Principal Predictor Analysis With Application to Dynamic Process Monitoring

Shumei Chen and S. Joe Qin, *Fellow, IEEE*

***Abstract*—Modern engineering and scientific systems are usually equipped with abundant sensors to collect large-dimensional time series for monitoring and operations. In this article, we develop a novel principal predictor analysis (PPA) framework with reduced-dimensional dynamics to obtain parsimonious predictor models of large-dimensional time series data. Principal predictors are obtained by maximizing the variance of predictions from their past values. Unlike classical principal component analysis (PCA), which reduces the dimensionality without emphasizing the prediction, PPA focuses on extracting latent variables with the maximum predictive capability. The PPA application to dynamic process monitoring is performed with predictive monitoring indices to account for variations in the predictors and the unpredicted residuals, which can be subsequently modeled with PCA. PPA-based monitoring and diagnosis are demonstrated in an illustrative closed-loop system and the industrial Dow Challenge Problem and an extension to include known first-principles relations to show their effectiveness.**

***Index Terms*—Data reconstruction, dimensionality reduction, fault diagnosis, principal component analysis (PCA), principal predictor analysis (PPA), sustainable process operations.**

I. Introduction

PRINCIPAL component analysis (PCA) has been one of the most popular analysis methods for more than a century [1], [2], [3]. A plethora of PCA extensions, such as probabilistic, kernel, neural networks, and autoencoders, are still in active development [4], [5], [6], [7], [8], [9], [10], [11], which are suitable for anomaly detection and process monitoring [12], [13], [14], [15], [16], [17]. These PCA models perform low-dimensional mappings with maximum variance, but do not focus on extracting the predictable content of the data, as illustrated in the literature and in the Lorenz attractor example in [18].

Recent deployment of the Industrial Internet of Things (IIoT) for intelligent and autonomous systems yields abundant large time series data. Dimensionality reduction is often necessary for the use of massive data for the monitoring, prediction, and decision-making of the system. On the other hand, many industrial manufacturing processes already have large-dimensional operation data that can be used for process data analytics. These data series tend to have both cross-correlation among variables and serial correlation over time, which is referred to as co-moving dynamics or *co-dynamics* in this article. In the recent decade, the problem of data co-dynamics has been treated systematically as latent dynamic modeling tasks [19], [20], [21], [22], [23], [24], [25]. Relevant work is found in econometrics and financial data analysis as dynamic factor models (DFM) [26], [27], [28] and in machine learning as slow and predictable factor analysis [29], [30]. A recent review of these methods and their relationships is available in [31].

Using univariate latent dynamics, dynamic inner PCA (DiPCA) and dynamic inner canonical correlation analysis (DiCCA) have been developed with latent univariate autoregressive (AR) models [21], [22], [32]. Univariate latent models promote self-predicting dynamic latent variables (DLV) in the data, which are convenient for the extraction of oscillatory factors and troubleshooting [33]. To overcome the limited representability of these univariate AR models, a latent vector AR (LaVAR) model is developed in [18], [34], [35] with a canonical correlation analysis (CCA) objective and oblique projections to capture the predictable content in a multivariate time series. The LaVAR-CCA algorithm solves all DLVs simultaneously with an iterative algorithm. Nonlinear extensions with kernel functions are developed in [36]. The LaVAR-CCA DLVs are extracted in descending order of predictability, and each DLV requires the past values of all other DLVs to make a prediction. Since these methods with CCA objectives to extract DLVs focus on the angles between prediction $\hat{\boldsymbol{v}}_k$ and projected values $\boldsymbol{v}_k$, they ignore the magnitude of the variance of the DLVs.

Based on the aforementioned work, a new principal predictor analysis (PPA) is proposed in this article to compress large-dimensional data to a number of DLVs with the most predictable variances from their past values. The compression loading matrix is made orthonormal to retain the variance information in the DLVs, whereas the prediction models of the DLVs are implemented with vector AR (VAR) models for simplicity. It is possible to implement other forms of latent dynamic models, such as state-space models [37], [38].

The PPA model aims to achieve the maximum predicted variance of the principal predictors. After the predictable variations are extracted with a number of DLVs, the prediction residuals are essentially independent over time, which can

Received 7 July 2025; revised 11 September 2025; accepted 12 September 2025. This work was supported in part by grants from the General Research Fund of the Research Grants Council (RGC) of Hong Kong SAR, China, under Project 11303421 and Project 13300525, and in part by the Research Impact Fund by RGC of Hong Kong under Project 130272. This article was recommended by Associate Editor G. Zhu. *(Corresponding author: S. Joe Qin.)*

The authors are with the School of Data Science, Lingnan University, Hong Kong (e-mail: joeqin@ln.edu.hk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2025.3610475.

Digital Object Identifier 10.1109/TCYB.2025.3610475

be further analyzed with PCA. As a consequence, the PPA model followed by PCA is used to develop a global index for dynamic process monitoring that accounts for variations in the prediction residuals and the principal predictors.

Denoting $\boldsymbol{y}_k \in \Re^p$ as the measurement vector of p variables in a time series, the PPA model is to find $\ell < p$ DLVs $\boldsymbol{v}_k \in \Re^\ell$ to capture the most predictable variations in $\boldsymbol{y}_k$ using the past $\boldsymbol{v}_k$. A unique advantage of PPA is that the predictor model for $\boldsymbol{v}_k$ is self-dependent and parsimonious. In addition, the excessive dimensions orthogonal to the co-dynamic factors have little serial dependence and are useful for fault monitoring.

The remainder of this article is organized as follows. The next section defines the PPA of latent dynamic data with orthonormal loadings compared to traditional full-dimensional predictor models. In Section III the PPA algorithm is developed with a maximum likelihood (ML) formulation which has an interpretation to maximize the predicted variances of the predictors. In Section IV, an overall predictive monitoring index is developed that accounts for the prediction residuals and principal predictors. PPA integrated with first-principles relations and reconstruction-based contributions (RBC) are developed for PPA based fault diagnosis. The developed PPA analysis and monitoring algorithms are illustrated in Section V in a simple closed-loop control system and the Dow Challenge process data set [39] to show the effectiveness of the proposed method compared to other state-of-the-art methods. The last section gives conclusions.

II. Dynamic Latent Variable Models

A. Full-dimensional Dynamic Series Modeling

Let $\boldsymbol{y}_k \in \Re^p$ be the measurement vector of variables in time k of a dynamic system, which forms a time series $\{\boldsymbol{y}_k\}_{k=1}^{N+s}$ with $\mathbb{E}\{\boldsymbol{y}_k\} = \boldsymbol{0}$. The series is considered to be serially correlated if $\mathbb{E}\{\boldsymbol{y}_k \boldsymbol{y}_{k-j}^\intercal\} \neq \boldsymbol{0}$ for some $j > 0$. In this article, we consider time series of reduced-dimensional dynamicss (RDDss), which was defined in [18] and is given as follows.

Definition 1: A serially correlated time series $\{\boldsymbol{y}_k\}$ is referred to as a RDDs series if there exists $\mathbf{a} \neq \boldsymbol{0} \in \Re^p$ such that $\mathbf{a}^\intercal \boldsymbol{y}_k$ is serially uncorrelated. Otherwise, it is termed as a full-dimensional dynamic (FDD) series [18].

Traditional FDD series modeling has been studied intensively, e.g., [40]. In general, a FDD series can be represented by its best prediction $\hat{\boldsymbol{y}}_k$ and the unpredictable innovation $\boldsymbol{e}_k$ as follows:

$$\boldsymbol{y}_k = \hat{\boldsymbol{y}}_k + \boldsymbol{e}_k \tag{1}$$

where $\hat{\boldsymbol{y}}_k \in \Re^p$ depends on the past data $\{\boldsymbol{y}_k\}_{-\infty}^{k-1}$ but $\boldsymbol{e}_k$ does not. The predictor $\hat{\boldsymbol{y}}_k$ can be implemented with a suitable model, such as state space or an AR integrated moving average (ARIMA). For a VAR model, which is popular in many applications, the prediction $\hat{\boldsymbol{y}}_k \in \Re^p$ depends on finite past data $\{\boldsymbol{y}_{k-j}\}_{j=1}^s$, which is

$$\hat{\boldsymbol{y}}_k = \sum_{j=1}^{s} \mathbf{A}_j \boldsymbol{y}_{k-j} \tag{2}$$

where $\mathbf{A}_j$'s can be found using the ML or least squares methods.

B. Principal Predictors

A full-dimensional predictor $\hat{\boldsymbol{y}}_k$ always has the same dimension as the observation vector $\boldsymbol{y}_k$. A principal dynamic predictor $\hat{\boldsymbol{v}}_k$ has a smaller dimension than the measurement $\boldsymbol{y}_k$ and can be represented as

$$\boldsymbol{y}_k = \mathbf{P}\hat{\boldsymbol{v}}_k + \boldsymbol{e}_k \tag{3}$$

where $\hat{\boldsymbol{v}}_k \in \Re^\ell$, $l < p$, is the *principal predictor* estimated from past data, i.e., $\hat{\boldsymbol{v}}_k = f(\boldsymbol{y}_{k-1}, \ldots, \boldsymbol{y}_{k-s}, \ldots,)$, $\mathbf{P} \in \Re^{p\times\ell}$ has full column rank, and $\boldsymbol{e}_k \in \Re^p$ is the residual whose covariance is to be minimized. Premultiplying the Moore–Penrose pseudo-inverse $\mathbf{P}^+$ on (3) gives the DLV

$$\boldsymbol{v}_k = \mathbf{P}^+ \boldsymbol{y}_k = \hat{\boldsymbol{v}}_k + \mathbf{P}^+ \boldsymbol{e}_k = \hat{\boldsymbol{v}}_k + \boldsymbol{\varepsilon}_k \tag{4}$$

where $\boldsymbol{\varepsilon}_k = \mathbf{P}^+ \boldsymbol{e}_k$. It is clear that $\hat{\boldsymbol{v}}_k$ is a predictor of $\boldsymbol{v}_k = \mathbf{P}^+ \boldsymbol{y}_k$, which is in the principal predictor subspace (PPS).

Remark 1: A PCA model is given as follows:

$$\boldsymbol{y}_k = \mathbf{P}\boldsymbol{t}_k + \boldsymbol{e}_k$$

where $\boldsymbol{t}_k = \mathbf{P}^\intercal \boldsymbol{y}_k$ has the maximum variances with an orthonormal $\mathbf{P}$, which differs from the PPA loadings in (3).

The PPA model (3) involves two simultaneous tasks, that is, the estimation of the loading matrix $\mathbf{P}$ and that of the predictor $\hat{\boldsymbol{v}}_k$ from past DLV values.

In practice, the dimension ℓ is unknown, so we need to find its estimate $\hat{\ell}$. If $\hat{\ell} > \ell$, $\boldsymbol{v}_k$ could include all dynamics in $\boldsymbol{y}_k$ but this leads to over-parameterization, while $\hat{\ell} < \ell$ leads to under-parameterization. When $\hat{\ell} = \ell$ the dynamics $\boldsymbol{y}_k$ is properly included in $\boldsymbol{v}_k$ and the innovation $\boldsymbol{e}_k \in \Re^p$ is not correlated with the previous data, that is, $\mathbb{E}\{\boldsymbol{e}_k \boldsymbol{y}_{k-j}^\intercal\} = \boldsymbol{0}$ for all $j > 0$.

The principal predictor model for $\hat{\boldsymbol{v}}_k$ can be any linear, nonlinear, or a recurrent neural net function of its past, i.e.,

$$\hat{\boldsymbol{v}}_k = \mathbb{E}\{\boldsymbol{v}_k | \{\boldsymbol{v}_{k-j}\}_{j=1}^\infty\} = g(\boldsymbol{v}_{k-1}, \ldots, \boldsymbol{v}_{k-s}, \ldots,). \tag{5}$$

To parameterize this predictor model, the works in [21], [32] used univariate AR models. [18] used a latent VAR model, while [37] used a state space model and [38] used a state space model with a deterministic control sequence $\{\boldsymbol{u}_k\}$ for latent system identification.

