol{P}^K = \left[\boldsymbol{p}_1 \ \cdots \ \boldsymbol{p}_l\right]$ and $\boldsymbol{T} = \left[\boldsymbol{t}_1 \ \cdots \ \boldsymbol{t}_l\right]^T$.

3) Update $\boldsymbol{D}$

$$\begin{aligned} \boldsymbol{D}^{i+1} =& diag\left\{d_1^{i+1}, d_2^{i+1}, \cdots, d_s^{i+1}\right\} \\ d_j^{i+1} =& \frac{1}{\left|\beta_j^{i+1}\right| + \epsilon}, \quad j = 1, \cdots, s \end{aligned} \tag{24}$$

where ϵ is a small positive value to avoid ill-conditioning issue.

Algorithm 2 summarizes the procedure of M-APCA, where dynamic components are acquired sequentially. When $K = 1$, let $\boldsymbol{X}^1 = \boldsymbol{X}_1$ and Algorithm 2 is also applied. When $g \ge 2$, once a dynamic component is extracted, deflate $\boldsymbol{X}_{\phi,K}$ as $\boldsymbol{X}_{\phi,K} - \boldsymbol{X}_{\phi,K}\boldsymbol{w}_g\boldsymbol{p}_g^T$. Thus, $(\boldsymbol{X}_{\phi,K}^{(s+1)})^T \boldsymbol{X}_{\phi,K}^{(j)}$ is calculated recursively by

$$\begin{aligned} &(\boldsymbol{X}_{\phi,K}^{(s+1)})^T \boldsymbol{X}_{\phi,K}^{(j)} \\ =&\boldsymbol{p}_g^T\boldsymbol{w}_g^T(\boldsymbol{X}_{\phi,K}^{(s+1)})^T \boldsymbol{X}_{\phi,K}^{(j)}\boldsymbol{w}_g\boldsymbol{p}_g - (\boldsymbol{X}_{\phi,K}^{(s+1)})^T \boldsymbol{X}_{\phi,K}^{(j)}\boldsymbol{w}_g\boldsymbol{p}_g \\ &- \boldsymbol{p}_g^T\boldsymbol{w}_g^T(\boldsymbol{X}_{\phi,K}^{(s+1)})^T \boldsymbol{X}_{\phi,K}^{(j)} + (\boldsymbol{X}_{\phi,K}^{(s+1)})^T \boldsymbol{X}_{\phi,K}^{(j)} \end{aligned} \tag{25}$$

where $j = 1, \dots, s$, and (25) is adopted in (19) and (23).

3.4. M-APCA for multimode process monitoring

Similar to DiPCA, define the latent attention score $\boldsymbol{t}_g = \boldsymbol{X}_{\phi,K}\boldsymbol{w}_g$ and matrix $\boldsymbol{T} = \left[\boldsymbol{t}_1 \ \cdots \ \boldsymbol{t}_l\right]^T$, where $\boldsymbol{w}_g$ is generated from $\boldsymbol{W}^K$ with $g = 1, \dots, l$. Similar to (8), construct $\boldsymbol{T}_j$ from $\boldsymbol{T}$, $j = 1, \dots, s+1$. Then, the dynamic relations between $\boldsymbol{T}_{s+1}$ and $\boldsymbol{T}_1, \dots, \boldsymbol{T}_s$ can be represented by a VAR model [25], namely,

$$\begin{aligned} \boldsymbol{T}_{s+1} =& \boldsymbol{T}_1\boldsymbol{\Theta}_s + \boldsymbol{T}_2\boldsymbol{\Theta}_{s-1} + \cdots + \boldsymbol{T}_s\boldsymbol{\Theta}_1 + \boldsymbol{V} \\ =& \bar{\boldsymbol{T}}_{\boldsymbol{s}}\boldsymbol{\Theta} + \boldsymbol{V} \end{aligned}$$

where $\bar{\boldsymbol{T}}_{\boldsymbol{s}} = \left[\boldsymbol{T}_1 \ \boldsymbol{T}_2 \ \cdots \ \boldsymbol{T}_s\right]$ and $\boldsymbol{\Theta} = \left[\boldsymbol{\Theta}_s \ \boldsymbol{\Theta}_{s-1} \ \cdots \ \boldsymbol{\Theta}_1\right]$. The least squares estimate for $\boldsymbol{\Theta}$ is

$$\hat{\boldsymbol{\Theta}} = \left(\bar{\boldsymbol{T}}_{\boldsymbol{s}}^T\bar{\boldsymbol{T}}_{\boldsymbol{s}}\right)^{-1}\bar{\boldsymbol{T}}_{\boldsymbol{s}}^T\boldsymbol{T}_{s+1} \tag{26}$$

Then, the prediction of $\boldsymbol{T}_{s+1}$ is calculated by

$$\hat{\boldsymbol{T}}_{s+1} = \bar{\boldsymbol{T}}_{\boldsymbol{s}}\hat{\boldsymbol{\Theta}} \tag{27}$$

Since $\hat{\boldsymbol{T}}_{s+1}$ is generally dynamic, monitoring the latent variables directly would lead to high false alarm rates. To enhance the monitoring performance, define the dynamic residual $\boldsymbol{V}$:

$$\boldsymbol{V} = \boldsymbol{T} - \hat{\boldsymbol{T}}_{s+1} \tag{28}$$

which is usually stationary if the process operates in a normal condition. Then, this paper builds a monitoring statistic based on the Mahalanobis distance to evaluate the variation of dynamics [30]:

$$M_V^2 = (\boldsymbol{v} - \boldsymbol{\mu}_v)\boldsymbol{\Sigma}_V^{-1}(\boldsymbol{v} - \boldsymbol{\mu}_v)^T \tag{29}$$

where $\boldsymbol{\mu}_v$ and $\boldsymbol{\Sigma}_V$ are the mean value and covariance of $\boldsymbol{V}$. After extracting the dynamic features, the remaining features are static and the static prediction error is calculated by

$$\boldsymbol{E} = \boldsymbol{X}_{\phi}^{(s+1)} - \boldsymbol{T}_{s+1}(\boldsymbol{P}^K)^T \tag{30}$$

Similarly, an index is defined to measure the changes of static features:

$$M_E^2 = (\boldsymbol{e} - \boldsymbol{\mu}_e)\boldsymbol{\Sigma}_E^{-1}(\boldsymbol{e} - \boldsymbol{\mu}_e)^T \tag{31}$$

where $\boldsymbol{\mu}_e$ and $\boldsymbol{\Sigma}_E$ are the mean value and covariance of $\boldsymbol{E}$.

Since dynamic and static characteristics may exist simultaneously, two statistics (29) and (31) should be considered simultaneously. The thresholds of two monitoring statistics are determined by kernel density estimation (KDE) [31]. If two statistics are lower than their corresponding thresholds, the process operates normally; otherwise, a fault is detected and an alarm is triggered. The flowchart of offline training and online monitoring phases is depicted in Figure 2. Fault detection rate (FDR) and false alarm rate (FAR) are considered to evaluate the monitoring performance.

Remarks

In M-APCA modeling, the dynamic order s and the number of dynamic latent variables l need to be determined before optimizing the objective (15). The detailed estimation method can refer to DiPCA [26]. Once s is determined, 95% of auto-covariance is extracted by the first l dynamic latent variables. Therefore, l can be regarded as a function of s and

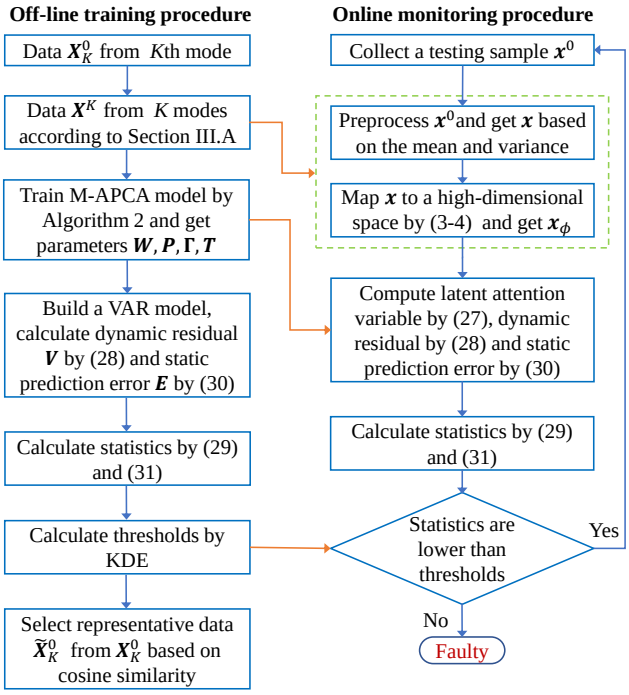


Figure 2: The flowchart of M-APCA for multimode dynamic process monitoring

be written as $l = l(s)$. It is desired that the prediction error matrix $\boldsymbol{E}$ contains little dynamic information after extracting l dynamic latent variables. To estimate an optimal s, an M-APCA model is trained first and then the prediction error matrix $\boldsymbol{E}$ is calculated based on the validation data. The sample crosscorrelation of any two variables in $\boldsymbol{E}$ should approximate 0, except when $s = 0$. The calculation of relevant confidence bounds could refer to [32]. When all pairs of variables are considered, the total violations of these confidence bounds are calculated for any $(s, l(s))$. The parameter $(s, l(s))$ is optimal when the corresponding violations are lowest. When s is too small, the VAR model (6) could not characterize the dynamic relationship. If the value of l is smaller than its real one, partial dynamic features would be contained in $\boldsymbol{E}$, which may cause high false alarms. When a new mode arrives, the parameters s and l need to be estimated again.

