Taylor & Francis
Taylor & Francis Group

Journal of Quality Technology
A Quarterly Journal of Methods, Applications and Related Topics

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ujqt20

In-profile monitoring for cluster-correlated data in advanced manufacturing system

Peiyao Liu, Juan Du, Yangyang Zang, Chen Zhang & Kaibo Wang

To cite this article: Peiyao Liu, Juan Du, Yangyang Zang, Chen Zhang & Kaibo Wang (2022): In-profile monitoring for cluster-correlated data in advanced manufacturing system, Journal of Quality Technology, DOI: 10.1080/00224065.2022.2106912

To link to this article: https://doi.org/10.1080/00224065.2022.2106912

JOURNAL OF QUALITY TECHNOLOGY
https://doi.org/10.1080/00224065.2022.2106912

In-profile monitoring for cluster-correlated data in advanced manufacturing system

Peiyao Liu[a], Juan Du[b,c], Yangyang Zang[d], Chen Zhang[a], and Kaibo Wang[a]

[a]Department of Industrial Engineering, Tsinghua University, Beijing, P.R. China; [b]Smart Manufacturing Thrust, Systems Hub, The Hong Kong University of Science and Technology, Guangzhou, P.R. China; [c]Department of Mechanical and Aerospace Engineering, The Hong Kong University of Science and Technology, Hong Kong SAR, China; [d]AVIC China Aero-Polytechnology Establishment, Beijing, P.R. China

ABSTRACT

Nowadays advanced sensing technology enables real-time data collection of key variables during manufacturing, known as multi-channel profiles. These data facilitate in-process monitoring and anomaly detection, which have been extensively studied in recent years. However, most studies treat each profile as a whole, e.g., a high-dimensional vector or function, and construct monitoring schemes accordingly. As a result, these methods cannot be implemented until the entire profile has been obtained, leading to long detection delay especially if anomalies occur in early sensing points of the profile. In addition, they require that profiles of different samples have the same time length and feature location, yet additional time-warping operation for real misaligned samples may weaken the anomaly patterns. To address these problems, this article proposes an in-profile monitoring (INPOM) control chart, which not only gives the feasibility of detecting anomalies inside the profile, but also can handle the misalignment problem of different samples. In particular, our INPOM scheme is built upon state space model (SSM). To better describe the clustered between-profile correlation and avoid overfitting, SSM is extended to a regularized SSM (RSSM), where regularizations are imposed as prior information and expectation maximization algorithm is integrated for posterior maximization to efficiently learn the model parameters. Furthermore, a monitoring statistic based on one-step-ahead prediction error of RSSM is constructed for INPOM control chart. Thorough numerical studies and real case studies demonstrate the effectiveness and applicability of our proposed RSSM-INPOM framework.

KEYWORDS

between-profile correlation; cluster-correlated data; in-profile monitoring; multi-channel profiles; regularized state space model

1. Introduction

With the current advanced sensing technologies, large amounts of in-process data become available. For example, Figure 1 shows signals collected by multiple sensors from a diffusion process in semiconductor manufacturing and a pipe-casing tightening process. Each sensor collects real-time values of one process variable such as voltage, temperature and torque. These data, commonly referred to as multi-channel profiles, provide a wealth of system state information that can be used for in-process monitoring and quality control.

So far, many studies have been conducted for multi-channel profile modeling and monitoring. For linear profiles with explanatory variables, multivariate multiple linear regression model has been widely adopted (Noorossana et al. 2010; Zou, Ning, and Tsung 2012; Eyvazian et al. 2011), where the functional relationship between responses and predictors is monitored. These works assumed the error terms either within or between profiles are independent. To better account for the within-profile correlation (WPC) inside each single profile, Zhang et al. (2014) adopted a Gaussian process, and Khedmati and Niaki (2016) imposed an autoregressive model on the error term. However, they have limited power to describe between-profile correlation (BPC). To address this issue, dimension reduction (feature extraction) techniques are generally applied, and the extracted low-dimensional features are used for monitoring. For example, Nomikos and MacGregor (1995) proposed a vectorized principal component analysis (VPCA) by combining multi-channel profiles into a long univariate profile for feature extraction. Clearly, this approach loses certain detection power since VPCA breaks the original BPC structure. To solve this problem, other PCA-based methods were proposed. Examples include uncorrelated multi-linear PCA (UMPCA) (Paynabar, Jin, and Pacella

CONTACT Chen Zhang zhangchen01@tsinghua.edu.cn Room 602, Shunde Building, Department of Industrial Engineering, Tsinghua University, Beijing 100084, P.R. China.