Our work is different from many existing works in econometrics and statistics, such as [28], that used the original data $\boldsymbol{y}_k$ to predict the DLVs, i.e.,

$$\hat{\boldsymbol{v}}_k = \mathbb{E}\{\boldsymbol{v}_k | \{\boldsymbol{y}_{k-j}\}_{j=1}^\infty\} = f\big(\boldsymbol{y}_{k-1}, \ldots, \boldsymbol{y}_{k-s}, \ldots,\big)$$

which loses the reduced dimensional parameterization, and thus is nonparsimonious. In this article, we specifically consider a latent VAR predictor as

$$\hat{\boldsymbol{v}}_k = \sum_{j=1}^{s} \mathbf{B}_j \boldsymbol{v}_{k-j}. \tag{6}$$

However, other models, such as the state space Kalman predictor [37] can be used.

C. Orthonormal Loadings for Principal Predictors

Since the DLV innovation $\boldsymbol{\varepsilon}_k$ in (4) is part of the $\boldsymbol{y}_k$ innovations, we define the following transformation:

$$\boldsymbol{e}_k = \mathbf{P}\boldsymbol{\varepsilon}_k + \bar{\mathbf{P}}\bar{\boldsymbol{\varepsilon}}_k = \begin{bmatrix}\mathbf{P} & \bar{\mathbf{P}}\end{bmatrix}\begin{bmatrix}\boldsymbol{\varepsilon}_k \\ \bar{\boldsymbol{\varepsilon}}_k\end{bmatrix} \tag{7}$$

where $\bar{\mathbf{P}}$ has full column rank and $[\mathbf{P} \quad \bar{\mathbf{P}}]$ is invertible. This transformation always exists since $[\mathbf{P} \quad \bar{\mathbf{P}}]$ is invertible, which makes $\boldsymbol{\varepsilon}_k$ and $\bar{\boldsymbol{\varepsilon}}_k$ uncorrelated in time. With the above relation, the predictor form (3) has the following form:

$$\boldsymbol{y}_k = \mathbf{P}\hat{\boldsymbol{v}}_k + \mathbf{P}\boldsymbol{\varepsilon}_k + \bar{\mathbf{P}}\bar{\boldsymbol{\varepsilon}}_k = \mathbf{P}\boldsymbol{v}_k + \bar{\mathbf{P}}\bar{\boldsymbol{\varepsilon}}_k \tag{8}$$

which is referred to as the *data generation process*.

It is clear that the loadings $\mathbf{P}$ cannot be uniquely identified due to the bilinear form with $\boldsymbol{v}_k$. We give the following theorem to represent all possible realizations of data $\boldsymbol{y}_k$ generated by (8) and a specific orthonormal realization.

Theorem 1: For any nonsingular $\mathbf{M}_1 \in \Re^{\ell\times\ell}$, $\mathbf{M}_2 \in \Re^{(p-\ell)\times(p-\ell)}$, and any $\mathbf{N} \in \Re^{\ell\times(p-\ell)}$, data $\boldsymbol{y}_k$ generated by (8) can be equivalently realized by

$$\boldsymbol{y}_k = \mathbf{P}'\boldsymbol{v}'_k + \bar{\mathbf{P}}'\bar{\boldsymbol{\varepsilon}}'_k \tag{9}$$

where

$$\mathbf{P}' = \mathbf{P}\mathbf{M}_1^{-1}, \quad \boldsymbol{v}'_k = \mathbf{M}_1(\boldsymbol{v}_k + \mathbf{N}\bar{\boldsymbol{\varepsilon}}_k)$$
$$\bar{\mathbf{P}}' = (\bar{\mathbf{P}} - \mathbf{P}\mathbf{N})\mathbf{M}_2^{-1}, \quad \bar{\boldsymbol{\varepsilon}}'_k = \mathbf{M}_2\bar{\boldsymbol{\varepsilon}}_k.$$

Furthermore, we have the following orthonormal realization:

$$\begin{bmatrix}\mathbf{P}' & \bar{\mathbf{P}}'\end{bmatrix}^\mathsf{T}\begin{bmatrix}\mathbf{P}' & \bar{\mathbf{P}}'\end{bmatrix} = \mathbf{I} \tag{10}$$

if $\mathbf{N} = (\mathbf{P}^\mathsf{T}\mathbf{P})^{-1}\mathbf{P}^\mathsf{T}\bar{\mathbf{P}}$, $\mathbf{M}_1 = (\mathbf{P}^\mathsf{T}\mathbf{P})^{0.5}$ and $\mathbf{M}_2 = ((\bar{\mathbf{P}} - \mathbf{P}\mathbf{N})^\mathsf{T}(\bar{\mathbf{P}} - \mathbf{P}\mathbf{N}))^{0.5}$.

The proof of this theorem is straightforward algebra. It is clear that $\mathbf{P}'$ and $\mathbf{P}$ represent the same column space. Therefore, without loss of generality, we drop the prime in the notation in the rest of this article to implement an orthonormal $\mathbf{P}$, i.e. $\mathbf{P}^\mathsf{T}\mathbf{P} = \mathbf{I}$. As a consequence, the magnitude of the predictable content is contained in $\hat{\boldsymbol{v}}_k$ of (3). In the next section, we show with Theorem 2 that the optimal orthonormal $\mathbf{P}$ permits a diagonal covariance of $\hat{\boldsymbol{v}}_k$.

With (3) and (6), we obtain the following reduced-dimensional VAR (ReDVAR) model [34]:

$$\boldsymbol{y}_k = \mathbf{P}\sum_{j=1}^{s}\mathbf{B}_j\boldsymbol{v}_{k-j} + \boldsymbol{e}_k = \sum_{j=1}^{s}\mathbf{P}\mathbf{B}_j\mathbf{P}^\mathsf{T}\boldsymbol{y}_{k-j} + \boldsymbol{e}_k \tag{11}$$

with $\boldsymbol{v}_k = \mathbf{P}^\mathsf{T}\boldsymbol{y}_k$ and the VAR parameter matrices $\{\mathbf{A}_j = \mathbf{P}\mathbf{B}_j\mathbf{P}^\mathsf{T}\}$ having reduced rank compared to (2). Therefore, PPA is equivalent to the ReDVAR model (11).

III. Principal Predictor Analysis

A. PPA Objective and Optimal Solution

The PPA modeling aims to find orthonormal loadings $\mathbf{P} \in \Re^{p\times\ell}$ to make the latent vector $\boldsymbol{v}_k \in \Re^\ell$ most predictable. From (4) and (8), it is seen that the PPA model projects $\boldsymbol{y}_k$ to a lower dimension $\boldsymbol{v}_k$ and then makes $\boldsymbol{v}_k$ the most predictable from its past values. We assume $\boldsymbol{e}_k \sim \mathcal{N}(\mathbf{0}, \Sigma_{\boldsymbol{e}})$, which is more general than the probabilistic PCA formulation [41] which assumes an identity covariance for $\boldsymbol{e}_k$.

The ML solution amounts to minimizing the following likelihood function [35]:

$$L^{\boldsymbol{y}} = N\ln|\Sigma_{\boldsymbol{e}}| + \sum_{k=s+1}^{s+N}(\boldsymbol{y}_k - \mathbf{P}\hat{\boldsymbol{v}}_k)^\mathsf{T}\Sigma_{\boldsymbol{e}}^{-1}(\boldsymbol{y}_k - \mathbf{P}\hat{\boldsymbol{v}}_k) \tag{12}$$

$$= N\ln|\Sigma_{\boldsymbol{e}}| + \sum_{k=s+1}^{s+N}\left(\boldsymbol{y}_k - \sum_{j=1}^{s}\mathbf{P}\mathbf{B}_j\boldsymbol{v}_{k-j}\right)^\mathsf{T}\Sigma_{\boldsymbol{e}}^{-1}\left(\boldsymbol{y}_k - \sum_{j=1}^{s}\mathbf{P}\mathbf{B}_j\boldsymbol{v}_{k-j}\right). \tag{13}$$

Differentiating (13) with respect to $(\mathbf{P}\mathbf{B}_j)$ and setting it to zero lead to

$$\sum_{k=s+1}^{s+N}\left(\boldsymbol{y}_k - \sum_{i=1}^{s}\mathbf{P}\mathbf{B}_i\boldsymbol{v}_{k-i}\right)\boldsymbol{v}_{k-j}^\mathsf{T} = \mathbf{0}, \quad j = 1,\ldots,s. \tag{14}$$

Premultiplying $\mathbf{P}^\mathsf{T}$ on (14) and using $\mathbf{P}^\mathsf{T}\mathbf{P} = \mathbf{I}$ and $\boldsymbol{v}_k = \mathbf{P}^\mathsf{T}\boldsymbol{y}_k$, we obtain

$$\sum_{k=s+1}^{s+N}\left(\boldsymbol{v}_k - \sum_{i=1}^{s}\mathbf{B}_i\boldsymbol{v}_{k-i}\right)\boldsymbol{v}_{k-j}^\mathsf{T} = \mathbf{0}, \quad j = 1,\ldots,s. \tag{15}$$

Differentiating (12) with respect to $\mathbf{P}$ by including $\mathbf{P}^\mathsf{T}\mathbf{P} = \mathbf{I}$ with Lagrange multipliers and setting it to zero lead to

$$\sum_{k=s+1}^{s+N}(\boldsymbol{y}_k - \mathbf{P}\hat{\boldsymbol{v}}_k)\hat{\boldsymbol{v}}_k^\mathsf{T} = \mathbf{0}. \tag{16}$$

Further, differentiating (12) with respect to $\Sigma_{\boldsymbol{e}}^{-1}$ and setting it to zero lead to

$$\Sigma_{\boldsymbol{e}} = \frac{1}{N}\sum_{k=s+1}^{s+N}(\boldsymbol{y}_k - \mathbf{P}\hat{\boldsymbol{v}}_k)(\boldsymbol{y}_k - \mathbf{P}\hat{\boldsymbol{v}}_k)^\mathsf{T}. \tag{17}$$

Denoting

$$\mathbb{V} = \begin{bmatrix}\mathbf{V}_{s-1} & \mathbf{V}_{s-2} & \cdots & \mathbf{V}_0\end{bmatrix} \tag{18}$$

$$\mathbb{B} = \begin{bmatrix}\mathbf{B}_1 & \mathbf{B}_2 & \cdots & \mathbf{B}_s\end{bmatrix}^\mathsf{T} \tag{19}$$

$$\mathbf{V}_i = \begin{bmatrix}\boldsymbol{v}_{i+1} & \boldsymbol{v}_{i+2} & \cdots & \boldsymbol{v}_{i+N}\end{bmatrix}^\mathsf{T} = \mathbf{Y}_i\mathbf{P} \tag{20}$$

$$\mathbf{Y}_i = \begin{bmatrix}\boldsymbol{y}_{i+1} & \boldsymbol{y}_{i+2} & \cdots & \boldsymbol{y}_{i+N}\end{bmatrix}^\mathsf{T}, \qquad i = 0, 1, \ldots, s$$

$$\hat{\mathbf{V}}_s = \begin{bmatrix}\hat{\boldsymbol{v}}_{s+1} & \hat{\boldsymbol{v}}_{s+2} & \cdots & \hat{\boldsymbol{v}}_{s+N}\end{bmatrix}^\mathsf{T} \tag{21}$$

where $\mathbf{Y}_i$ and $\mathbf{V}_i$ are submatrices of

$$\mathbf{Y} = \begin{bmatrix}\boldsymbol{y}_1 & \boldsymbol{y}_2 & \cdots & \boldsymbol{y}_N & \cdots & \boldsymbol{y}_{s+N}\end{bmatrix}^\mathsf{T}$$
$$\mathbf{V} = \begin{bmatrix}\boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_N & \cdots & \boldsymbol{v}_{s+N}\end{bmatrix}^\mathsf{T}$$

respectively. The solution for $\mathbb{B}$ given $\mathbf{P}$ can be found from (15) as

$$\mathbb{B} = \left(\mathbb{V}^\mathsf{T}\mathbb{V}\right)^{-1}\mathbb{V}^\mathsf{T}\mathbf{V}_s = \mathbb{V}^+\mathbf{V}_s \tag{22}$$

where $\mathbb{V}^+$ is the Moore–Penrose pseudo-inverse. It should be noted that (22) coincides with the least squares solution for $\mathbb{B}$, which does not require the assumption of the probability distribution for $\boldsymbol{e}_k$ in (3). The prediction of $\mathbf{V}_s$ from (6) is

$$\hat{\mathbf{V}}_s = \begin{bmatrix}\hat{\boldsymbol{v}}_{s+1} & \cdots & \hat{\boldsymbol{v}}_{s+N}\end{bmatrix}^\mathsf{T} = \mathbb{V}\mathbb{B} = \mathbb{V}\mathbb{V}^+\mathbf{V}_s \tag{23}$$

which yields

$$\hat{\mathbf{V}}_s^\mathsf{T}\hat{\mathbf{V}}_s = \mathbf{V}_s^\mathsf{T}\mathbb{V}\mathbb{V}^+\mathbb{V}\mathbb{V}^+\mathbf{V}_s = \mathbf{V}_s^\mathsf{T}\mathbb{V}\mathbb{V}^+\mathbf{V}_s = \mathbf{V}_s^\mathsf{T}\hat{\mathbf{V}}_s \tag{24}$$