4. Discussion and comparative analysis

We discuss the relationship between APCA and self-attention PCA [33], DiPCA. Then, the association between the proposed M-APCA algorithm and recent monitoring methods with continual learning ability is discussed, including MNSDiPCA [22], SDiPCA–MSI [5] and PCA–EWC [19]. MCVA is utilized as a representative approach of traditional multimode process monitoring methods [9] and would be compared with approaches using continual learning.

4.1. APCA and self-attention PCA[33]

Both methods are proposed to focus on the local and global important information, and adapted to monitor a single dynamic mode. However, there are four distinct differences:

a) *Optimization objective*. Self-attention PCA extracts dynamic features via maximizing the variance of mapped data and the attention output acts as the input of the PCA-based process monitoring model. APCA is designed within the framework of DiPCA and maximizes the covariance between the latent variables and predictions.

b) *Ingredients of attention mechanism*. The query, keys and value of self-attention PCA are generated from the same sensing data. The keys and values of APCA are estimated by a certain rule, as described in Section 3.

c) *Similarity measure*. The dot product is adopted in self-attention PCA, while scaled dot-product and negative Euclidean distance are utilized to measure the similarity in APCA.

d) *Parameter estimation*. For self-attention PCA, the critical parameters are estimated by singular value decomposition and the number of principal components is estimated by cumulative percentage variance. However, the estimation of the arithmetic sequence is not provided. For APCA, the parameters are estimated by optimizing (15) when $K = 1$. The number of dynamic latent variables l and the order of VAR model s can refer to Section III.D.

4.2. APCA and DiPCA

For multimode processes, the keys $\boldsymbol{C}$ represent critical features and distribution of multimodal data. Through an attention mechanism, a subset of keys are adaptively selected and the most relevant information is concentrated. Compared with DiPCA, the 'position information' of each mode's data is considered automatically. Intuitively speaking, assume one sample $\boldsymbol{x}$ generated from mode $\mathcal{M}_K$ $(K = 1, 2, \ldots)$ and the key $\boldsymbol{c}_i$ $(i \in \{1, \ldots, q\})$ approaches the cluster center of mode $\mathcal{M}_K$, the mapped sample $\boldsymbol{x}_\phi(i)$ is highly significant after attention mapping. Therefore, M-APCA will provide excellent performance for sequential modes.

4.3. M-APCA and MNSDiPCA [22]

M-APCA and MNSDiPCA share partial common characteristics. First, they are both originally motivated from DiPCA and thus dynamic latent features are extracted via maximizing the covariance between the latent variable and its prediction. Replay continual learning is employed to overcome the catastrophic forgetting problem for multimode processes. The monitoring model is retrained based on the current mode data and the representative data from previous modes, which are selected from each mode based on cosine similarity. However, there exist two distinctions between M-APCA and MNSDiPCA:

a) *Data preprocessing manner*. MNSDiPCA maps data into a high-dimensional space via a polynomial function to settle the nonlinearity. M-APCA focuses on local and global important information and the dynamic relationship is characterized through the weight of attention mechanism, namely, $\texttt{softmax}(\boldsymbol{x}, \boldsymbol{C})$. Two alternative manners are presented to estimate the keys in the attention mechanism, and more motivation and theoretical analysis have been provided in Section 3.2.

b) *Dimension of processed data*. For MNSDiPCA, the dimension of the weights vector is fixed and there is a definite functional relationship with the dimension of measured data. For M-APCA, the dimension of the weights vector is equal to the number of keys q, which is arbitrary and determined by prior knowledge. Generally, q is set to be large enough to leave space for forthcoming novel dynamic modes.

4.4. M-APCA and regularization-based methods with continual learning ability

The relationship between M-APCA and regularization-based methods is discussed, including SDiPCA–MSI [5] and PCA–EWC [19]. Three methods with continual learning ability are investigated for mitigating the catastrophic forgetting problem of a single model for sequential modes. There are several differences as follows:

a) *The manner of preserving information from previous modes*. M-APCA adopts the principles of replay continual learning and extracts significant features of all modes from data in a raw format. With regard to SDiPCA–MSI and PCA–EWC, a quadratic regularization term is added to the loss function and the previously learned knowledge is consolidated by slowing down the learning rate of mode-sensitive parameters.

b) *Data requirement for training*. M-APCA selects and stores a few representative data from each mode, which are replayed together along with new mode data to establish a single model. When a new mode arrives, SDiPCA–MSI and PCA–EWC only utilizes the current mode data and the existing model parameters to build a monitoring model for successive modes. Since the training data are discarded once the learning process finishes, SDiPCA–MSI and PCA–EWC need less storage space than M-APCA.

c) *Applications*. Since M-APCA extracts features from all modes' sensing data, it can monitor diverse modes via a single model and can be applied to long-term monitoring tasks. SDiPCA–MSI and PCA–EWC require similarity among different modes and are appropriate for short-term monitoring tasks. Besides, M-APCA and SDiPCA–MSI are proper for multimode dynamic processes while PCA–EWC was investigated for multimode stationary processes.

Table 2
Comparison of online computational complexity

Methods	Complexity (*flam*)
M-APCA	$(m+2l+4)q+l^2s+3l+m$ or $(2m+2l+4)q+l^2s+3l+m$
MNSDiPCA	$\frac{m^4}{4}+\frac{5m^3}{2}+(\frac{25}{4}+l)m^2+(3l+6)m+(s+1)l^2+3l$
SDiPCA–MSI	$m^2+(l+3)m+(s+1)l^2+3l$
PCA–EWC	$2m^2+3m$
MCVA	$8(s^2m^2+sm)K+2K$

4.5. M-APCA and MCVA [9]

MCVA is one typical traditional multimode process monitoring method, where multimodal data are divided into several clusters and local monitoring models are built corresponding to each mode. It requires complete data and the model would be retrained from scratch using all normal data when a novel mode arrives. The storage and computational resources would increase with the successive emergence of novel modes in future. Different from MCVA, M-APCA with continual learning ability is free from the limitation of complete data and assumes that data from multiple modes are collected sequentially. When the model training finishes, a few representative data from this mode are selected to reduce redundancy and stored for future learning. When a new mode arrives, replay data from all previous modes and the current mode data are unified to construct a single model. Thus, M-APCA needs fewer storage sources than MCVA.

4.6. Online computational complexity

Online computational complexity is an important evaluation index of monitoring performance. The term *flam* is adopted to measure the complexity, which contains one addition and one multiplication [34]. For each testing sample, the preprocessing step needs m *flam*. For the attention mapping procedure, calculating $\phi(\boldsymbol{x})$ requires $(m+1)q$ *flam* if scaled dot-product is utilized and $(2m+1)q$ *flam* if negative Euclidean distance is utilized. Then, calculating (4) needs $2q$ *flam*. Calculating the latent attention variables and its prediction by (27) needs ql and l^2s *flam* respectively. Subsequently, calculating the dynamic residual by (28) and static prediction error by (30) requires l and $ql+q$ *flam*. Eventually, calculating two statistics needs $2(l+q)$ *flam*. In summary, the online complexity of M-APCA is $(m+2l+4)q+l^2s+3l+m$ *flam* using scaled dot-product, and $(2m+2l+4)q+l^2s+3l+m$ *flam* using negative Euclidean distance.

The online computational complexity of five methods is summarized in Table 2. When $q > \frac{m^2+3m}{2}$, the computational complexity of M-APCA is higher than that of MNSDiPCA. Note that the bound in the parameter l is different for these methods. For PCA–EWC and SDiPCA–MSI, $l \leq m$. Similarly, $l \leq \frac{m^2+3m}{2}$ for MNSDiPCA and $l \leq q$ for M-APCA. In contrast to these four methods, the complexity of MCVA will increase with the successive emergence of new modes.

5. Case studies

In this paper, four state-of-the-art methods are used for comparison with the proposed method. The effectiveness of the proposed method with two attention mechanisms is illustrated by a CSTH, the TEP and a practical coal pulverizing system. Besides, an ablation study is conducted to illustrate the necessity of attention mechanism.

5.1. Comparative experiments and setting

5.1.1. Comparative experiments

This paper considers four successive modes and the comparative experiments are designed in Table 5 and Table 7, where the training information, testing mode and the model label are listed. MNSDiPCA [22], SDiPCA–MSI [5], PCA–EWC [19] and MCVA [9] are compared with M-APCA, to highlight the continual learning ability for monitoring sequential modes. Specifically, two critical properties of continual learning, namely, forward transfer learning and backward transfer learning, can be reflected by the detection accuracy.