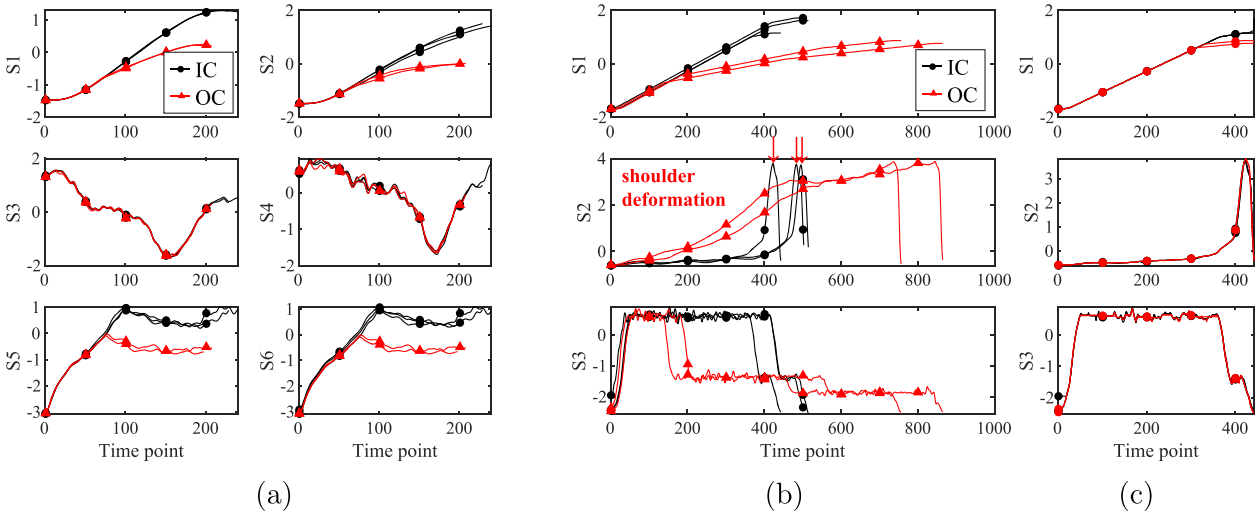


Figure 1. Multi-channel profiles (after normalization) from two advanced manufacturing processes: (a) The six selected sensor profiles (bottom voltage, top voltage, temperature of pipe 1, temperature of pipe 2, pressure of cylinder 1, and pressure of cylinder 2) over three in-control (IC) samples and two out-of-control (OC) samples from the diffusion process in semiconductor manufacturing; (b) The three-channel profiles (number of turns, torque signal, and rotating speed) over three IC samples and two OC samples from the pipe-casing tightening process before alignment and (c) after time-warping.

2013), multi-linear PCA (MPCA) (Grasso, Colosimo, and Pacella 2014), and multi-channel functional PCA (MFPCA) (Paynabar, Zou, and Qiu 2016).

All the above methods assumed multi-channel profiles are strongly correlated. However, this assumption may violate the reality, especially if the number of profile channels is large. As shown in Figure 1(a), different profiles are actually cluster-correlated. Profiles within a cluster (e.g., S1 and S2, S3 and S4, S5 and S6) have similar patterns and are correlated, while profiles from different clusters (e.g., S1 and S3) have quite different features and are almost uncorrelated. This is the so-called clustered BPC, which has recently received increasing attention in the literature. For instance, Zhang et al. (2018b) proposed a sparse multi-channel functional PCA (SMFPCA) for weakly-correlated multi-channel profile monitoring. Li et al. (2020) further extended it to tensor data and proposed a graph-based tensor completion model.

Though these existing methods have shown promising results in multi-channel profile monitoring, there are still some challenges to be addressed. First, the monitoring efficiency is sacrificed since these detection algorithms cannot be implemented until the whole profile has been collected. When the profile already shows significantly irregular pattern in early sensing points, it is uneconomical to wait for monitoring until the entire profile is obtained, especially for processes with high production costs and limited transmission resources. Second, the above methods require that profiles of different samples be synchronized with equal number of sensing points and have the same features located at the same sensing point, i.e., have the same time length and feature location. However, these assumptions cannot be satisfied in general since different samples are usually misaligned due to system inherent fluctuations. We demonstrate this point using data from the pipe-casing tightening process shown in Figure 1(b). Three sensors (S1-S3) located on a backup tong are used to collect the number of turns, torque signal, and rotating speed respectively during this process. Since the initial states of different pipes vary according to their different pre-tightening results, different pipes take different amounts of time to complete the tightening process and hence have unequal time lengths. Also, the local shoulder deformation occurs at different time points, as the observable peaks show. Consequently, for all current monitoring algorithms, time-warping method was commonly adopted to align profiles as a preprocessing step in functional data analysis (Ramsay and Silverman 2005; Zhang et al. 2021; Paynabar, Zou, and Qiu 2016; Zhang et al. 2018a). However, this operation forces OC profiles to resemble the IC reference one, which may weaken the OC patterns, especially when the misalignment problem is significant (Grasso et al. 2016), as shown for the S2 profile in Figure 1(c). This fault-weakening effect is more significant for sparse shifts, which can be demonstrated by the poor detection power of the PCA-based methods for spike shift in our numerical studies.