Algorithm 1 PPA Algorithm

1: Given data $\{\boldsymbol{y}_k\}_{k=1}^{N+s}$, form $\mathbf{Y}_s \in \Re^{N\times p}$ and scale $\mathbf{Y}_s$ to zero mean and unit variance. Form $\mathbf{Y} \in \Re^{(N+s)\times p}$ and scale $\mathbf{Y}$ according to the scaling of $\mathbf{Y}_s$.
2: Perform SVD on $\mathbf{Y}_s$ as

$$\mathbf{Y}_s = \mathcal{U}_s\mathcal{D}_s\mathcal{V}_s^\intercal$$

and initialize $\hat{\mathbf{P}} = \mathcal{V}_s(:, 1{:}\ell)$ for a selected ℓ.
3: **repeat**
Calculate $\mathbf{V} = \mathbf{Y}\hat{\mathbf{P}}$ and calculating $\hat{\mathbf{V}}_s$ by forming $\mathbb{V} = \begin{bmatrix}\mathbf{V}_{s-1}\ \mathbf{V}_{s-2}\ \cdots\ \mathbf{V}_0\end{bmatrix}$ from $\mathbf{V}$.
Perform EVD (25) and calculate $\hat{\mathbf{P}} = \mathbf{W}(:, 1{:}\ell)$.
4: **until** convergence.
5: Choose $\hat{\bar{\mathbf{P}}} = \mathbf{W}(:, \ell+1{:}p)$.

because $\mathbb{V}\mathbb{V}^+\mathbb{V} = \mathbb{V}$.

Note that the solution for $\mathbb{B}$ depends on knowing $\mathbf{P}$, while the solution for $\mathbf{P}$ from (16) depends on knowing $\mathbb{B}$. Therefore, we adopt an alternate optimization procedure following [18]. Further, if $\mathbf{P}$ is a solution, $\mathbf{P}\mathcal{O}$ would also be a solution for any orthogonal matrix $\mathcal{O}$. Therefore, we have an opportunity to rotate $\mathbf{P}$ such that $\hat{\boldsymbol{v}}_k$ has maximized variances with its variances arranged in a nonincreasing order.

B. PPA Solution and Algorithm

We give the following theorem to solve for $\mathbf{P}$ to make the entries of $\hat{\boldsymbol{v}}_k$ have maximized variances arranged in a nonincreasing order.

Theorem 2: With an initial $\mathbf{P} \in \Re^{p\times\ell}$ to calculate $\hat{\mathbf{V}}_s$ from (22), perform an eigen-vector decomposition (EVD) on

$$\mathbf{Y}_s^\intercal\hat{\mathbf{V}}_s\hat{\mathbf{V}}_s^+\mathbf{Y}_s/N = \mathbf{W}\Lambda\mathbf{W}^\intercal \tag{25}$$

where Λ contains the eigenvalues in a nonincreasing order. The solution

$$\hat{\mathbf{P}} = \mathbf{W}(:, 1{:}\ell) \tag{26}$$

makes the entries of $\hat{\boldsymbol{v}}_k$ have maximized variances arranged in a nonincreasing order. Further

$$\hat{\Sigma}_{\hat{\boldsymbol{v}}} = \Lambda(1{:}\ell, 1{:}\ell) = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_\ell) \tag{27}$$

contains the variances of the ℓ DLVs with in a nonincreasing order.

The proof of the theorem is given in Appendix A. To estimate the data generation model (8), we can choose $\bar{\mathbf{P}} = \mathbf{W}(:, \ell{+}1{:}p)$. To initialize the algorithm, we choose $\hat{\mathbf{P}}$ to be the first ℓ principal component (PC) loadings of $\mathbf{Y}_s$. The complete PPA algorithm is summarized in Algorithm 1.

Remark 2: Another simple approach to initializing Algorithm 1 is to first build univariate AR models for each variable in $\boldsymbol{y}_k$. Then rank-order the variables according to their univariate predictability in descending order and choose the ℓ most predictable variables in $\boldsymbol{y}_k$ as the initial $\boldsymbol{v}_k$.

C. Analysis and Extension With Known Relations

Denoting $\Pi_{\hat{\mathbf{V}}} = \hat{\mathbf{V}}_s\hat{\mathbf{V}}_s^+$ as an orthogonal projection matrix, (25) is equivalent to the EVD of $\mathbf{Y}_s^\intercal\Pi_{\hat{\mathbf{V}}}\mathbf{Y}_s/N$, which is the projection of $\mathbf{Y}_s$ onto the column space of $\hat{\mathbf{V}}_s$. Recall that PCA performs EVD on $\mathbf{Y}_s^\intercal\mathbf{Y}_s/N$. Therefore, the projection of $\mathbf{Y}_s$ by $\Pi_{\hat{\mathbf{V}}}$ is a key difference between PPA and PC analysis (PCA).

Next, we compare PPA with the dynamic factor model algorithm in Lam et al. [26], which performs EVD on

$$\mathbf{L} = \mathbf{Y}_s^\intercal\left(\sum_{j=1}^{s}\mathbf{Y}_{s-j}\mathbf{Y}_{s-j}^\intercal\right)\mathbf{Y}_s/N^2 = \mathbf{Y}_s^\intercal\mathbb{Y}\mathbb{Y}^\intercal\mathbf{Y}_s/N^2$$

and assign its leading ℓ eigenvectors as $\mathbf{P}$, where $\mathbb{Y} = \begin{bmatrix}\mathbf{Y}_{s-1}\ \mathbf{Y}_{s-2}\ \cdots\ \mathbf{Y}_0\end{bmatrix}$. Note that the weight matrix $\mathbb{Y}\mathbb{Y}^\intercal$ in the middle is not a projection matrix. Therefore, [26] performs merely SVD on a covariance matrix $\mathbf{Y}_s^\intercal\mathbb{Y}/N$, which does not give an explicit latent predictor.

The convergence of Algorithm 1 can be secured using the expectation-maximization (EM) interpretation [42]. The log-likelihood function in (13) makes Algorithm 1 an EM solution of the PPA model (3), where $\boldsymbol{y}_k$ are the observed variables, $\hat{\boldsymbol{v}}_k$ the latent variables, and $(\{\mathbf{B}_j\}, \mathbf{P})$ the model parameters. The existing EM convergence results in [42] guarantee that the log-likelihood function (13) increases monotonically during iterations and converges to a local solution. However, a global optimum is not guaranteed unless (13) is unimodal.

In many engineering applications, it is often desirable to retain first-principles relations or factors while exploring other data-driven latent variables.

Remark 3: Given $\mathbf{C}^\intercal\boldsymbol{y}_k = \mathbf{0}$ as a known first-principles relation or $\mathbf{C}^\intercal\boldsymbol{y}_k$ is a known factor to be enforced, they can be integrated in the PPA algorithm by transforming the data matrix $\mathbf{Y}$ with

$$\tilde{\mathbf{Y}} = \mathbf{Y}\left(\mathbf{I} - \mathbf{C}\mathbf{C}^+\right) \tag{28}$$

and treat $\tilde{\mathbf{Y}}$ as the data matrix to be computed via Algorithm 1 to obtain the data-driven latent variables and predictors. The case of $\mathbf{C}^\intercal\boldsymbol{y}_k = \mathbf{0}$ is regarded as known static factors that represent collinearity, while a known prior factor $\mathbf{C}^\intercal\boldsymbol{y}_k$ represents a factor of physical significance.

D. Determining the Number of DLVs

The number of DLVs ℓ can be chosen to capture the most predictable variations in the data by the DLVs so that there are virtually no predictable variations left in the residuals. For monitoring purposes, the work in [43] proposed using the cumulative proportion of predicted variance (CPPV) to select the number of DLVs for univariate DLV models. For PPA models, the DLV relations are multivariate. Given $\hat{\boldsymbol{v}}_k$ and $\hat{\mathbf{P}}$, the corresponding prediction is $\hat{\boldsymbol{y}}_k = \hat{\mathbf{P}}\hat{\boldsymbol{v}}_k$. The predicted variance with ℓ DLVs is

$$\mathtt{trace}\left(\hat{\Sigma}_{\hat{\boldsymbol{y}}}(\ell)\right) = \mathtt{trace}\left(\hat{\mathbf{P}}\hat{\Sigma}_{\hat{\boldsymbol{v}}}\hat{\mathbf{P}}^\intercal\right) = \sum_{i=1}^{\ell}\lambda_i$$

which gives the total predictable variance when $\ell = p$. We define the proportion of total variance (PTV) of ℓ DLVs as

$$\mathrm{PTV}(\ell) = \frac{\mathtt{trace}\left(\hat{\Sigma}_{\hat{y}}(\ell)\right)}{\mathtt{trace}\left(\hat{\Sigma}_{y}\right)} = \frac{\sum_{i=1}^{\ell}\lambda_i}{\mathtt{trace}\left(\hat{\Sigma}_{y}\right)}. \tag{29}$$

Further, the proportion of predictable variance (PPV) of ℓ DLVs is defined as

$$\mathrm{PPV}(\ell) = \frac{\mathtt{trace}\left(\hat{\Sigma}_{\hat{y}}(\ell)\right)}{\mathtt{trace}\left(\hat{\Sigma}_{\hat{y}}(p)\right)} = \frac{\sum_{i=1}^{\ell}\lambda_i}{\sum_{i=1}^{p}\lambda_i}. \tag{30}$$

To determine the number of DLVs, one simple method is to choose ℓ so that $\mathrm{PPV}(\ell)$ accounts for, say, 95% of the predicted variance using all p latent variables. A final model with ℓ DLVs is built, which can be used for process monitoring purposes.

IV. PPA for Monitoring

A. Impact of Faults on Prediction Residuals

The DLV prediction $\hat{\boldsymbol{v}}_k$ depends on the past $\boldsymbol{v}_k$ as given in (6), which further depends on the past $\boldsymbol{y}_k$. The prediction error according to (11) is

$$\boldsymbol{e}_k = \boldsymbol{y}_k - \sum_{j=1}^{s}\mathbf{P}\mathbf{B}_j\mathbf{P}^{\mathsf{T}}\boldsymbol{y}_{k-j}.$$

If a fault affects the data as

$$\boldsymbol{y}_k = \boldsymbol{y}_k^* + \boldsymbol{f}_k$$

where $\boldsymbol{f}_k$ is the actual fault and $\boldsymbol{y}_k^*$ is the fault-free portion of $\boldsymbol{y}_k$. The impact of the fault on the residual is

$$\begin{aligned}\boldsymbol{e}_k &= \boldsymbol{y}_k^* - \sum_{j=1}^{s}\mathbf{P}\mathbf{B}_j\mathbf{P}^{\mathsf{T}}\boldsymbol{y}_{k-j}^* + \boldsymbol{f}_k - \sum_{j=1}^{s}\mathbf{P}\mathbf{B}_j\mathbf{P}^{\mathsf{T}}\boldsymbol{f}_{k-j}\\ &= \boldsymbol{e}_k^* + \left(\mathbf{I} - \sum_{j=1}^{s}\mathbf{P}\mathbf{B}_j\mathbf{P}^{\mathsf{T}}q^{-1}\right)\boldsymbol{f}_k \end{aligned}\tag{31}$$

where $\boldsymbol{e}_k^*$ is the fault-free portion of $\boldsymbol{e}_k$ and q^{-1} is the backward-shift operator. It is seen from (31) that the impact of the fault on the prediction error is filtered, which can reduce the sensitivity to detect it.