Situations 1–19 are designed to illustrate the continual learning ability of M-APCA and the catastrophic forgetting issue of APCA. The experiment schemes and the monitoring results with Attention I are listed in Table 5, where the similarity is measured by the scaled dot-product and a maximum likelihood estimator is utilized to estimate the keys. Table 6 summarizes the monitoring results using Attention II, where the negative Euclidean distance is the similarity metric and the keys are determined by an online k-means clustering algorithm. Note that Table 5 and Table 6 share the same simulation schemes and aim to illustrate effectiveness of the proposed method with two different attention mechanisms. Consider the first two modes as an example to depict the experiment scheme. When the first mode $\mathcal{M}_1$ has been trained, the representative data $\boldsymbol{\mathcal{D}}_2(\tilde{\boldsymbol{X}}_1)$ are selected based on cosine similarity, which are sufficient to represent the operating conditions of mode $\mathcal{M}_1$. When a new mode $\mathcal{M}_2$ arrives, data $\boldsymbol{X}_2$ are collected and utilized to update the key $\boldsymbol{C}$ by a maximum likelihood estimator or an online k means clustering algorithm. Then, $\boldsymbol{\mathcal{D}}_2$ and $\boldsymbol{X}_2$ are adopted to establish a M-APCA model, which furnishes the continual learning ability and aims to monitor two modes simultaneously. Furthermore, Situation 5 is designed to illustrate the catastrophic forgetting issue of APCA for multiple modes, namely, the features of mode $\mathcal{M}_1$ are overwritten when a new model is learned. As illustrated by Situations 6–19, when a new mode arrives, the scheme is designed in a similar way. M-APCA needs to store representative data $\boldsymbol{\mathcal{D}}$ from previous modes for future learning, thus it consumes moderate storage resources.

Schemes and monitoring results of comparative methods are listed in Table 7. Similar to [5, 19], Situations 20–49 are designed to illustrate the continual learning ability of MNSDiPCA, SDiPCA–MSI and PCA–EWC. Similar to M-APCA, MNSDiPCA adopted the replay continual learning, where the model is trained based on the current mode data and representative data when a new mode arrives. It is desired that the performance of Situations 20–29 is excellent. Assuming that modes arrive in a sequential manner, the monitoring model of SDiPCA–MSI and PCA–EWC is updated based on the current data and the model parameters of the previous modes. Since only data from the current mode are stored and are discarded when the training procedure finishes, SDiPCA–MSI and PCA–EWC require the least storage space among the five comparative methods. When the successive modes share similarity, the performance of Situations 30–49 may be satisfactory.

For this work, MCVA divided data into several clusters by Gaussian mixture model and a local CVA model was built within each cluster. Then, a global monitoring model was constructed based on a weighted sum of local models. For Situations 50–58, when a new mode is encountered, the MCVA model needs to be retrained from scratch, without any use of learned knowledge. It requires complete data from all potential modes for training, so the complete data from all previous modes must be stored. Thus, it has the greatest computation and storage requirements than other comparative methods.

5.1.2. Experimental setting

To enhance the data quality and ensure monitoring performance, several data preprocessing measures are utilized for three experiments, including data filtering to remove noise, dealing with outliers, selecting key variables. Besides, the mode labels of these experiments are available in advance and a new mode is judged by expert experience, prior knowledge and data characteristics.

To compare the monitoring performance conveniently, the setting of the critical parameters is discussed for the aforementioned methods. For M-APCA, SDiPCA–MSI and MNSDiPCA, the order of VAR model s and the number of dynamic latent variables l are key parameters and determined in Section 3.4. For MAPCA, the number of keys q is generally set to be large to leave more space for future modes and deal with nonlinearity. The hyper-parameter λ_1 is predefined by users and let λ_1 be 0.01 in this paper. For PCA–EWC, the number of principal components is estimated by cumulative variance contribution rate and its threshold is 0.85. For MCVA, one critical parameter is the number of local models, which is equivalent to the number of modes and is a priori in this paper. The monitoring thresholds of these statistics are calculated by KDE and the confidence level is 0.99. The detailed values of key parameters are summarized in Table 3.

5.2. CSTH

The CSTH process is a popular benchmark for multimode dynamic process monitoring, where hot water and cold water are mixed to meet the requirements [5, 6]. Water level, temperature and flow are controlled by PI controllers. For detailed description, one may refer to [35]. This paper considers two cases and the settings are summarized in Table 4, where data from each mode are collected in a sequential manner. Six key variables are adopted for monitoring. For

Table 3
The key parameter setting

Methods	CSTH	TEP	Coal pulverizing system
M-APCA	q = 16, l = 10, s = 3, d = 20	q = 66, l = 28, s = 3, d = 200	q = 20, l = 10, s = 3, d = 80
	q = 16, l = 10, s = 3, d = 16	q = 66, l = 30, s = 3, d = 200	q = 30, s = 3, l = 18, d = 180
MNSDiPCA	s = 3, l = 14	s = 3, l = 30	s = 3, l = 20
SDiPCA–MSI	s = 2, l = 3	s = 3, l = 18	s = 3, l = 7
MCVA	h = 2, l = 3	h = 2, l = 4	h =3 , l = 8

Table 4
Normal operating modes of CSTH

Cases	Mode label	Level SP	Temperature SP	Hot water valve
Case 1	$\mathcal{M}_1$	13	11	5
	$\mathcal{M}_2$	11	10.5	4
	$\mathcal{M}_3$	10	11	4
	$\mathcal{M}_4$	12	10.5	5
Case 2	$\mathcal{M}_1$	10	8	4
	$\mathcal{M}_2$	12	8	4
	$\mathcal{M}_3$	12	10.5	5.5
	$\mathcal{M}_4$	9	10.5	4.5

each mode, 1000 normal samples are collected for training and 1000 testing samples are generated below:

- Case 1: level is added by 0.1 from 501th sample;
- Case 2: temperature is added by 0.3 from 501th sample.

As listed in Table 5, M-APCA with Attention I can accurately monitor multiple modes based on a single model. For Case 1, the FDRs of M-APCA are 100% and the FARs are no more than 4.6%, meaning that M-APCA effectively enables monitoring of sequential modes. The FDRs of Situations 6 and 9 are 100% and 87.32%. This reflects the forward transfer learning ability of M-APCA, namely, the information from previous modes $\mathcal{M}_1$ and $\mathcal{M}_2$ could enhance the monitoring performance for future similar modes. Conversely, the FARs of Situations 10–11 and 17–19 are higher than 90%. This phenomenon indicates that APCA with Attention I suffers from the catastrophic forgetting issue for successive modes, where the monitoring model for a single mode fails to detect the fault in other modes. Similarly, for Case 2, the FDRs of the Situations 2, 3, 6–8 and 12–15 are higher than 99.80%, and the FARs are no more than 5.60%. However, the FARs of Situations 5, 10–11 and 17–19 are higher than 36%. In summary, M-APCA with Attention I can monitor successive dynamic modes based on a single model.

The benchmark results for M-APCA and APCA with Attention II are listed in Table 6. For Case 1 and Case 2, the FDRs of M-APCA are higher than 99% and the FARs are no more than 6.4%. However, the FARs of APCA are higher than 9% for Situations 5, 10–11, and 17–19. With regard to Case 2, the FARs of Situations 12 and 16 are 0.8% and 4.4% respectively, which reflects the forward transfer learning ability of M-APCA that the information from mode $\mathcal{M}_1$ enhances the monitoring performance of future similar modes. Overall, M-APCA with Attention II provides outstanding performance for successive modes.

The simulation consequences of comparative schemes are listed in Table 7. MNSDiPCA enables to monitor Case 1 and Case 2 accurately, where the FDRs are 100% and the FARs are no more than 5%. However, it may cost the most expensive computational resources among five methods. SDiPCA–MSI is capable of monitoring Case 1 accurately but fails to monitor Case 2, where the FARs of Situations 31, 32, 36–39 are higher than 13%. SDiPCA–MSI requires that data from multiple modes have a certain degree of similarity, so may be particularly inappropriate for Case 2. For Case 1, PCA–EWC fails to provide better detection performance than M-APCA since the FARs of of Situations 41–49 are higher than 8.40%. For Case 2, the FARs of PCA–EWC are higher than M-APCA in most situations. MCVA cannot monitor sequential modes accurately. The FDRs of Case 1 are lower than 89% and the FARs are higher than 12%. For Case 2, the FDRs of Situations 52, 53, 55 and 56 are lower than 93%.

The testing time of M-APCA with two attention mechanisms is less than 0.021 second and is similar for each situation, as listed in Table 8. The testing time of comparative methods is summarized in Table 9. It is obvious that the testing time of SDiPCA–MSI is lowest and just lower than M-APCA. The testing time of MNSDiPCA and PCA–EWC is higher than that of M-APCA, which would not increase with continuous emergence of new modes. MCVA costs the most expensive computational resources for online applications.

In summary, M-APCA with both Attention I or Attention II can provide superior monitoring performance compared to MNSDiPCA, SDiPCA–MSI, PCA–EWC and MCVA, in terms of detection accuracy and online computational complexity.