A more reasonable method is to model the profile in a dynamic view. Instead of considering the whole profile as a functional curve, a dynamic model treats the values of a profile as observations sequentially generated by a

dynamic system. Hence, we aim to model the dynamic evolution mechanism behind the system rather than the static features of the curve. As such, profiles of different samples can vary in time length, and features can be unsynchronized with time variations. More importantly, it gives the feasibility of detecting anomalies inside the profile. We highlight this idea as "in-profile monitoring" (INPOM), which is the focus of this article.

So far to our best knowledge, the only existing works able to deal with INPOM are Qiu and Xiang (2014) and Qiu, Zi, and Zou (2018), where each sample was treated as a separate process whose dynamic longitudinal behavior was of interest. Their application scenario is different from our online profile monitoring scenario where samples are coming in a sequential way. That is, once a sample goes OC at a certain time point, its following time points and all the time points of subsequent samples will always be OC. Also, both works only consider single profiles, and cannot be trivially extended to multi-channel profile modeling. Another related research field of INPOM is time series monitoring, where sequential observations of a certain sample are collected as a time series from a dynamic system. Among all methods, state space model (SSM) has been generally applied for process modeling and monitoring (Barber 2012). SSM supposes the observed data are functions of certain unobserved (latent) states, which are autocorrelated and evolve over time. This structure is very general, and includes autoregressive moving average (ARMA) model as a special case (Gevers and Wertz 1984). Furthermore, this latent structure is particularly suitable for manufacturing systems, where the evolution of measured variables is generally driven by several underlying physical or mechanical characteristics. For example, in the diffusion process shown in Figure 1(a), electronic sensors (S1 and S2) are jointly influenced by circuit power, while temperature sensors located adjacently (S3 and S4) are both linearly depended on thermal energy. Thermal energy and circuit power can then be regarded as two latent states in this case. Out of this concern, SSM is naturally able to handle the clustered BPC by modeling different profiles as generations from different latent states. However, most existing works focused on traditional multivariate process monitoring, and assumed data of sequential samples follow an SSM (Shi 2006; Xiang and Tsung 2008; Liu, Shi, and Hu 2009). Yet in our case, the observations of different sensing time points inside each multi-channel profile evolve dynamically and can be described by an SSM, while different samples are independent.

To address the limitations of existing works, this article proposes an SSM-based framework to dynamically model and monitor multi-channel profile data, with especial application in manufacturing system. To account for the clustered BPC among different profiles, SSM is extended to a regularized SSM (RSSM) by imposing a graph Laplacian regularization on the observation matrix of SSM. An l_1 regularization is also imposed on the transition matrix of SSM to avoid overfitting. By treating the above regularizations as prior information, the model parameters can be efficiently learned via Bayesian inference, where expectation maximization (EM) algorithm is incorporated for posterior maximization. Built upon this, a T^2 monitoring statistic based on one-step-ahead prediction error is constructed for INPOM.

To understand the superiority of RSSM-INPOM framework, we would like to further provide some insights of it. First, though profile data of each sample look like smooth curve(s), they are in essence generated sequentially over time according to the dynamic evolution mechanism of system. This is in accordance with the modeling mechanism of SSM, where profiles are characterized by a set of dynamic temporal bases stacked by the latent state variables over time. This brings the merit that the bases of SSM are described in a dynamic way and can be different for different samples, which is extremely effective for misaligned samples since it avoids time-warping operation that may weaken the anomaly patterns. More importantly, since profile data are treated as observations sequentially generated from a dynamic system, SSM can detect anomalies inside the profile as long as observations up to the current time point have provided enough evidence. This advantage is more significant for processes that need long time to complete. Finally, by enforcing certain structure regularizations on the parameters of SSM, our extended RSSM makes it more suitable for multi-channel profile modeling in advance manufacturing.

The remainder of this article is organized as follows. Section 2 introduces our proposed method in detail. In Section 3, simulation studies are demonstrated to investigate the performance of the RSSM-INPOM framework by comparing with six benchmarks. Two real case studies are conducted in Section 4. Finally, Section 5 concludes this article with several remarks.

2. Methodology

In this section, a RSSM-INPOM scheme is developed for Phase II monitoring of multi-channel profiles with clustered BPC. We first introduce the model formulation of RSSM in Section 2.1. Then in Section 2.2,

maximum a posteriori (MAP) estimation is adopted for parameter learning. In Section 2.3, one-step-ahead prediction errors are further used to construct an INPOM control chart. Finally, Section 2.3, discusses the practical guidelines for designing and implementing our RSSM-INPOM framework.