To illustrate the reduced sensitivity, we consider the following simple example where $s = 1$, $\mathbf{B}_1 = 0.5$, $\mathbf{P} = [1 \quad 0]^{\mathsf{T}}$. Equation (31) gives

$$\begin{bmatrix} e_{1k} \\ e_{2k} \end{bmatrix} = \begin{bmatrix} e_{1k}^* \\ e_{2k}^* \end{bmatrix} + \begin{bmatrix} 1-0.5q^{-1} & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} f_{1k} \\ f_{2k} \end{bmatrix}.$$

If there is a step fault in f_{1k}, the impact magnitude on e_{1k} is 0.5 at steady state, while its impact on y_{1k} is 1.0. Since $y_{1k} = 0.5y_{1,k-1} + e_{1k}$, we have $\mathtt{var}(y_1) = (4/3)\mathtt{var}(e_1)$. The ratio of the magnitude of f_{1k} to the standard deviation of y_{1k} is

$$1/\sqrt{\mathtt{var}(y_1)} = 1/\sqrt{\frac{4}{3}\mathtt{var}(e_1)} = \frac{\sqrt{3}}{2}/\sqrt{\mathtt{var}(e_1)}$$

while the ratio of the magnitude of the fault in e_{1k} is $0.5/\sqrt{\mathtt{var}(e_1)}$. Therefore, there is a reduction in fault sensitivity when e_{1k} is used for fault detection, although it removes autocorrelation in y_{1k}.

B. PPA-Based Monitoring

Many works dealt with the monitoring of faults based on models from dynamic data with various degrees of effectiveness. One approach is to build monitoring indices on $\boldsymbol{v}_k - \hat{\boldsymbol{v}}_k$ and static noise $\bar{\boldsymbol{\varepsilon}}_k$. The approach is sound since these residuals are not serially correlated, but it can lose sensitivity to faults, as shown in the previous example. In this work, we propose an improved monitoring scheme based on (3), including monitoring the prediction residuals $\{\hat{\boldsymbol{e}}_k\}$ in (32) and the principal predictors $\{\hat{\boldsymbol{v}}_k\}$.

1) Monitoring the Prediction Residuals: Given the PPA model and the prediction

$$\hat{\boldsymbol{y}}_k = \hat{\mathbf{P}}\hat{\boldsymbol{v}}_k \tag{32}$$

we calculate the prediction error matrix $\hat{\mathbf{E}}_s = \mathbf{Y}_s - \hat{\mathbf{V}}_s\hat{\mathbf{P}}^{\mathsf{T}}$ and the prediction error $\hat{\boldsymbol{e}}_k = \boldsymbol{y}_k - \hat{\mathbf{P}}\hat{\boldsymbol{v}}_k$, and perform EVD on

$$\hat{\Sigma}_e = \hat{\mathbf{E}}_s^{\mathsf{T}}\hat{\mathbf{E}}_s/N = \mathbf{W}_e\Lambda_e\mathbf{W}_e \tag{33}$$

where $\Lambda_e = \mathrm{diag}(\lambda_{e,1}, \lambda_{e,2}, \ldots, \lambda_{e,p})$ is arranged in a descending order and contains the variances of the PCs. By selecting the number of PCs ℓ_e to capture, for example 95% of the total variance in $\hat{\mathbf{E}}_s$, we have the loadings of the PCs $\mathbf{P}_e = \mathbf{W}_e(:, 1{:}\ell_e)$.

The monitoring of $\hat{\boldsymbol{e}}_k$ can be performed using the principal loadings $\mathbf{P}_e$ to calculate the scores $\boldsymbol{t}_{e,k} = \mathbf{P}_e^{\mathsf{T}}\hat{\boldsymbol{e}}_k$ and residual

$$\tilde{\boldsymbol{e}}_k = \left(\mathbf{I} - \mathbf{P}_e\mathbf{P}_e^{\mathsf{T}}\right)\hat{\boldsymbol{e}}_k. \tag{34}$$

We define the Hotelling's index

$$T_e^2 = \sum_{i=1}^{\ell_e}\lambda_{e,i}^{-1}t_{e,ik}^2$$

where $t_{e,ik}$ is the i^{th} entry of $\boldsymbol{t}_{e,k}$. T_e^2 approximately follows a χ^2 distribution with l_e degrees of freedom, which has an upper control limit of $\chi_\alpha^2(l_e)$ with $(1-\alpha)$ as the confidence level.

The residual $\tilde{\boldsymbol{e}}_k$ is monitored by the Q index as follows:

$$Q_e = \tilde{\boldsymbol{e}}_k^{\mathsf{T}}\tilde{\boldsymbol{e}}_k. \tag{35}$$

Assuming $\hat{\boldsymbol{e}}_k$ is normally distributed with $\lambda_{e,i}$ as the variance of the i^{th} component, based on the theorem of Box [14], [44], $g_e^{-1}Q_e$ approximately follows a $\chi^2(h_e)$ distribution with an upper control limit of $\chi_\alpha^2(h_e)$, where

$$g_e = \frac{\sum_{i=\ell_e+1}^{p}\lambda_{e,i}^2}{\sum_{i=\ell_e+1}^{p}\lambda_{e,i}}; \quad h_e = \frac{\left(\sum_{i=\ell_e+1}^{p}\lambda_{e,i}\right)^2}{\sum_{i=\ell_e+1}^{p}\lambda_{e,i}^2}.$$

Sometimes, it is preferred to monitor a combined index of T_e^2 and Q_e rather than monitoring them separately, which has been proposed for static process monitoring [14], [45]. Since both $T_e^2 \sim \chi^2(\ell_e)$ and $g_e^{-1}Q_e \sim \chi^2(h_e)$ follow χ^2 distributions and are independent, the combined index

$$\phi_e = T_e^2 + g_e^{-1}Q_e \sim \chi^2(\ell_e + h_e) \tag{36}$$

has a control limit $\chi_\alpha^2(\ell_e + h_e)$ with confidence level $1-\alpha$.

2) Monitoring the Principal Predictors: From (3) it is seen that the DLV prediction $\{\hat{\boldsymbol{v}}_k\}$ should also be monitored. If a fault occurs in the past data that are used to predict $\{\hat{\boldsymbol{v}}_k\}$, it can enhance the detection of such a fault.

The monitoring of $\{\hat{\boldsymbol{v}}_k\}$ is straightforward due to Theorem 2. Since the number of DLVs ℓ is chosen to include significantly predictable ones, the covariance of the principal predictors $\hat{\boldsymbol{\Sigma}}_{\hat{\boldsymbol{v}}} = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_\ell)$ is well conditioned. The monitoring of principal predictors can be implemented with the following Hotelling's index:

$$T^2_{\hat{v}} = \hat{\boldsymbol{v}}_k^{\mathsf{T}} \hat{\boldsymbol{\Sigma}}_{\hat{\boldsymbol{v}}}^{-1} \hat{\boldsymbol{v}}_k = \sum_{i=1}^{\ell} \lambda_i^{-1} \hat{v}_{ik}^2 \sim \chi^2(\ell) \tag{37}$$

where $\hat{v}_{ik}$ is the ith entry of $\hat{\boldsymbol{v}}_k$, which has a control limit $\chi^2_\alpha(\ell)$ with confidence level $1-\alpha$.

Sometimes, it is desirable to implement an overall monitoring index for both the prediction residuals and the principal predictors. Using the fact that the principal predictors $\{\hat{\boldsymbol{v}}_k\}$ are uncorrelated to their residuals, an overall index is calculated by combining $T^2_{\hat{v}}$ with ϕ_e as

$$\phi_o = T^2_{\hat{v}} + \phi_e \sim \chi^2(\ell + \ell_e + h_e) \tag{38}$$

which has a control limit $\chi^2_\alpha(\ell+\ell_e+h_e)$ with confidence level $1-\alpha$.

V. Fault Identification and Reconstruction

Assuming $\{\boldsymbol{y}_k\}$ is fault-free up to time k and Fault i affects the subsequent data $\{y_{k+j}\}_{j=1}^m$ as follows:

$$\boldsymbol{y}_{k+j} = \boldsymbol{y}^*_{k+j} + \Xi_i \mathbf{f}_{k+j} \tag{39}$$

where $\boldsymbol{y}^*_{k+j}$ is the fault-free portion of the data and unknown after the fault happened. We would like to estimate the fault matrix $\Xi_i \in \Re^{p\times \ell_i}$ from the faulty data. The PCA-based reconstruction [46], [47] can be extended here. Denoting

$$\hat{\boldsymbol{e}}_{k+j|k} = \boldsymbol{y}_{k+j} - \hat{\mathbf{P}}\hat{\boldsymbol{v}}_{k+j|k} \tag{40}$$

where $\hat{\boldsymbol{v}}_{k+j|k}$ is the j-steps ahead prediction which is fault-free, we have

$$\hat{\boldsymbol{e}}_{k+j|k} = \hat{\boldsymbol{e}}^*_{k+j|k} + \Xi_i \boldsymbol{f}_{k+j} \tag{41}$$

where $\hat{\boldsymbol{e}}^*_{k+j|k} = \boldsymbol{y}^*_{k+j} - \hat{\mathbf{P}}\hat{\boldsymbol{v}}_{k+j|k}$ is fault free. Denoting

$$\begin{aligned}
\mathbf{E}_m &= \left[\hat{\boldsymbol{e}}_{k+1|k} \ \cdots \ \hat{\boldsymbol{e}}_{k+m|k}\right] \\
\mathbf{E}^*_m &= \left[\hat{\boldsymbol{e}}^*_{k+1|k} \ \cdots \ \hat{\boldsymbol{e}}^*_{k+m|k}\right] \\
\mathbf{F}_m &= \left[\boldsymbol{f}_{k+1} \ \cdots \ \boldsymbol{f}_{k+m}\right]
\end{aligned}$$

we have

$$\mathbf{E}_m = \mathbf{E}^*_m + \Xi_i \mathbf{F}_m. \tag{42}$$

When the fault magnitude $\boldsymbol{f}_{k+j}$ becomes significant, the normal prediction residuals $\mathbf{E}^*_m$ are relatively insignificant, therefore, Ξ_i and $\mathbf{E}_m$ approximately share the column space. By performing SVD on $\mathbf{E}_m = \mathbf{U}_m\mathbf{D}_m\mathbf{V}_m^{\mathsf{T}}$, we can choose $\Xi_i = \mathbf{U}_m(:, 1{:}\ell_i)$ as the estimated fault directions.

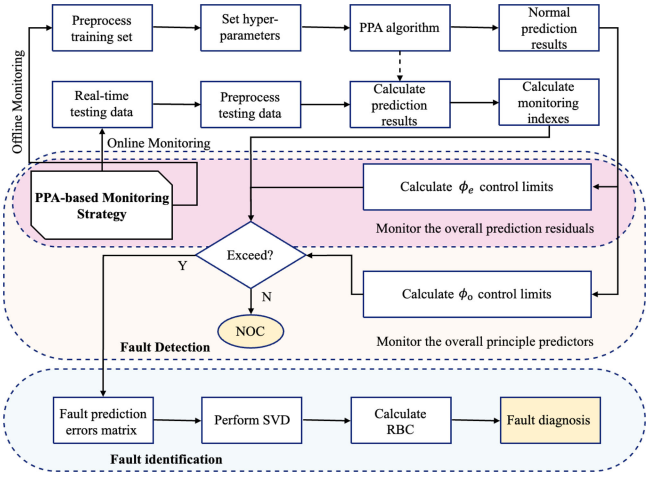

Fig. 1. Flowchart of the PPA-based modeling and monitoring strategy.