5.3. Tennessee Eastman process

The Tennessee Eastman process is a model of an industrial complex process and has been widely utilized to illustrate the effectiveness of multimode process monitoring methods [36]. For detailed information, refer to [37]. The data are generated from the Simulink model, which can be downloaded from http://depts.washington.edu/control/LARRY/TE/download.html. This paper considers four successive modes of process operation at three different G/H mass ratios, as listed in Table 10. 22 measured variables and 9 manipulated variables are utilized for monitoring. The

Table 5
Monitoring results (FDR(%) and FAR (%)) of M-APCA and APCA based on Attention I

	Methods	Training data	Testing mode	Model label	CSTH				TEP				Coal pulverizing system			
					Case 1		Case 2		Case 3		Case 4		Case 5		Case 6	
					FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR
Situation 1	APCA	$\boldsymbol{X}_1$	$\mathcal{M}_1$	$\mathcal{A}$	100	1.80	**99.80**	**0.80**	98.48	1.25	100	1.50	98.16	0	100	0.45
Situation 2	M-APCA	$\boldsymbol{X}_2$, $\boldsymbol{D}_2$	$\mathcal{M}_2$	$\mathcal{B}$	100	**0.60**	99.80	0.60	97.36	0.75	100	1.50	**100**	**0.37**	100	0.75
Situation 3	M-APCA	-	$\mathcal{M}_1$	$\mathcal{B}$	100	2.80	99.60	2.40	97.50	1.50	100	1.50	98.66	4.20	**100**	**0.27**
Situation 4	APCA	$\boldsymbol{X}_2$	$\mathcal{M}_2$	$\mathcal{C}$	100	0.80	100	0.80	99.01	2.50	100	3.25	100	1.99	100	0.75
Situation 5	APCA	-	$\mathcal{M}_1$	$\mathcal{C}$	100	6.60	100	36.20	98.62	3.00	100	3.25	100	38.66	100	30.27
Situation 6	M-APCA	$\boldsymbol{X}_3$, $\boldsymbol{D}_3$	$\mathcal{M}_3$	$\mathcal{D}$	**100**	**0.60**	100	1.20	98.81	0.25	**100**	**1.00**	93.97	1.50	**100**	**0.47**
Situation 7	M-APCA	-	$\mathcal{M}_1$	$\mathcal{D}$	100	2.80	98.79	0.80	97.63	1.25	100	1.25	**98.49**	**0**	**100**	**0.55**
Situation 8	M-APCA	-	$\mathcal{M}_2$	$\mathcal{D}$	100	4.40	100	1.40	97.30	2.00	100	2.75	100	2.24	**100**	**0.75**
Situation 9	APCA	$\boldsymbol{X}_3$	$\mathcal{M}_3$	$\mathcal{E}$	87.32	1.20	100	2.40	99.41	0.75	100	1.50	95.55	3.76	100	41.18
Situation 10	APCA	-	$\mathcal{M}_1$	$\mathcal{E}$	100	99.60	100	99.80	100	100	100	100	100	100	100	100
Situation 11	APCA	-	$\mathcal{M}_2$	$\mathcal{E}$	100	98.80	100	100	100	100	100	100	100	100	100	100
Situation 12	M-APCA	$\boldsymbol{X}_4$, $\boldsymbol{D}_4$	$\mathcal{M}_4$	$\mathcal{F}$	**100**	**0.80**	100	0.80	**99.01**	**0.25**	100	1.00	**98.18**	**0**	**100**	**0**
Situation 13	M-APCA	-	$\mathcal{M}_1$	$\mathcal{F}$	100	2.60	100	2.20	98.02	1.25	100	1.25	98.16	0	**100**	**0.18**
Situation 14	M-APCA	-	$\mathcal{M}_2$	$\mathcal{F}$	100	4.60	100	2.00	97.76	0.50	100	1.25	100	1.61	100	0.50
Situation 15	M-APCA	-	$\mathcal{M}_3$	$\mathcal{F}$	100	3.80	100	5.60	**99.14**	**0.75**	100	1.50	95.80	1.50	100	0.71
Situation 16	APCA	$\boldsymbol{X}_4$	$\mathcal{M}_4$	$\mathcal{G}$	100	0.80	100	2.40	**99.41**	**0.75**	100	1.50	98.48	1.55	100	0.31
Situation 17	APCA	-	$\mathcal{M}_1$	$\mathcal{G}$	100	97.40	100	100	100	100	100	100	100	100	100	100
Situation 18	APCA	-	$\mathcal{M}_2$	$\mathcal{G}$	100	93.60	100	100	100	100	100	100	100	100	100	100
Situation 19	APCA	-	$\mathcal{M}_3$	$\mathcal{G}$	100	97.00	100	100	100	100	100	100	100	100	100	100

Table 6
Monitoring results (FDR(%) and FAR (%)) of M-APCA and APCA based on Attention II

	Methods	CSTH				TEP				Coal pulverizing system			
		Case 1		Case 2		Case 3		Case 4		Case 5		Case 6	
		FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR
Situation 1	APCA	100	1.40	**100**	**1.60**	**98.48**	**1.00**	100	1.25	**98.66**	**0**	100	1.00
Situation 2	M-APCA	100	1.20	**100**	**0.60**	**97.56**	**0.50**	100	1.25	100	0.50	100	0.88
Situation 3	M-APCA	100	5.80	**100**	**2.20**	**97.56**	**0.50**	**100**	**0.50**	98.83	3.36	100	1.73
Situation 4	APCA	100	1.20	100	1.20	98.75	2.25	100	2.75	100	5.34	100	1.12
Situation 5	APCA	100	9.20	100	17.60	98.68	4.75	100	5.00	99.66	31.93	100	37.18
Situation 6	M-APCA	100	0.80	100	1.00	**99.21**	**0.75**	100	1.50	94.17	1.50	100	0.71
Situation 7	M-APCA	100	6.40	100	1.60	**97.63**	**0.75**	**100**	**0.75**	**98.49**	**0**	100	0.91
Situation 8	M-APCA	100	3.40	**100**	**0.80**	**97.50**	**0.75**	**100**	**1.50**	100	7.33	100	1.00
Situation 9	APCA	100	1.40	100	6.40	99.47	0.50	100	1.25	97.92	3.01	100	37.16
Situation 10	APCA	100	28.00	100	86.40	99.74	55.00	100	55.25	100	100	100	98.64
Situation 11	APCA	100	29.40	100	74.00	99.87	75.00	100	75.50	100	100	100	100
Situation 12	M-APCA	100	0.80	100	0.80	98.62	0.50	100	1.25	**98.18**	**0**	100	0.20
Situation 13	M-APCA	100	6.00	99.40	1.20	97.63	0.50	**100**	**0.50**	98.16	0	100	0.36
Situation 14	M-APCA	100	4.00	**100**	**0.60**	97.56	0.25	**100**	**1.00**	100	2.48	100	0.88
Situation 15	M-APCA	100	4.20	100	4.60	98.88	0.50	100	1.25	96.99	1.50	100	0.83
Situation 16	APCA	98.99	1.20	100	4.40	99.47	1.00	100	1.75	98.48	3.91	100	0.51
Situation 17	APCA	100	40.00	100	99.40	99.54	23.50	100	24.00	100	100	100	96.91
Situation 18	APCA	100	60.80	100	98.40	99.34	34.75	100	35.25	100	100	100	97.00
Situation 19	APCA	100	18.20	100	95.00	100	91.25	100	91.50	100	100	100	100

sampling time is 3 minutes. Two cases (Case 3 and Case 4) are considered and share the same training data, namely 1920 normal samples from each mode in Table 10. 1920 testing samples, including the first 400 normal samples and subsequent 1520 faulty samples, are generated from two typical faults, and the fault numbers are IDV(11) (Case 3) and IDV(14) (Case 4).

The monitoring results of M-APCA and APCA with Attention I and Attention II are listed in Table 5 and Table 6 respectively. M-APCA with Attention I could furnish continual learning ability and monitor Case 3 and Case 4 accurately, where the FDRs are higher than 97% and the FARs are lower than 3.0%. When a new mode arrives, a few data are replayed and utilized to train the M-APCA model, which can still provide similar monitoring performance with a single APCA-based monitoring model. For instance, the FDRs and FARs of Situations 12–15 are close to those of Situations 16, 1, 4 and 9. The FARs of Situations 10, 11, 17–19 are 100%, which reflects the catastrophic forgetting issue of APCA for multimode processes. The aforementioned analysis can equally be applied to M-APCA and APCA with Attention II, as listed in Table 6.