We propose to select ℓ_i incrementally so that the residuals $\mathbf{E}_m$ in (42) can be reconstructed within its normal control limit. Denoting the reconstruction of $\mathbf{E}^*_m$ in (42) as

$$\mathbf{E}^r_m = \mathbf{E}_m - \Xi_i \mathbf{F}^r_m \tag{43}$$

the best reconstruction is to minimize $\|\mathbf{E}^r_m\|^2_F$, which leads to

$$\mathbf{F}^r_m = \Xi_i^{\mathsf{T}} \mathbf{E}_m \tag{44}$$

$$\mathbf{E}^r_m = \left(\mathbf{I} - \Xi_i \Xi_i^{\mathsf{T}}\right)\mathbf{E}_m. \tag{45}$$

Denoting $\hat{\boldsymbol{e}}^r_{k+j|k}$ as the j^{th} column of $\mathbf{E}^r_m$ to calculate $\tilde{\boldsymbol{e}}^r_{k+j|k}$ using $\tilde{\boldsymbol{e}}^r_{k+j|k} = (\mathbf{I} - \mathbf{P}_e\mathbf{P}_e^{\mathsf{T}})\hat{\boldsymbol{e}}^r_{k+j|k}$ and $\phi^r_e(k+j)$ using (36), $\phi^r_e(k+j)$ can be brought back to the normal control limit by increasing ℓ_i. The smallest ℓ_i that brings $\phi^r_e(k+j)$ back to the normal control limit is selected. This is the reconstruction-based approach to selecting ℓ_i.

After the effective ℓ_i is determined, the columns of $\Xi_i = \mathbf{U}_m(:, 1{:}\ell_i)$ can be used to define the RBC of the fault as

$$\texttt{RBC} = \texttt{diag}\{\Xi_i \mathbf{D}^2_m(1{:}\ell_i, 1{:}\ell_i)\Xi_i^{\mathsf{T}}\} \tag{46}$$

where $\texttt{diag}\{\cdot\}$ denotes the diagonal elements of the matrix. The elements of RBC serve as contributions to the fault of each variable. Fig. 1 shows the flow chart of the offline modeling and online monitoring strategy based on PPA.

VI. Case Studies

A. Illustrative Example

We simulate a simple dynamic example with a process fault as shown in Fig. 2. The process disturbance is measured as $y_{2k} \sim \mathcal{N}(0, 1)$. The measured control variable is

$$y_{1k} = \frac{1}{1 + \frac{Kq^{-1}}{1-q^{-1}}} \frac{1}{1-q^{-1}} y_{2k} = \frac{1}{1-(1-K)q^{-1}} y_{2k}.$$

It is straightforward to calculate that the variance of y_{1k} is $(1/[1-(1-K)^2])$ since the variance of y_{2k} is one.

The measured time series of $\boldsymbol{y}_k = [y_{1k} \ \ y_{2k}]^{\mathsf{T}}$ is analyzed with the proposed PPA for process monitoring. We use the normal process gain of $K = 0.5$ to generate 100 fault-free samples as training data to build a model for process

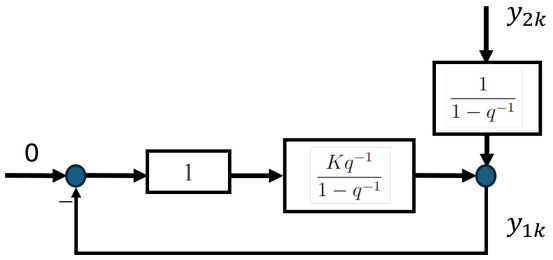

Fig. 2. Illustrative example of a process fault under feedback control.

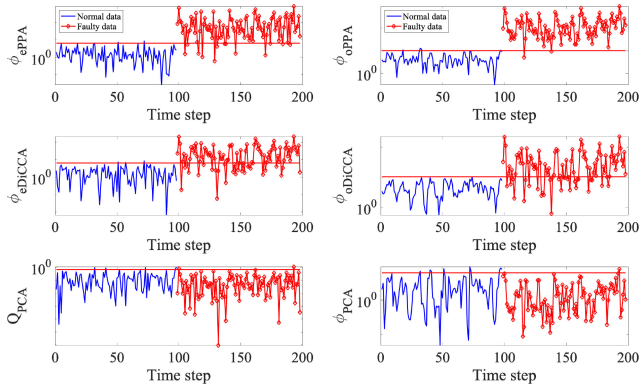

Fig. 3. Monitoring charts of ϕ_e and ϕ_o using 2 DLVs compared to DiCCA and PCA-based monitoring for the illustrative example.

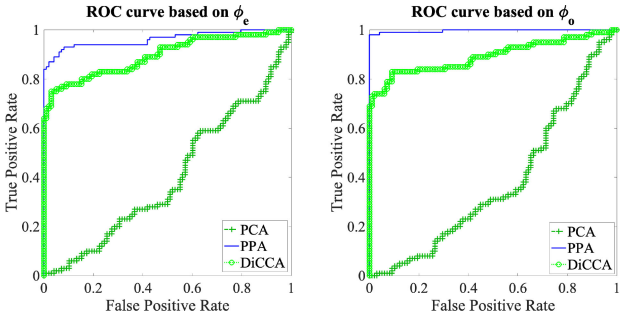

Fig. 4. ROC curves of DiCCA and PCA-based methods based on ϕ_e and ϕ_o monitoring indices for the illustrative example.

monitoring. A process gain fault occurs with $K = 1.5$ and 100 new fault samples are generated as a test set to test the effectiveness of the PPA, dynamic inner CCA (DiCCA), and PCA algorithms. We specifically design the change in K that does not lead to a change in the variance of the process output data. The PCA focusing on variance changes does not detect any anomalies. Fig. 3 shows the fault monitoring results of the example with the three methods. For better visual representation, the monitoring indices using PPA, DiCCA, and PCA have been transformed to a scale based on $\log_{10}$. It is clearly seen that both PPA and DiCCA are able to effectively detect the fault, but PCA that focuses on variance only fails completely.

It is further noted that the fault detection indices of PPA and DiCCA in Fig. 3 can detect the process gain change immediately after its occurrence, which requires few samples to detect. This advantage of PPA and DiCCA makes them practical since real-world operations typically have much fewer anomalous data than normal data. The numerical result of $\phi_{e\mathrm{PPA}}$ having a lower detection date than $\phi_{o\mathrm{PPA}}$ verifies the analysis in Section V(A), where dynamic residual-based monitoring tends to reduce the sensitivity for fault detection, although it removes temporal dependence. $\phi_{o\mathrm{PPA}}$ should be preferred to achieve high detection rates.

Although this fault detection problem is different from a classification problem where two classes of data are used to define a classifier, it is of interest to show receiver operating characteristic (ROC) curves to assess the ability to separate the normal and faulty samples based on the monitoring indices generated with various models. Fig. 4 gives the ROC curves to compare the area under the curve (AUC) values of PPA, DiCCA, and PCA based on ϕ_e and ϕ_o monitoring indices, revealing that the PPA model has the highest AUC values at 0.92 and 0.97 for ϕ_e and ϕ_o, respectively, followed by DiCCA at 0.83 and 0.85, and PCA at 0.48 and 0.46.

B. Application to the Dow Challenge Data Problem

In this article, the Dow Challenge dataset [39] is adopted to demonstrate the utility and test the effectiveness of the proposed PPA for process monitoring. The process with three distillation columns tends to accumulate impurities caused by the aging of the catalyst. There are two data sets provided for the Dow Challenge problem. One was collected from December 2015 to January 2017 [39], while the other was collected from February 2017 to October 2017. Changes in operation mode and anomalies were evident in the datasets. The data period from January 1, 2016 to May 31, 2016 contains few anomalies and is used to build models for monitoring purposes. The subsequent 15 days are used for the monitoring and diagnosis of anomalies and disturbances. The data set contains missing values and outliers that need to be preprocessed prior to building the normal models.

The flow diagram of the Dow Challenge process is depicted in Fig. 5 with relevant flow variables labeled. The primary column is chosen for the monitoring and diagnosis of faults with the 15 variables shown in Table I. Four other variables with virtually no variation are excluded. Based on the mass conservation of the process in Fig. 5, we can calculate two new variables

$$\mathtt{Disposal} = x_{24} \cdot 1000 - \mathtt{Input} \quad \text{with} \tag{47}$$

$$\mathtt{Input} = x_3 + x_4 + x_{23} \tag{48}$$

where x_3, x_4, x_{23}, and x_{24} are the Input Flow to Primary Column Bed 3, Input Flow to Primary Column Bed 2, Input Flow to Secondary Column, and the Secondary Column Tails Flow, respectively. The coefficient 1000 is used to scale all variables with the same unit. The Disposal flow in (47) is included in this case study.

1) Data Preprocessing for Dynamic Modeling: The training data in the Dow dataset have missing values and outliers. For dynamic data modeling, we cannot simply delete the outlying or missing samples since we must maintain the integrity of the time sequences. Therefore, we need to reconstruct or regenerate the samples that are missing or outlying using a dynamic model. In this article, the following steps are adopted

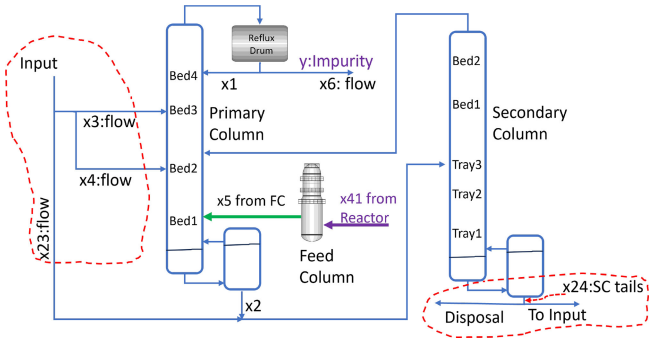

Fig. 5. Dow Challenge process flow diagram with relevant flow variables.

TABLE I
15 MEASURED PROCESS VARIABLES OF THE PRIMARY COLUMN IN THE DOW CHALLENGE DATASET

Variable	Description
x_1	Reflux Flow
x_2	Tails Flow
x_5	Feed Flow from the Feed Column (FC)
x_6	Make Flow
x_7	Base Level
x_{10}	Bed1 DP
x_{11}	Bed2 DP
x_{12}	Bed3 DP
x_{13}	Bed4 DP
x_{14}	Base Pressure
x_{15}	Head Pressure
x_{18}	Bed 4 Temperature
x_{19}	Bed 3 Temperature
x_{20}	Bed 2 Temperature
x_{21}	Bed 1 Temperature

to preprocess and curate a refined training set for subsequent modeling.

1) Missing values are replaced by interpolating between neighboring valid values.
2) An initial PCA model is built to generate the Q index to detect outliers that exceed the control limit with super-high confidence 99.9%, as shown in the top panel of Fig. 6. It is evident in the figure that sporadically high outliers are detected. These outliers are further treated using the missing-value replacement procedure in the previous step. There are also high Q values in three segments from around 13-Feb, 07-Mar, and 01-Apr. These periods could be caused by normal dynamics or disturbances. Subsequently, dynamic PPA analysis is performed to decide if they are dynamics or disturbances.
3) A PPA model is built on the PCA-preprocessed training data with the ϕ_e index shown in the middle panel of Fig. 6. It is seen that two periods of high Q values from around 13-Feb, and 07-Mar appear normal, which shows that the first two periods are normal dynamic variations, but the period around 01-Apr remains high and anomalous. A comparison between the predicted and actual values in Fig. 7 shows that the "Bed 4 DP" values in this period are anomalous, since the actual and predicted "Bed 4 DP" values deviate significantly after a step change from around 01-Apr. Consequently, the predicted "Bed 4 DP" values for the period from 01-Apr are used as PPA-processed data to build a refined PPA model for monitoring and diagnosis. The resulting ϕ_e index of the refined PPA model is shown in the bottom panel of Fig. 6, which is normal except for some sporadic outlying values, leading to an acceptable Type-I error of 4.25%.

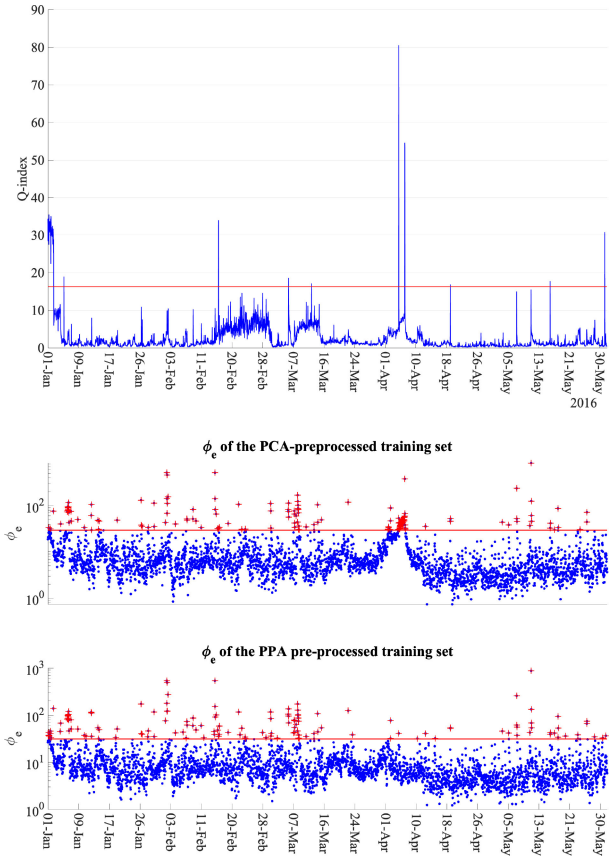

Fig. 6. (Top panel) Q-index of PCA for outlier detection with 99.9% confidence; (Middle panel) ϕ_e of PPA using the PCA-preprocessed training data; (Bottom panel) ϕ_e of the PPA model using PPA-processed training data.

With the preprocessed data of the 16 variables, we conduct the following two scenarios of fault detection and reconstruction and compare the results.

1) Apply the PPA based method in this article to the 16 variables.
2) Using the macro-balance of the process in Fig. 5,

$$1000 \cdot (x_5 - x_6) - \texttt{Disposal} = 0$$

perform residual calculations using (47) and (28) with $C = [1000, -1000, -1]$ for x_5, x_6, and Disposal and other elements in C are 0. Then investigate the effectiveness of PPA for fault detection and diagnosis.

2) Proportion of Predictable Variance and the Number of DLVs: To illustrate the effectiveness of PPA in capturing the highest PPVs in the data, we built PPA, LaVAR-CCA, and

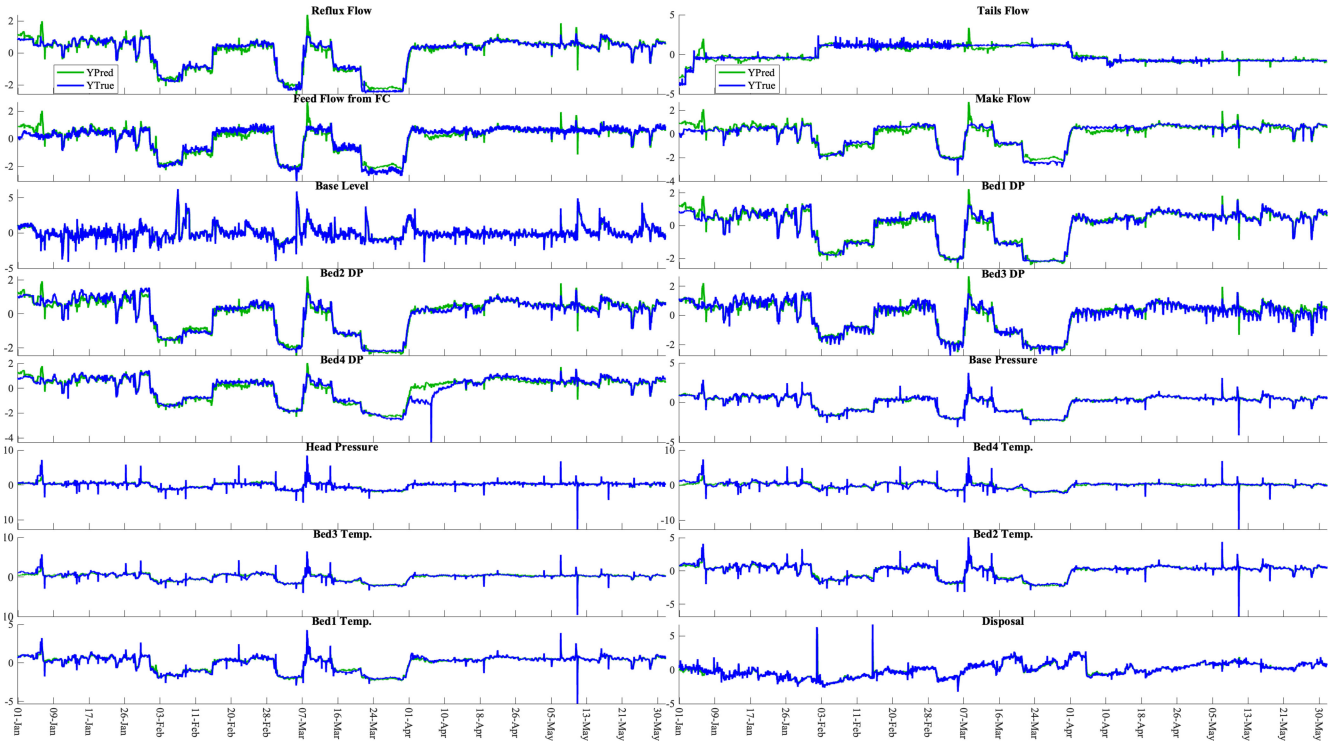

Fig. 7. PPA predicted values and true values of the 16 variables using the PCA-preprocessed training data.

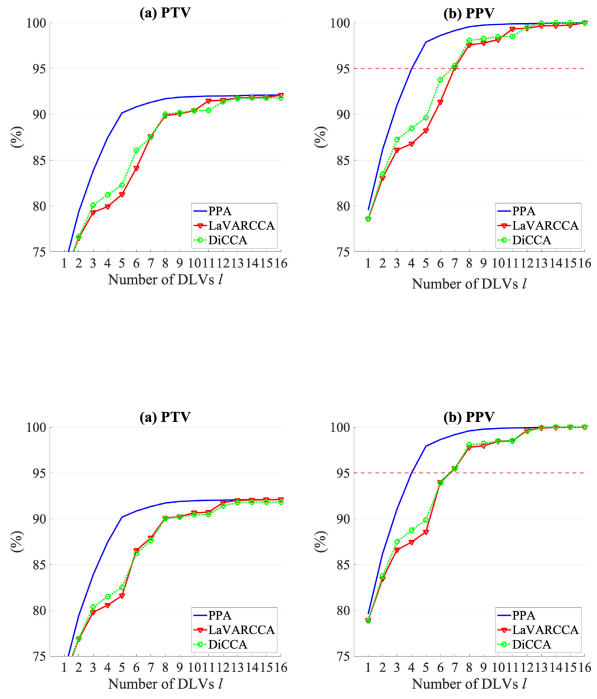

Fig. 8. PTVs and PPVs for PPA, LaVAR, and DiCCA models of the Dow Challenge dataset: (Top panels) original variables; (Bottom panels) residualized variables.

DiCCA models on the training data of the original variables and residualized variables in the top panels and bottom panels of Fig. 8, respectively. It is evident that PPA is capable of achieving the highest PTV for the same number of DLVs among the three methods. This result is due to the maximized covariance objective in the PPA model, while in the other two models the canonical correlations are maximized.

The number of PPA DLVs ℓ is chosen with PPV to account for at least 95% of the total predictable variances. It is seen that $\ell = 4$ is sufficient to account for 95% of PPVs for both PPA models. For comparison, the PPV(ℓ) and PTV(ℓ) for LaVARCCA and DiCCA are also shown in the figure.

3) Anomaly Detection and Diagnosis of the Test Data: After the final PPA models and the corresponding monitoring control limits are obtained, we apply them for the fault detection and diagnosis on the test set from June 1 to June 15. Fig. 9 shows the ϕ_e monitoring charts for the test set with the original variables in the top panel and residualized variables in the bottom panel. The ϕ_e indices have two periods of very high values that exceed the respective control limits. They are two periods of unknown disturbances that are detected from the 16 process variables. To verify that these disturbances have a significant impact on product impurity, Fig. 10 shows the impurity samples of the training and test sets. The two periods are labeled Disturbance 1 and Disturbance 2 for further diagnosis.

For the two detected disturbance periods, we perform the fault diagnosis with reconstruction using (46) and (45). To determine the appropriate number of fault directions to adequately reconstruct the monitoring index ϕ_e, Fig. 11 shows the reconstructed ϕ_e^r and the original ϕ_e in logarithmic scale for the two disturbances. It reveals that $\ell_1 = 7$ and $\ell_2 = 7$ are needed to reconstruct Disturbances 1 and 2 in the original data, while $\ell_1 = 5$ and $\ell_2 = 7$ are sufficient to reconstruct Disturbances 1 and 2 in the residualized data. The subtle difference is due to the fact that the residualized data excluded one mass conservation relation.

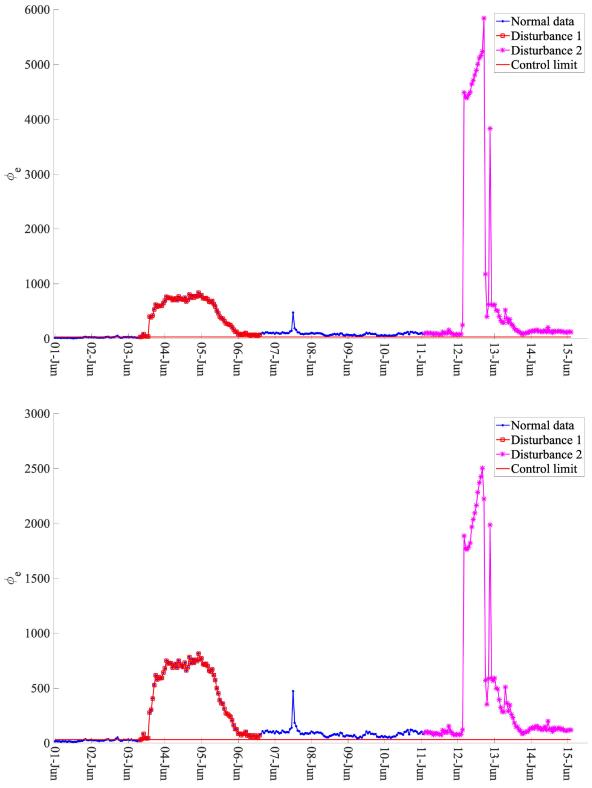

Fig. 9. PPA-based ϕ_e detects Disturbances 1 and 2 in the test set: (Top panel) original variables; (Bottom panel) residualized variables.

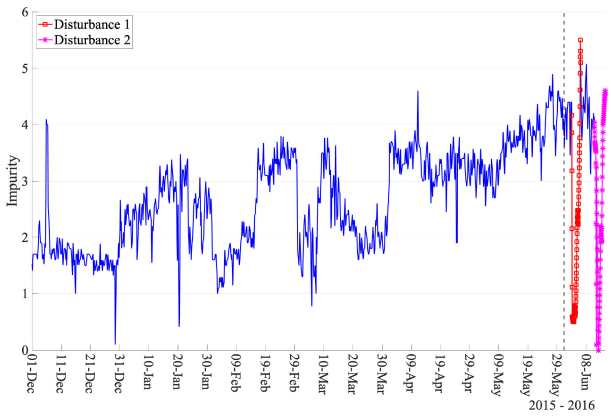

Fig. 10. Training data and test data separated with a vertical dashed line. The two PPA-detected disturbance periods are highlighted with different colors.