Table 7
Monitoring results (FDR(%) and FAR (%)) of comparative methods

	Methods	Training data	Testing mode	Model label	CSTH				TEP				Coal pulverizing system			
					Case 1		Case 2		Case 3		Case 4		Case 5		Case 6	
					FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR
Situation 20	MNSDiPCA	$\boldsymbol{X}_1$	$\mathcal{M}_1$	$\mathcal{H}$	100	2.60	100	2.60	99.14	1.75	100	2.00	98.16	0	100	5.09
Situation 21	MNSDiPCA	$\boldsymbol{X}_2, \boldsymbol{D}_2$	$\mathcal{M}_2$	$\mathcal{I}$	100	0.60	100	0.80	98.88	2.75	100	3.50	100	0.50	100	1.00
Situation 22	MNSDiPCA	-	$\mathcal{M}_1$	$\mathcal{I}$	100	3.40	100	2.20	98.48	2.00	100	2.25	98.83	5.88	100	13.82
Situation 23	MNSDiPCA	$\boldsymbol{X}_3, \boldsymbol{D}_3$	$\mathcal{M}_3$	$\mathcal{J}$	100	0.60	100	0.80	99.60	2.50	100	3.25	94.07	1.50	100	33.14
Situation 24	MNSDiPCA	-	$\mathcal{M}_1$	$\mathcal{J}$	100	2.80	**100**	**1.00**	98.35	1.75	100	2.00	**98.49**	**0**	100	2.27
Situation 25	MNSDiPCA		$\mathcal{M}_2$	$\mathcal{J}$	100	1.80	100	1.20	98.55	1.50	100	2.25	100	2.73	100	0.75
Situation 26	MNSDiPCA	$\boldsymbol{X}_4, \boldsymbol{D}_4$	$\mathcal{M}_4$	$\mathcal{K}$	100	0.80	100	0.60	**99.47**	**0.75**	100	1.25	**98.18**	**0**	**100**	**0**
Situation 27	MNSDiPCA	-	$\mathcal{M}_1$	$\mathcal{K}$	100	5.00	**100**	**0.60**	**98.22**	**1.00**	100	1.25	98.16	0	100	3.55
Situation 28	MNSDiPCA	-	$\mathcal{M}_2$	$\mathcal{K}$	100	3.80	**100**	**0.60**	98.35	1.75	100	2.25	100	1.37	100	0.88
Situation 29	MNSDiPCA	-	$\mathcal{M}_3$	$\mathcal{K}$	100	2.00	100	1.80	99.41	2.75	100	3.50	**96.94**	**0**	100	45.09
Situation 30	SDiPCA	$\boldsymbol{X}_1$	$\mathcal{M}_1$	$\mathcal{L}$	100	0.40	73.09	0.60	95.98	1.00	**100**	**1.00**	98.16	0.84	**100**	**0.09**
Situation 31	SDiPCA–MSI	$\boldsymbol{X}_2 + \mathcal{L}$	$\mathcal{M}_2$	$\mathcal{N}$	**100**	**0.40**	97.59	13.60	93.14	0.75	100	1.50	100	3.73	**100**	**0.37**
Situation 32	SDiPCA–MSI	-	$\mathcal{M}_1$	$\mathcal{N}$	**100**	**0.40**	91.16	35.60	95.45	3.00	100	3.00	**98.33**	**0**	100	2.27
Situation 33	SDiPCA–MSI	$\boldsymbol{X}_3 + \mathcal{N}$	$\mathcal{M}_3$	$\mathcal{O}$	**100**	**0.40**	100	5.80	97.56	0.75	100	1.50	93.04	1.50	100	39.05
Situation 34	SDiPCA–MSI	-	$\mathcal{M}_1$	$\mathcal{O}$	**100**	**0.40**	97.79	4.60	97.89	18.75	100	19.00	98.00	0	100	9.45
Situation 35	SDiPCA–MSI	-	$\mathcal{M}_2$	$\mathcal{O}$	**100**	**0.40**	98.39	1.00	98.02	30.35	100	30.75	100	1.86	100	33.25
Situation 36	SDiPCA–MSI	$\boldsymbol{X}_4 + \mathcal{O}$	$\mathcal{M}_4$	$\mathcal{P}$	**100**	**0.40**	100	16.40	93.54	0.75	100	1.25	98.30	1.79	99.78	0
Situation 37	SDiPCA–MSI	-	$\mathcal{M}_1$	$\mathcal{P}$	**100**	**0.40**	100	32.00	96.57	7.00	100	7.00	95.66	0	100	1.27
Situation 38	SDiPCA–MSI	-	$\mathcal{M}_2$	$\mathcal{P}$	**100**	**0.40**	100	34.40	95.06	3.00	100	3.75	100	6.34	100	17.13
Situation 39	SDiPCA–MSI	-	$\mathcal{M}_3$	$\mathcal{P}$	**100**	**0.40**	100	26.20	98.75	25.00	100	25.50	92.89	6.02	**100**	**0.59**
Situation 40	PCA	$\boldsymbol{X}_1$	$\mathcal{M}_1$	$\mathcal{Q}$	20.40	0	22.40	0	95.59	0	99.87	0	97.50	0	99.71	0.91
Situation 41	PCA–EWC	$\boldsymbol{X}_2 + \mathcal{Q}$	$\mathcal{M}_2$	$\mathcal{R}$	100	11.40	100	6.20	95.86	0.25	**100**	**0.75**	100	2.73	100	0.38
Situation 42	PCA–EWC	-	$\mathcal{M}_1$	$\mathcal{R}$	100	12.80	100	4.40	96.91	0.75	99.87	1.25	97.67	0	99.71	5.64
Situation 43	PCA–EWC	$\boldsymbol{X}_3 + \mathcal{R}$	$\mathcal{M}_3$	$\mathcal{S}$	100	10.00	100	8.60	98.36	0.25	97.50	0.50	**98.96**	**0**	100	41.18
Situation 44	PCA–EWC	-	$\mathcal{M}_1$	$\mathcal{S}$	100	13.60	100	6.40	97.30	3.25	99.34	3.50	97.67	0	100	29.45
Situation 45	PCA–EWC	-	$\mathcal{M}_2$	$\mathcal{S}$	100	11.40	100	6.40	96.97	10.25	100	10.25	**100**	**1.49**	100	1.75
Situation 46	PCA–EWC	$\boldsymbol{X}_4 + \mathcal{S}$	$\mathcal{M}_4$	$\mathcal{T}$	100	12.40	100	6.00	95.72	0.25	**100**	**0.25**	98.18	10.58	99.34	0
Situation 47	PCA–EWC	-	$\mathcal{M}_1$	$\mathcal{T}$	100	11.20	100	9.00	97.63	2.25	99.87	2.25	98.00	0	99.71	0.09
Situation 48	PCA–EWC	-	$\mathcal{M}_2$	$\mathcal{T}$	100	9.80	100	8.40	94.87	1.50	100	2.00	100	7.70	**100**	**0.38**
Situation 49	PCA–EWC	-	$\mathcal{M}_3$	$\mathcal{T}$	100	8.40	100	12.00	98.82	13.00	100	13.25	99.95	13.53	100	8.88
Situation 50	MCVA	$\boldsymbol{X}_1, \boldsymbol{X}_2$	$\mathcal{M}_1$	$\mathcal{U}$.	88.48	18.20	99.80	2.40	96.17	1.25	98.55	1.25	98.31	0	100	0.55
Situation 51	MCVA	-	$\mathcal{M}_2$	$\mathcal{U}$	81.62	17.00	100	1.00	96.11	1.25	99.93	2.00	100	22.48	100	3.25
Situation 52	MCVA	$\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3$	$\mathcal{M}_1$	$\mathcal{V}$	88.08	18.20	92.94	0.20	95.78	1.00	98.09	1.00	98.82	1.68	100	43.73
Situation 53	MCVA	-	$\mathcal{M}_2$	$\mathcal{V}$	81.41	17.00	89.72	0.20	95.84	1.00	99.87	1.75	100	21.49	100	5.12
Situation 54	MCVA	-	$\mathcal{M}_3$	$\mathcal{V}$	75.56	25.20	**100**	**0.20**	96.17	0	95.12	0.25	92.77	4.51	100	46.51
Situation 55	MCVA	$\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3, \boldsymbol{X}_4$	$\mathcal{M}_1$	$\mathcal{W}$	87.07	17.60	92.94	0.20	94.39	0.25	98.81	0.25	**98.48**	**0**	100	45.91
Situation 56	MCVA	-	$\mathcal{M}_2$	$\mathcal{W}$	78.99	16.80	89.52	0.20	93.60	0	**100**	**0.50**	**100**	**0.87**	100	5.25
Situation 57	MCVA	-	$\mathcal{M}_3$	$\mathcal{W}$	76.97	26.40	**100**	**0.20**	97.23	0.25	98.09	0.75	100	80.45	100	36.57
Situation 58	MCVA	-	$\mathcal{M}_4$	$\mathcal{W}$	78.18	12.60	**100**	**0.20**	98.15	1.00	100	1.25	99.09	54.03	100	0.41

The monitoring consequences of four comparative methods are listed in Table 7. MNSDiPCA can monitor Case 3 and Case 4 accurately, where the FDRs are higher than 98% and the FARs are no more than 3.50%. SDiPCA–MSI fails to deliver excellent performance for Case 3 and Case 4, where the FARs of Situations 34, 35 and 39 are not less than 19%. For PCA–EWC, the FARs of Situations 45 and 49 are higher than 10%. In other words, methods based on regularization continual learning could not offer desirable detection accuracy. Since PCA–EWC and SDiPCA–MSI required that multiple modes share similarity in a sense, this phenomenon may be caused by diverse modes. MCVA cannot provide better performance than M-APCA with either of the two attention mechanisms. With regard to Case 3, the FDRs of Situations 50–56 are lower than 97%.

As listed in Tables 8 and 9, the testing time of M-APCA, SDiPCA–MSI and PCA–EWC is similar, which indicates that the online computational complexity is close and could meet the real-time monitoring demand. The online computational complexity of MNSDiPCA is medium and the testing time is less than 2.1 seconds. Similar to CSTH case, the online computational complexity of MCVA is the highest and would still increase as K increases, which is in accordance with the theoretical analysis in Section 4.6.

In conclusion, M-APCA with two attention mechanisms and MNSDiPCA can deliver optimal performance in consideration of detection accuracy compared with PCA–EWC, SDiPCA–MSI and MCVA. However, M-APCA inherits other virtues due to attention mechanisms as mentioned in Sections 4.2 and 4.3. Besides, M-APCA is obviously less complicated than MNSDiPCA in this case. Thus, M-APCA with two attention mechanisms are optimal among five methods.

5.4. Coal pulverizing system

This paper adopts the coal pulverizing system to illustrate the effectiveness of M-APCA, which is one key unit of a 1030-MW ultra-supercritical thermal power plant in China. The coal pulverizing system is constructed by coal feeder, coal mill, rotary separator, raw coal hopper and stone coal scuttle, as shown in [5, 19]. To improve combustion

Table 8
Testing time (s) of M-APCA and APCA based on Attention I and Attention II

	Methods	CSTH				TEP				Coal pulverizing system			
		Case 1		Case 2		Case 3		Case 4		Case 5		Case 6	
		Attention I	Attention II	Attention I	Attention II	Attention I	Attention II	Attention I	Attention II	Attention I	Attention II	Attention I	Attention II
Situation 1	APCA	0.0180	0.0166	0.0195	0.0158	0.1158	0.1187	0.1014	0.1123	0.0206	0.0179	0.0319	0.0281
Situation 2	M-APCA	0.0164	0.0202	0.0148	0.0183	0.1034	0.1138	0.1038	0.1148	0.0298	0.0256	0.0287	0.0280
Situation 3	M-APCA	0.0151	0.0159	0.0155	0.0148	0.1031	0.1163	0.1044	0.1573	0.0175	0.0190	0.0317	0.0257
Situation 4	APCA	0.0153	0.0178	0.0148	0.0169	0.1053	0.1151	0.1030	0.1144	0.0294	0.0260	0.0303	0.0278
Situation 5	APCA	0.0150	0.0163	0.0167	0.0160	0.1875	0.2110	0.1873	0.2097	0.0180	0.0175	0.0310	0.0287
Situation 6	M-APCA	0.0143	0.0157	0.0175	0.0196	0.1058	0.1141	0.1014	0.1140	0.0722	0.0629	0.0212	0.0211
Situation 7	M-APCA	0.0148	0.0148	0.0148	0.0141	0.1033	0.1144	0.1022	0.1123	0.0188	0.0154	0.0279	0.0245
Situation 8	M-APCA	0.0176	0.0145	0.0147	0.0139	0.1014	0.1155	0.1029	0.1162	0.0244	0.0235	0.0284	0.0252
Situation 9	APCA	0.0144	0.0153	0.0146	0.0156	0.1032	0.1135	0.1022	0.1141	0.0663	0.0631	0.0217	0.0202
Situation 10	APCA	0.0149	0.0162	0.0162	0.0155	0.1892	0.2110	0.1956	0.2111	0.0175	0.0181	0.0302	0.0278
Situation 11	APCA	0.0161	0.0155	0.0162	0.0164	0.1889	0.2098	0.1891	0.2080	0.0331	0.0336	0.0302	0.0274
Situation 12	M-APCA	0.0154	0.0147	0.0164	0.0136	0.1060	0.1152	0.1063	0.1146	0.0995	0.0963	0.0292	0.0274
Situation 13	M-APCA	0.0141	0.0146	0.0143	0.0152	0.1065	0.1126	0.1029	0.1145	0.0162	0.0154	0.0283	0.0247
Situation 14	M-APCA	0.0141	0.0145	0.0147	0.0135	0.1028	0.1134	0.1050	0.1136	0.0253	0.0230	0.0272	0.0251
Situation 15	M-APCA	0.0148	0.0151	0.0149	0.0144	0.1032	0.1135	0.1039	0.1144	0.0638	0.0612	0.0189	0.0175
Situation 16	APCA	0.0158	0.0205	0.0203	0.0145	0.1053	0.1169	0.1024	0.1122	0.1126	0.0973	0.0297	0.0272
Situation 17	APCA	0.0156	0.0157	0.0165	0.0168	0.1885	0.2081	0.1880	0.2109	0.0177	0.0169	0.0318	0.0285
Situation 18	APCA	0.0155	0.0183	0.0163	0.0157	0.1950	0.2110	0.1887	0.2126	0.0317	0.0278	0.0313	0.0279
Situation 19	APCA	0.0156	0.0164	0.0160	0.0159	0.1904	0.2123	0.1024	0.2122	0.0694	0.0646	0.0208	0.0190

Table 9
Testing time (s) of comparative methods

	Methods	CSTH		TEP		Pulverizing system	
		Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
Situation 20	MNSDiPCA	0.1042	0.0964	1.4373	1.4750	0.1107	0.3337
Situation 21	MNSDiPCA	0.1008	0.0931	1.4996	1.3380	0.1277	0.3411
Situation 22	MNSDiPCA	0.1119	0.0943	1.3127	1.2847	0.1067	0.2940
Situation 23	MNSDiPCA	0.0942	0.0953	1.2864	1.2829	0.2179	0.2253
Situation 24	MNSDiPCA	0.0946	0.1097	1.3129	1.2911	0.1044	0.2867
Situation 25	MNSDiPCA	0.1021	0.0947	1.2883	1.2597	0.1259	0.2828
Situation 26	MNSDiPCA	0.0975	0.0957	1.3252	1.2668	0.2728	0.2800
Situation 27	MNSDiPCA	0.1288	0.0897	1.3639	1.9038	0.1310	0.2845
Situation 28	MNSDiPCA	0.0943	0.0976	1.3209	1.3456	0.1295	0.2785
Situation 29	MNSDiPCA	0.0942	0.0933	1.3500	2.0262	0.2171	0.2218
Situation 30	SDiPCA	0.0180	0.0206	0.1733	0.1259	0.0299	0.0213
Situation 31	SDiPCA–MSI	0.0092	0.0136	0.1143	0.0985	0.0179	0.0182
Situation 32	SDiPCA–MSI	0.0080	0.0141	0.1283	0.1292	0.0115	0.0111
Situation 33	SDiPCA–MSI	0.0054	0.0100	0.2262	0.1108	0.0804	0.0136
Situation 34	SDiPCA–MSI	0.0086	0.0072	0.1152	0.1132	0.0090	0.0160
Situation 35	SDiPCA–MSI	0.0055	0.0096	0.1126	0.1247	0.0150	0.0098
Situation 36	SDiPCA–MSI	0.0065	0.0091	0.1058	0.1461	0.1224	0.0153
Situation 37	SDiPCA–MSI	0.0064	0.0067	0.1058	0.1037	0.0103	0.0082
Situation 38	SDiPCA–MSI	0.0086	0.0068	0.1097	0.1011	0.0070	0.0094
Situation 39	SDiPCA–MSI	0.0065	0.0101	0.1125	0.1068	0.0534	0.0202
Situation 40	PCA	0.0998	0.0439	0.2593	0.1269	0.1006	0.1116
Situation 41	PCA–EWC	0.1134	0.0966	0.1234	0.1084	0.0996	0.1238
Situation 42	PCA–EWC	0.0998	0.0929	0.1054	0.1200	0.0834	0.1179
Situation 43	PCA–EWC	0.1270	0.0916	0.1027	0.1168	0.0987	0.1114
Situation 44	PCA–EWC	0.0921	0.0858	0.0975	0.1174	0.1036	0.0989
Situation 45	PCA–EWC	0.0987	0.2025	0.0964	0.1146	0.0942	0.1095
Situation 46	PCA–EWC	0.1078	0.0929	0.1050	0.1356	0.0932	0.1037
Situation 47	PCA–EWC	0.0925	0.0831	0.1904	0.1077	0.0997	0.1007
Situation 48	PCA–EWC	0.0826	0.1114	0.1680	0.1045	0.1080	0.0978
Situation 49	PCA–EWC	0.0968	0.0844	0.1159	0.1053	0.0871	0.0953
Situation 50	MCVA	0.3989	0.4146	2.6580	2.4805	0.3322	1.5023
Situation 51	MCVA	0.3673	0.3623	2.4690	2.3747	0.4502	1.4792
Situation 52	MCVA	0.4726	0.4362	3.4866	4.0478	0.4222	2.2105
Situation 53	MCVA	0.4638	0.4350	3.5590	4.1025	0.5713	2.1287
Situation 54	MCVA	0.4628	0.4242	3.4743	3.4800	1.0721	1.6441
Situation 55	MCVA	0.5868	0.5456	4.7801	4.5769	0.5157	2.7298
Situation 56	MCVA	0.6504	0.5378	4.7486	4.5575	0.7462	2.7629
Situation 57	MCVA	0.5694	0.5559	4.4991	4.5585	1.3898	2.0696
Situation 58	MCVA	0.6062	0.5321	4.5069	4.5449	1.8268	2.8189

efficiency and ensure the operating safety, this system grinds raw coal into pulverized coal with desired temperature and fineness [19, 25]. Two popular types of faults are investigated in this section, including the abnormalities from rotary separators (Case 5) and the coal feeders (Case 6). The variables are selected based on professional knowledge, the system theory and correlation analysis. Data information is summarized in Table 11.