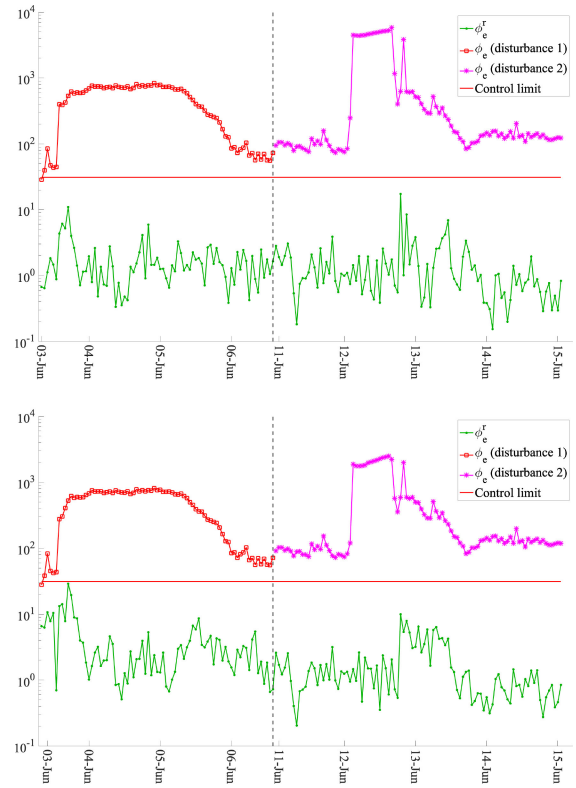

Fig. 11. Reconstructed ϕ_e^r and the original ϕ_e of Disturbances 1 and 2: (Top two panels) original variables; (Bottom two panels) residualized variables.

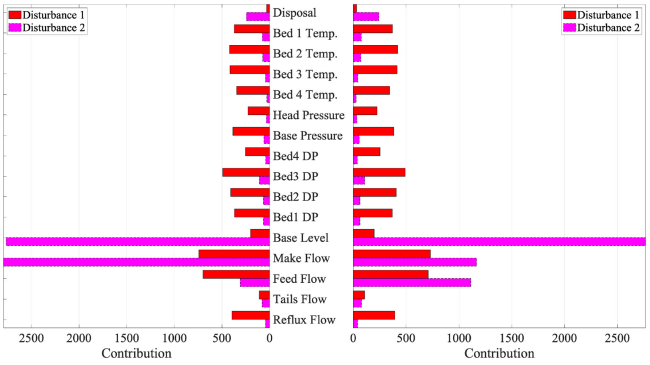

Fig. 12. RBC of each variable to Disturbances 1 and 2: (Left panel) using PPA of the original variables; (Right panel) using PPA of the residualized variables.

Fig. 12 shows the RBCs of Disturbances 1 and 2, with the left panel for the original variables and the right panel for the residualized variables. The two disturbances are seen to have very different signatures. Disturbance 1 seems to have uniform contributions from nearly all variables. On the other hand, for Disturbance 2 "Make Flow" and "Base Level" have dominant contributions using the original data, while "Feed Flow" appears as another strong contributor for the result using the residualized data. Since Feed Flow and Make Flow are the process inlet and output flows that co-move with the throughput, the diagnosis result with the residualized variables in the right panel gives a complete picture of the root cause in Disturbance 2. This result shows the advantage of incorporating first-principles relations in the PPA model for diagnosis.

With the reconstructed values using (44), it is possible to generate for what the normal operation data would look like if there were no disturbances. Denoting a period of anomalous data $\{y_{k+j}\}_{j=1}^{m}$ with $\mathbf{Y}_m = [\boldsymbol{y}_{k+1} \ \cdots \ \boldsymbol{y}_{k+m}]$, the reconstructed data is

$$\mathbf{Y}_m^r = \mathbf{Y}_m - \Xi_i \mathbf{F}_m^r. \tag{49}$$

We generate the reconstructed data for the detected disturbance period in the training set as well as the two disturbances in the test set, and compare them with their actual values as shown in Fig. 13 with the PPA model of the original data and in Fig. 14 with the PPA model of the residualized data. It is clear that

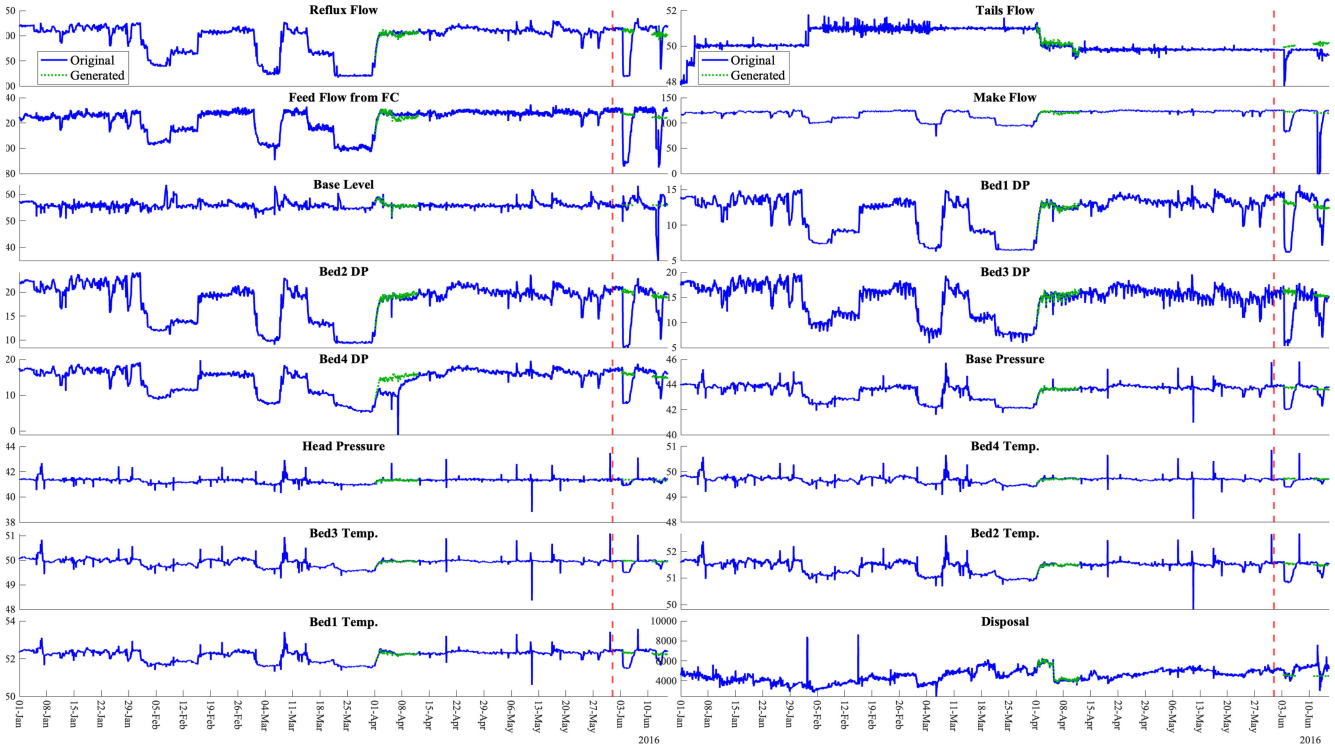

Fig. 13. Original (blue) and PPA-generated (green) values for the anomaly periods in the training and test sets with the PPA model of the original data.

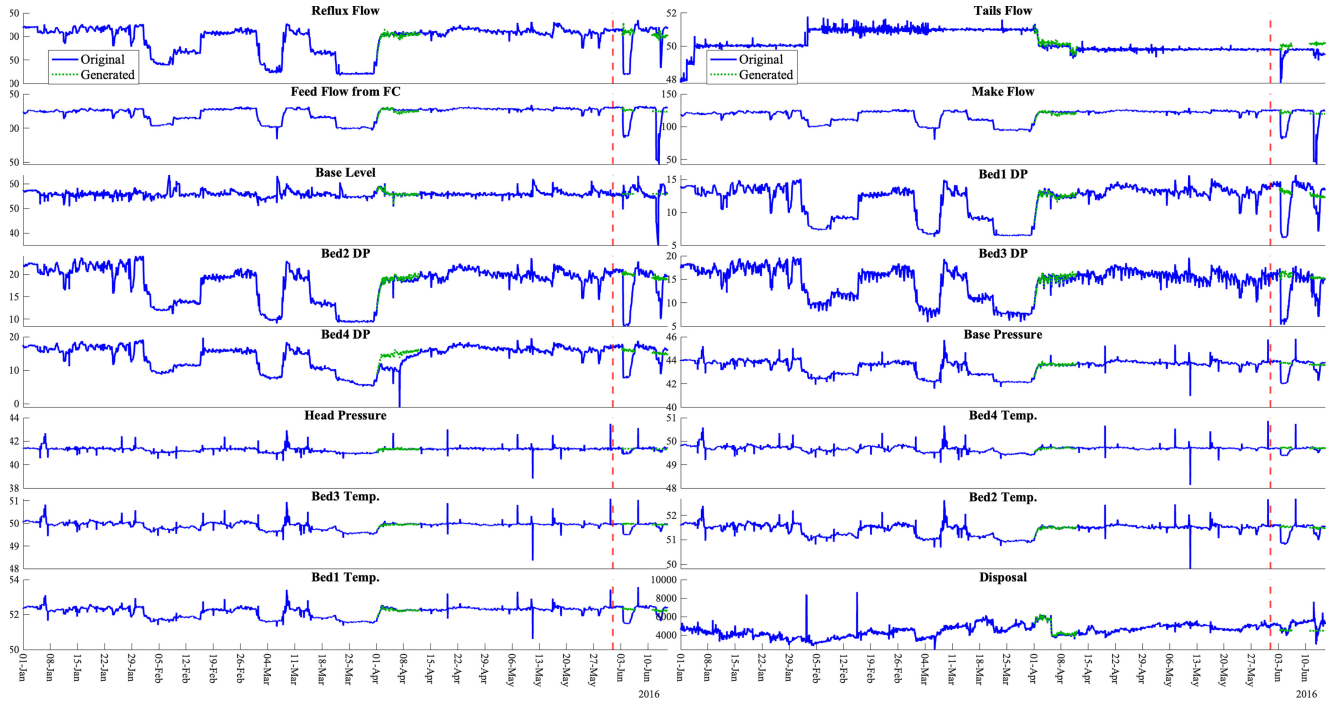

Fig. 14. Original (blue) and generated (green) values for the anomaly periods in the training and test sets with the PPA model of the residualized data.

the original values for these three disturbances show a notable discrepancy from their normal state, but the reconstructed values in green share similar characteristics to the normal data, demonstrating the effectiveness of the reconstruction method for data generation and curation. Furthermore, the test data for Disturbance 1 show discrepancies in nearly all variables, while those for Disturbance 2 show large discrepancies in Feed Flow, Make Flow, and Base Level. This detailed result verifies the result of the RBC-based diagnosis in Fig. 12.

VII. Conclusion

This article develops a PPA method and applies it to the monitoring of multidimensional time series data. The maximum predicted variance of the latent predictors is achieved with orthonormal loadings. The objective separates predictable content from unpredictable content in multiple time series and differs from the maximum-variance objective in PCA that does not focus on predictions. The number of principal predictors is selected to capture a high PPV for applications, such as process monitoring. Using the Dow Challenge process data, the PPA model is shown to capture more predicted variances than other competing latent dynamic models with the same number of latent variables.

The application of the PPA method to the Dow Challenge process data shows its significant effectiveness in fault detection and the reconstruction-based diagnosis. The real application also shows that many practical issues must be addressed in data-driven monitoring, including data cleaning and preprocessing. The application to the Dow Challenge data with incorporating first-principles relations shows the advantage of the reconstruction-based contribution for the diagnosis of root causes. In addition, the effectiveness of dynamics-based monitoring over PCA-based monitoring is clearly demonstrated with a closed-loop control example.

For future work, it is possible to extend PPA to deal with missing data using either the EM framework or the reconstruction technique proposed in the article. Furthermore, the proposed PPA with orthonormal loadings is simpler than our previous work [34], which implements oblique projections to achieve uncorrelated dynamic and static noise terms. These solutions represent different realizations based on Theorem 1. Although their pros and cons are likely problem-dependent, PPA achieves maximal predicted variances for a given number of latent variables.