The monitoring results of APCA and M-APCA with Attention I and Attention II are listed in Table 5 and Table 6 respectively, where a maximum likelihood estimator and online k-means clustering algorithm are adopted to estimate the keys in the attention mechanisms. As shown in Table 5, M-APCA with Attention I can detect the faults in multimode processes accurately. For Case 5, the FARs of M-APCA are lower than 4.3%. The FDRs of Situations 6, 9 and 15 are 93.97%, 95.55% and 95.80%, which indicates that the fault in mode $\mathcal{M}_3$ is slightly difficult to detect. The FDRs of Situations 2, 3, 7, 8, 12–14 approach 100%. The FARs of Situations 5, 10, 11 and 17–19 are higher than 38%, which signifies that APCA suffers from the catastrophic forgetting issue for successive modes and the model for one mode fails to detect faults in another mode. For Case 6, the FDRs of M-APCA are 100%. The FARs of Situations 6 and 9 are 0.47% and 41.18%, which means that information from modes $\mathcal{M}_1$ and $\mathcal{M}_2$ enhances the monitoring performance for future similar mode $\mathcal{M}_3$. This demonstrates the forward transfer learning ability of M-APCA. Furthermore, the FARs of Situations 1 and 13 are 0.45% and 0.18%, which indicates that the information of future mode $\mathcal{M}_4$ is favorable of improving the performance towards the previous mode $\mathcal{M}_1$. This phenomenon reflects the backward forward learning ability of M-APCA. Similar to Case 5, the FARs of Situations 5, 10, 11 and 17–19 are higher than 30% and are unacceptable. Note that the FARs of Situations 6 and 15 are 0.47% and 0.71%, respectively. This phenomenon may be due to the significant difference between modes $\mathcal{M}_3$ and $\mathcal{M}_4$. The mode $\mathcal{M}_4$ is not able to provide valuable information for monitoring mode $\mathcal{M}_3$. Since a few replay data from

Table 10
Four operating modes of TEP (Case 3 and Case 4)

Mode label	Desired G/H mass ratio	Desired production
$\mathcal{M}_1$	50/50	14076
$\mathcal{M}_2$	10/90	14077
$\mathcal{M}_3$	90/10	11111
$\mathcal{M}_4$	50/50	Maximum

Table 11
Experimental data of the practical coal pulverizing system

Cases	Mode label	Number of training data	Number of testing data	Fault location	Fault cause
Case 5	$\mathcal{M}_1$	2880	720	120	Rotor separator cooling fan trips
	$\mathcal{M}_2$	2880	1080	806	Rotary separator trip
	$\mathcal{M}_3$	2880	2160	134	Large vibration
	$\mathcal{M}_4$	2880	2880	1230	Cooling fan trip of inverter cabinet
Case 6	$\mathcal{M}_1$	2160	1440	1101	Coal block of the coal pipe
	$\mathcal{M}_2$	2520	1440	801	The coal feeder belt is broken
	$\mathcal{M}_3$	1080	1080	846	The coal feeder does not drop coal
	$\mathcal{M}_4$	2160	2160	984	The coal feeder does not drop coal

$\mathcal{M}_3$ are utilized for training the model $\mathcal{F}$, the FARs show a slightly upward trend for Situations 6 and 15. The analysis can be equally applied to the results in Table 6, which also illustrates the effectiveness of M-APCA with Attention II for sequential modes. In summary, the continual learning ability of the proposed M-APCA is illustrated through Situations 1–19, highlighting the forward transfer learning ability and backward transfer learning ability.

The monitoring results of the comparative methods are summarized in Table 7. MNSDiPCA can monitor Case 5 accurately, but fails to detect the faults of Case 6, where the FARs of Situations 22, 23 and 29 are higher than 13%. Similar to M-APCA, the FARs of Situations 23 and 29 are 33.14% and 45.09% respectively. M-APCA and MNSDiPCA used replay continual learning and may encounter the similar issues. Besides, for mode $\mathcal{M}_3$, the testing procedure is affected by manual intervention when the system operates normally, which is also a critical factor causing high FARs. SDiPCA–MSI can provide ordinary performance for Case 5, where the FDRs are higher than 92.5% and the FARs are lower than 7%. However, it fails to monitor Case 6 accurately, where the FARs of Situations 33, 35 and 38 are higher than 17%. Similarly, PCA–EWC fails to detect the faults accurately in Cases 5 and 6. For Case 5, the FARs of Situations 46 and 49 are higher than 10%. For Case 6, the FARs of Situations 43 and 44 are higher than 29%. MCVA is unable to provide outstanding performance for Cases 5 and 6. For Case 5, the FARs of Situations 51, 53, 57 and 58 are higher than 21%. Besides, the FARs of Situations 52, 54, 55 and 57 are higher than 36% for Case 6.

With regard to the online computational complexity, the testing time of SDiPCA–MSI is the lowest in most situations and M-APCA takes second place; PCA–EWC and MNSDiPCA place in the center; MCVA is the highest. According to the analysis mentioned above, the proposed M-APCA method with Attention I or Attention II offers the optimal performance considering accuracy with respect to computing and storage resources.

5.5. Ablation study

In this section, an ablation study is conducted to illustrate the effectiveness of two attention mechanisms. Similar to the experiments designed in Table 5, attention mechanism is not utilized and the method is referred to M-PCA for multimode process monitoring.

The monitoring results of six cases are summarized in Table 12. For Case 1, the FDRs of Situations 6 and 12 are 74.20% and 77.00%. For Case 2, the FDRs of Situations 1–3 are lower than 94%. M-PCA fails to monitor four modes of CSTH based on a model. Using attention mechanism is beneficial to focusing on high-value information and enhancing the performance. For TEP, M-PCA could provide similar monitoring performance with M-APCA, where the FDRs are higher than 95%. For the practical coal pulverizing system, M-PCA can detect the fault of Case 5 accurately. However, M-PCA cannot provide satisfactory performance for Case 6, where the FARs of Situations 3, 6 and 15 are higher than 16%. Through the comparative results in Tables 5, 6 and 12, it can be concluded that attention mechanism is necessary and significant for delivering optimal monitoring performance.

6. Conclusion

This paper has introduced a novel efficient multimodal attentional PCA with continual learning ability for multimode dynamic processes, where an attention mechanism is adopted to focus on the important information from massive data via the dynamic features. Two types of attention models are embedded with a VAR model. To address continual learning with multimode tasks, the idea of replay is used to store previous data selectively. When a new mode arrives, the current mode data and replayed data are jointly used to build a single monitoring model for multiple modes. For either type of attention mechanism, the associated key C^K is updated using maximum likelihood estimation or online k-means cluster algorithm as appropriate. In contrast to traditional multimode methods, data from multiple modes are assumed to be collected sequentially, and only the representative data for each mode are stored for future learning, which reduces consumption of computing and storage resources. Moreover, compared with PCA–EWC and SDiPCA–MSI, M-APCA does not require modes to be similar and can be applied to long-term monitoring tasks. Compared with several state-of-the-art methods, the effectiveness of M-APCA is illustrated through benchmark case studies of a continuous

Table 12
Monitoring results (FDR(%) and FAR (%)) of M-PCA

	Methods	CSTH				TEP				Coal pulverizing system			
		Case 1		Case 2		Case 3		Case 4		Case 5		Case 6	
		FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR	FDR	FAR
Situation 1	PCA	99.20	0	86.00	0	97.11	0	99.87	0	97.00	0	99.71	3.18
Situation 2	M-PCA	94.80	0	93.80	0.20	96.51	0	100	0	100	0.62	100	0.75
Situation 3	M-PCA	100	1.60	93.60	0.40	96.91	0	99.87	0	97.84	0	100	16.64
Situation 6	M-PCA	74.20	0	100	0	98.29	0.25	100	0.25	95.76	0	100	32.66
Situation 7	M-PCA	100	1.40	100	0	96.58	0	99.87	0	97.67	0	99.71	9.55
Situation 8	M-PCA	100	2.20	100	0	95.79	0	100	0	100	5.09	100	0.63
Situation 12	M-PCA	77.00	0	100	0	97.43	0	99.93	0	98.00	0.16	99.34	0
Situation 13	M-PCA	100	1.00	100	0	96.45	0	99.87	0	97.84	0	99.71	1.91
Situation 14	M-PCA	100	1.40	100	0	95.79	0	100	0	100	6.96	100	0.75
Situation 15	M-PCA	100	0.80	100	1.20	98.29	0	100	0	98.37	1.50	100	34.67

stirred tank heater, the Tennessee Eastman process and a practical coal pulverizing system.

In future, the automatic mode identification would be investigated and the graceful forgetting ability would be considered to leave abundant space for future modes. Besides, the interrelationship among different modes would be explored.

Acknowledgements

This work was supported by National Natural Science Foundation of China [grant number 62303114], Natural Science Foundation of Jiangsu Province [grant number BK20230825], the Fundamental Research Funds for the Central Universities, and Zhishan Young Scholar of Southeast University.

References

[1] S. Zhang, C. Zhao, Concurrent analysis of variable correlation and data distribution for monitoring large-scale processes under varying operation conditions, Neurocomputing 349 (2019) 225–238.

[2] L. Ma, J. Dong, K. Peng, Root cause diagnosis of quality-related faults in industrial multimode processes using robust gaussian mixture model and transfer entropy, Neurocomputing 285 (2018) 60–73.

[3] C. Yang, L. Zhou, K. Huang, H. Ji, C. Long, X. Chen, Y. Xie, Multimode process monitoring based on robust dictionary learning with application to aluminium electrolysis process, Neurocomputing 332 (2019) 305–319.