Appendix
Proof of Theorem 2

Since

$$\begin{aligned}\hat{\mathbf{V}}_s &= \hat{\mathbf{V}}_s\hat{\mathbf{V}}_s^+\hat{\mathbf{V}}_s = \hat{\mathbf{V}}_s\left(\hat{\mathbf{V}}_s^\intercal\hat{\mathbf{V}}_s\right)^{-1}\hat{\mathbf{V}}_s^\intercal\hat{\mathbf{V}}_s \\ &= \hat{\mathbf{V}}_s\left(\hat{\mathbf{V}}_s^\intercal\hat{\mathbf{V}}_s\right)^{-1}\hat{\mathbf{V}}_s^\intercal\mathbf{V}_s \\ &= \hat{\mathbf{V}}_s\hat{\mathbf{V}}_s^+\mathbf{V}_s,\end{aligned}$$

we have from (24), (25), and (26),

$$\begin{aligned}\hat{\boldsymbol{\Sigma}}_{\hat{\mathbf{v}}} &= \hat{\mathbf{V}}_s^\intercal\hat{\mathbf{V}}_s/N = \mathbf{V}_s^\intercal\hat{\mathbf{V}}_s/N = \mathbf{V}_s^\intercal\hat{\mathbf{V}}_s\hat{\mathbf{V}}_s^+\mathbf{V}_s/N \\ &= \mathbf{P}^\intercal\mathbf{Y}_s^\intercal\hat{\mathbf{V}}_s\hat{\mathbf{V}}_s^+\mathbf{Y}_s\mathbf{P}/N = \mathbf{P}^\intercal\mathbf{W}\Lambda\mathbf{W}^\intercal\mathbf{P}. \qquad (50)\end{aligned}$$

To maximize the diagonal elements of $\hat{\boldsymbol{\Sigma}}_{\hat{\mathbf{v}}}$, the only choice is $\mathbf{P} = \mathbf{W}(:, 1{:}\ell)$, which leads to (27) with this choice. It is also convenient to choose $\bar{\mathbf{P}} = \mathbf{W}(:, \ell+1{:}p)$ that makes it orthogonal to $\mathbf{P}$.

References

[1] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Phil. Mag.*, vol. 2, pp. 559–572, Nov. 1901.

[2] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *J. Educ. Psychol.*, vol. 24, no. 6, pp. 417–441, 1933.

[3] I. Jolliffe, *Principal Component Analysis*, 2nd ed.New York, NY, USA: Springer, 2002.

[4] E. Oja, "Simplified neuron model as a principal component analyzer," *J. Math. Biol.*, vol. 15, no. 3, pp. 267–273, Nov. 1982.

[5] M. Kramer, "Nonlinear principal component analysis using autoassociative neural networks," *AIChE J.*, vol. 37, no. 2, pp. 233–243, 1991.

[6] H. Bourlard and Y. Kamp, "Auto-association by multilayer perceptrons and singular value decomposition," *Biol. Cybern.*, vol. 59, no. 4, pp. 291–294, 1988.

[7] H. Bourlard and S. Kabil, "Autoencoders reloaded," *Biol. Cybern.*, vol. 116, pp. 389–406, Aug. 2022.

[8] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[9] J. Zhang, H. Chen, S. Chen, and X. Hong, "An improved mixture of probabilistic PCA for nonlinear data-driven process monitoring," *IEEE Trans. Cybern.*, vol. 49, no. 1, pp. 198–210, Jan. 2019.

[10] C. Wu, H. Chen, B. Du, and L. Zhang, "Unsupervised change detection in multitemporal VHR images based on deep kernel PCA convolutional mapping network," *IEEE Trans. Cybern.*, vol. 52, no. 11, pp. 12084–12098, Nov. 2022.

[11] Z. Xia, Y. Chen, and C. Xu, "Multiview PCA: A methodology of feature extraction and dimension reduction for high-order data," *IEEE Trans. Cybern.*, vol. 52, no. 10, pp. 11068–11080, Oct. 2022.

[12] J. MacGregor and T. Kourti, "Statistical process control of multivariate processes," *Control Eng. Pract.*, vol. 3, no. 3, pp. 403–414, 1995.

[13] L. Chiang, E. Russell, and R. Braatz, *Fault Detection and Diagnosis in Industrial Systems* (Advanced Textbooks in Control and Signal Processing). London, U.K.: Springer-Verlag, 2001.

[14] S. J. Qin, "Statistical process monitoring: Basics and beyond," *J. Chemomet.*, vol. 17, nos. 8–9, pp. 480–502, 2003.

[15] A. Cinar, A. Palazoglu, and F. Kayihan, *Chemical Process Performance Evaluation*.Boca Raton, FL, USA: Taylor & Francis/CRC Press, 2007.

[16] X. Ma, D. Wu, S. Gao, T. Hou, and Y. Wang, "Autocorrelation feature analysis for dynamic process monitoring of thermal power plants," *IEEE Trans. Cybern.*, vol. 53, no. 8, pp. 5387–5399, Aug. 2023.

[17] J. Li, D. Ding, and F. Tsung, "Directional PCA for fast detection and accurate diagnosis: A unified framework," *IEEE Trans. Cybern.*, vol. 52, no. 11, pp. 11362–11372, Nov. 2022.

[18] S. J. Qin, "Latent vector autoregressive modeling and feature analysis of high dimensional and noisy data from dynamic systems," *AIChE J.*, vol. 68, no. 6, 2022, Art. no. e17703.

[19] G. Li, S. J. Qin, and D. Zhou, "A new method of dynamic latent-variable modeling for process monitoring," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6438–6445, Nov. 2014.

[20] Y. Dong and S. J. Qin, "Dynamic-inner partial least squares for dynamic data modeling," *IFAC-PapersOnLine*, vol. 48, no. 8, pp. 117–122, 2015.

[21] Y. Dong and S. J. Qin, "A novel dynamic PCA algorithm for dynamic data modeling and process monitoring," *J. Process Control*, vol. 67, pp. 1–11, 2018.

[22] Y. Dong, Y. Liu, and S. J. Qin, "Efficient dynamic latent variable analysis for high-dimensional time series data," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4068–4076, Jun. 2020.

[23] Q. She, Y. Gao, K. Xu, and R. Chan, "Reduced-rank linear dynamical systems," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 4050–4057.

[24] L. Zhou, G. Li, Z. Song, and S. J. Qin, "Autoregressive dynamic latent variable models for process monitoring," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 1, pp. 366–373, Jan. 2017.

[25] D. Pena, E. Smucler, and V. J. Yohai, "Forecasting multiple time series with one-sided dynamic principal components," *J. Amer. Stat. Assoc.*, vol. 114, no. 528, pp. 1683–1694, 2019.

[26] C. Lam, Q. Yao, and N. Bathia, "Estimation of latent factors for high-dimensional time series," *Biometrika*, vol. 98, no. 4, pp. 901–918, 2011.

[27] J. Pan and Q. Yao, "Modelling multiple time series via common factors," *Biometrika*, vol. 95, no. 2, pp. 365–379, 2008.

[28] Z. Gao and R. S. Tsay, "Modeling high-dimensional time series: A factor model with dynamically dependent factors and diverging eigenvalues," *J. Amer. Stat. Assoc.*, vol. 117, pp. 1–17, Feb. 2021.

[29] T. Blaschke, T. Zito, and L. Wiskott, "Independent slow feature analysis and nonlinear blind source separation," *Neural Comput.*, vol. 19, no. 4, pp. 994–1021, 2007.

[30] S. Richthofer and L. Wiskott, "Predictable feature analysis," in *Proc. IEEE 14th Int. Conf. Mach. Learn. Appl. (ICMLA)*, 2015, pp. 190–196.

[31] S. J. Qin, Y. Dong, Q. Zhu, J. Wang, and Q. Liu, "Bridging systems theory and data science: A unifying review of dynamic latent variable analytics and process monitoring," *Annu. Rev. Control*, vol. 50, pp. 29–48, Oct. 2020.
[32] Y. Dong and S. J. Qin, "Dynamic latent variable analytics for process operations and control," *Comput. Chem. Eng.*, vol. 114, pp. 69–80, Jun. 2018.
[33] S. J. Qin, Y. Liu, and Y. Dong, "Plant-wide troubleshooting and diagnosis using dynamic embedded latent feature analysis," *Comput. Chem. Eng.*, vol. 152, Sep. 2021, Art. no. 107392.
[34] Y. Mo, J. Yu, and S. J. Qin, "Probabilistic reduced-dimensional vector autoregressive modeling for dynamics prediction and reconstruction with oblique projections," in *Proc. 62nd IEEE Conf. Decis. Control*, 2023, pp. 7623–7628.
[35] Y. Mo and S. J. Qin, "Probabilistic reduced-dimensional vector autoregressive modeling with oblique projections," *Automatica*, vol. 180, Oct. 2025, Art. no. 112476.
[36] J. Yu, Y. Dong, and S. J. Qin, "Kernel latent state-space modeling for nonlinear dynamic process monitoring," *IEEE Trans. Ind. Informat.*, vol. 21, no. 9, pp. 6701–6711, Sep. 2025.
[37] J. Yu and S. J. Qin, "Latent state space modeling of high-dimensional time series with a canonical correlation objective," *IEEE Control Syst. Lett.*, vol. 6, pp. 3469–3474, 2022.
[38] J. Yu and S. J. Qin, "Low-dimensional latent state space identification with application to the shell control process," in *Proc. IEEE Conf. Control Technol. Appl.*, 2023, pp. 1004–1009.
[39] B. Braun et al., "Data science challenges in chemical manufacturing," in *Proc. IFAC World Congr.*, Berlin, Germany, 2020, pp. 1–4.
[40] H. Lütkepohl, *New Introduction to Multiple Time Series Analysis*.Berlin Germany: Springer, 2005.
[41] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *J. Royal Stat. Soc. B*, vol. 61, no. 3, pp. 611–622, 1999.
[42] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions* (Probability and Statistics), 2nd ed. Hoboken, NJ, USA: Wiley, 2008.
[43] Y. Dong and S. J. Qin, "New dynamic predictive monitoring schemes based on dynamic latent variable models," *Ind. Eng. Chem. Res.*, vol. 59, no. 6, pp. 2353–2365, 2020.
[44] G. E. P. Box, "Some theorems on quadratic forms applied in the study of analysis of variance problems, I. Effect of inequality of variance in the one-way classification," *Ann. Math. Stat.*, vol. 25, no. 2, pp. 290–302, 1954.
[45] A. Raich and A. Cinar, "Statistical process monitoring and disturbance diagnosis in multivariate continuous processes," *AIChE J.*, vol. 42, no. 4, pp. 995–1009, 1996.
[46] R. Dunia and S. J. Qin, "Subspace approach to multidimensional fault identification and reconstruction," *AIChE J.*, vol. 44, no. 8, pp. 1813–1831, 1998.
[47] C. F. Alcala and S. J. Qin, "Reconstruction-based contribution for process monitoring," *Automatica*, vol. 45, no. 7, pp. 1593–1600, 2009.

Shumei Chen received the B.E. degree in industrial engineering from Nanjing University of Science and Technology, Nanjing, China, in 2017, and the M.S. degree in mechanical engineering from Tongji University, Shanghai, China, in 2020. She is currently pursuing the Ph.D. degree in data science with Lingnan University, Hong Kong.

Her research interests include data science and analytics, statistical and machine learning, process monitoring, and fault diagnosis.

S. Joe Qin received the B.S. and M.S. degrees in automatic control from Tsinghua University, Beijing, China, in 1984 and 1987, respectively, and the Ph.D. degree in chemical engineering from the University of Maryland at College Park, College Park, MD, USA, in 1992.

He is the Wai Kee Kau Chair Professor of Data Science and the President of Lingnan University, Hong Kong. His research interests include data science and analytics, statistical and machine learning, process monitoring, model predictive control, system identification, smart manufacturing, smart cities, smart energy management, industrial AI, and predictive maintenance.

Dr. Qin is the recipient of the 2022 CAST Computing Award by AIChE, 2022 IEEE Control Systems Society Transition to Practice Award, and the U.S. NSF CAREER Award. His h-indices for Web of Science, SCOPUS, and Google Scholar are 71, 76, and 90, respectively. He is an elected member of the European Academy of Sciences and Arts and a Fellow of the Hong Kong Academy of Engineering, the U.S. National Academy of Inventors, IFAC, and AIChE.