[4] J. Zhang, M. Chen, X. Hong, Nonlinear process monitoring using a mixture of probabilistic pca with clusterings, Neurocomputing 458 (2021) 319–326.

[5] J. Zhang, D. Zhou, M. Chen, X. Hong, Continual learning for multimode dynamic process monitoring with applications to an ultra-supercritical thermal power plant, IEEE Trans. Autom. Sci. Eng. 20 (1) (2023) 137–150.

[6] M. Quiñones-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.

[7] H. Ma, Y. Hu, H. Shi, A novel local neighborhood standardization strategy and its application in fault detection of multimode processes, Chemometr. Intell. Lab. Syst. 118 (2012) 287–300.

[8] Y. Jiang, S. Yin, Recursive total principle component regression based fault detection and its application to vehicular cyber-physical systems, IEEE Trans. Industr. Inform. 14 (4) (2017) 1415–1423.

[9] Q. Wen, Z. Ge, Z. Song, Multimode dynamic process monitoring based on mixture canonical variate analysis model, Ind. Eng. Chem. Res. 54 (5) (2015) 1605–1614.

[10] W. Shao, Z. Ge, L. Yao, Z. Song, Bayesian nonlinear gaussian mixture regression and its application to virtual sensing for multimode industrial processes, IEEE Trans. Autom. Sci. Eng. 17 (2) (2020) 871–885.

[11] L. Yao, W. Shao, Z. Ge, Hierarchical quality monitoring for large-scale industrial plants with big process data, IEEE Trans. Neural Netw. Learn. Syst. 32 (8) (2019) 3330–3341.

[12] K. Huang, Y. Wu, C. Yang, G. Peng, W. Shen, Structure dictionary learning-based multimode process monitoring and its application to aluminum electrolysis process, IEEE Trans. Autom. Sci. Eng. 17 (4) (2020) 1989–2003.

[13] M. Delange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, T. Tuytelaars, A continual learning survey: Defying forgetting in classification tasks, IEEE Trans. Pattern Anal. Mach. Intell 44 (7) (2022) 3366–3385.

[14] N. Y. Masse, G. D. Grant, D. J. Freedman, Alleviating catastrophic forgetting using context-dependent gating and synaptic stabilization, Proc. Nat. Acad. Sci. USA 115 (44) (2018) E10467–E10475.

[15] J. Xu, Z. Zhu, Reinforced continual learning, Adv. Neural Inf. Process. Syst. 31 (2018).

[16] R. Aljundi, P. Chakravarty, T. Tuytelaars, Expert gate: Lifelong learning with a network of experts, in: CVPR, 2017, pp. 7120–7129.

[17] A. Mallya, S. Lazebnik, Packnet: Adding multiple tasks to a single network by iterative pruning, in: CVPR, 2018, pp. 7765–7773.

[18] K. Huang, Z. Tao, Y. Liu, B. Sun, C. Yang, W. Gui, S. Hu, Adaptive multimode process monitoring based on mode-matching and similarity-preserving dictionary learning, IEEE Trans. Cybern. 53 (6) (2023) 3974–3987.

[19] J. Zhang, D. Zhou, M. Chen, Monitoring multimode processes: a modified PCA algorithm with continual learning ability, J. Process Control 103 (2021) 76–86.

[20] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, Overcoming catastrophic forgetting in neural networks, Proc. Nat. Acad. Sci. USA 114 (13) (2017) 3521–3526.

[21] R. Hadsell, D. Rao, A. A. Rusu, R. Pascanu, Embracing change: Continual learning in deep neural networks, Trends Cogn. Sci. 24 (12) (2020) 1028–1040.

[22] J. Zhang, M. Chen, X. Hong, Monitoring multimode nonlinear dynamic processes: an efficient sparse dynamic approach with continual learning ability, IEEE Trans. Industr. Inform. 19 (7) (2023) 8029–8038.

[23] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: Proc. Int. Conf. Learn. Representations, 2015, pp. 1–15.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: Int.

Conf. Neural Inf. Process. Syst., Vol. 30, 2017, pp. 5998–6008.
[25] J. Zhang, J. Xiao, M. Chen, X. Hong, Multimodal continual learning for process monitoring: A novel weighted canonical correlation analysis with attention mechanism, IEEE Transactions on Neural Networks and Learning Systems (2023) 1–15.
[26] Y. Dong, S. J. Qin, A novel dynamic PCA algorithm for dynamic data modeling and process monitoring, J. Process Control 67 (2018) 1–11.
[27] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers, Now Foundations and Trends, 2011.
[28] S. Zhong, T. M. Khoshgoftaar, N. Seliya, Clustering-based network intrusion detection, Int. Jour. Reliab. Qual. Safety Eng. 14 (2007) 169–187.
[29] X. Hong, J. Gao, S. Chen, Zero-attracting recursive least squares algorithms, IEEE Trans. Veh. Technol. 66 (1) (2017) 213–221.
[30] Y. Wang, Q. Miao, E. W. M. Ma, K.-L. Tsui, M. G. Pecht, Online anomaly detection for hard disk drives based on mahalanobis distance, IEEE Trans. Reliab. 62 (1) (2013) 136–145.
[31] J. Zhang, H. Chen, S. Chen, X. Hong, An improved mixture of probabilistic PCA for nonlinear data-driven process monitoring, IEEE Trans. Cybern. 49 (1) (2019) 198–210.
[32] G. Box Gep, G. Box, and G. Jenkins, *Time series analysis: forecasting and control*. John Wiley & Sons, 2011, vol. 734.
[33] X. Ma, Z. Liu, M. Zheng, Y. Wang, Application and exploration of self-attention mechanism in dynamic process monitoring, IFAC-PapersOnLine 55 (6) (2022) 139–144.
[34] D. Cai, "Spectral regression: A regression framework for efficient regularized subspace learning," Ph.D. dissertation, University of Illinois at Urbana-Champaign, 2009.
[35] N. F. Thornhill, S. C. Patwardhan, S. L. Shah, A continuous stirred tank heater simulation model with applications, J. Process Control 18 (3) (2008) 347–360.
[36] B. Song, Y. Ma, H. Shi, Multimode process monitoring using improved dynamic neighborhood preserving embedding, Chemometr. Intell. Lab. Syst. 135 (2014) 17–30.
[37] N.L.Ricker, Optimal steady-state operation of the Tennessee Eastman challenge process, Comput. Chem. Eng. 19 (9) (1995) 949–959.

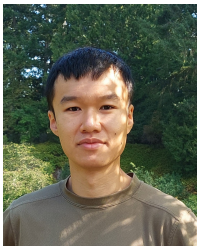
Jingxin Zhang received B.E. degree in School of Electrical Engineering and Automation from Harbin Engineering University, Harbin, China, the M.E. degree in Control Science and Engineering from Harbin Institute of Technology, Harbin, China, in 2014 and 2016, respectively, and the Ph.D. degree in Control Science and Engineering from Tsinghua University, Beijing, China, in 2022.

She is currently a lecture with the Department of Automation, Southeast University. Her research interests are data-driven fault detection and diagnosis, performance monitoring, photovoltaic power forecasting, and their applications in the industrial process.

Haikun Wei received the B.S. degree in industrial automation from the Department of Automation, North China University of Technology, Beijing, China, in 1994, and the M.S. and Ph.D. degrees in control theory and control engineering from the Research Institute of Automation, Southeast University, Nanjing, China, in 1997 and 2000, respectively. From 2005 to 2007, he was a Visiting Scholar with RIKEN Brain Science Institute, Japan.

He is currently a Professor with the School of Automation, Southeast University. His research interest is real and artificial in neural networks and industry automation.

Kanjian Zhang received the B.S. degree in mathematics from Nankai University, Tianjin, China, in 1994, and the M.S. and Ph.D. degrees in control theory and control engineering from Southeast University, Nanjing, China, in 1997 and 2000, respectively. He is currently a Professor with the School of Automation, Southeast University. His research interests include nonlinear control theory and its applications, with particular interest in robust output feedback design and optimization control.



James Xiao received his MSci degree in Chemistry (2014) from the University of Bristol, UK, followed by his MRes in Nanoscience and Nanotechnology (2015) and PhD in Physics (2019) from the University of Cambridge, UK. From 2019 to 2022 he worked as a postdoctoral research associate at the Cavendish Laboratory, University of Cambridge. His current interests include machine learning and computer vision.

Xia Hong received the B.Sc. and M.Sc. degrees from the National University of Defense Technology, China, in 1984 and 1987, respectively, and the Ph.D. degree from The University of Sheffield, U.K., in 1998, all in automatic control. She was a Research Assistant with the Beijing Institute of Systems Engineering, Beijing, China, from 1987 to 1993. She was a Research Fellow with the Department of Electronics and Computer Science, University of Southampton, from 1997 to 2001.

She is currently a Professor with the Department of Computer Science, School of Mathematical, Physical and Computational Sciences, University of Reading. She is actively involved in research into nonlinear systems identification, data modeling, estimation and intelligent control, neural networks, pattern recognition, learning theory, and their applications. She has authored over 170 research papers, and co-authored a research book. Dr. Hong received the Donald Julius Groen Prize from IMechE in 1999.