2.1. Regularized sate space model (RSSM)

Consider a system with p-dimensional (i.e., p-channel) profiles. For a particular sample i, its observation recorded at sensing time point t can be denoted as a p-dimensional vector $\boldsymbol{y}_t^{(i)} = [y_{1t}^{(i)}, ..., y_{pt}^{(i)}]'$. Then profile data of sample i is $\boldsymbol{Y}^{(i)} = \{\boldsymbol{y}_1^{(i)}, ..., \boldsymbol{y}_t^{(i)}, ..., \boldsymbol{y}_{T_i}^{(i)}\}$, where T_i is the profile length and can be different for different samples. We further assume q-dimensional latent state variable $\boldsymbol{x}_t^{(i)} = [x_{1t}^i, ..., x_{qt}^i]'$ and $\boldsymbol{X}^{(i)} = \{\boldsymbol{x}_1^{(i)}, ..., \boldsymbol{x}_t^{(i)}, ..., \boldsymbol{x}_{T_i}^{(i)}\}$. We consider $\boldsymbol{y}_t^{(i)}$ as a linear combination of $\boldsymbol{x}_t^{(i)}$ masked by an output noise $\boldsymbol{v}_t^{(i)}$ which follows a multivariate Gaussian distribution with mean $\boldsymbol{0}$ and covariance matrix $\boldsymbol{R}$, i.e., $\boldsymbol{v}_t^{(i)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R})$, then

$$\boldsymbol{y}_t^{(i)} = \boldsymbol{C}\boldsymbol{x}_t^{(i)} + \boldsymbol{v}_t^{(i)},\ i = 1, ..., N; t = 1, ..., T_i, \quad [1]$$

where $\boldsymbol{C}$ is the $p \times q$ observation matrix that describes the BPC structure and N is the number of samples. The WPC over time can be depicted by the evolution of the state variable as

$$\boldsymbol{x}_{t+1}^{(i)} = \boldsymbol{A}\boldsymbol{x}_t^{(i)} + \boldsymbol{\omega}_t^{(i)}, i = 1, ..., N; t = 1, ..., T_i - 1, \quad [2]$$

where $\boldsymbol{A}$ is the $q \times q$ transition matrix, $\boldsymbol{\omega}_t^{(i)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q})$ is the Gaussian distributed state noise, and the initial state variable is assumed to follow a multivariate Gaussian distribution as $\boldsymbol{x}_1^{(i)} \sim \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{V}_1)$. Eqs. [1] and [2] are known as the observation equation and the transition equation of SSM respectively. It is the probabilistic $\boldsymbol{x}_1^{(i)}, \boldsymbol{v}_t^{(i)}$, and $\boldsymbol{\omega}_t^{(i)}$ that allow different samples vary in time length and feature location. Let $\boldsymbol{\Theta} = \{\boldsymbol{A}, \boldsymbol{C}, \boldsymbol{Q}, \boldsymbol{R}, \boldsymbol{\mu}_1, \boldsymbol{V}_1\}$ denote the complete set of the parameters of SSM. It is noted that we are interested in single-stage manufacturing processes, and hence we assume the model parameters are time invariant. Different samples are independent and characterized by the same dynamic model.

To better capture the BPC of multi-channel or especially high-dimensional profiles, we now enforce certain structure regularizations on the model parameters of SSM considering the following two features demonstrated by real data in manufacturing.

Feature 1. *$\boldsymbol{C} = [\boldsymbol{c}_{:1}, ..., \boldsymbol{c}_{:q}] = [\boldsymbol{c}'_{1:}, ..., \boldsymbol{c}'_{p:}]'$ has a clustered structure. Consider that in practice, different profiles of a sample are jointly influenced by different types of system characteristics. Profiles (sensors j_1 and j_2) that depend on the same system characteristics have strong BPC, and thus their corresponding rows of $\boldsymbol{C}$, i.e., $\boldsymbol{c}_{j_1:}$ and $\boldsymbol{c}_{j_2:}$, tend to be similar. On the contrary, profiles (sensors) that depend on different system characteristics have weak BPC, and thus their corresponding rows of $\boldsymbol{C}$ tend to be different. In other words, clustered BPC will result in a clustered $\boldsymbol{C}$.*

Suppose we have the cluster information of multi-channel profiles through a predefined similarity matrix $\boldsymbol{W} = [w_{j_1 j_2}]$, where larger $w_{j_1 j_2}$ indicates that sensor j_1 and sensor j_2 are more correlated. For example, in the previous example of the diffusion process, $w_{j_1 j_2}$ of two closely located temperature sensors would be large. Generally, $\boldsymbol{W}$ can be set according to domain knowledge. If there is no preference on $\boldsymbol{W}$, we can simply set it according to profile similarity in a data-driven way, e.g., $w_{j_1 j_2} = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T_i} |y_{j_1 t}^{(i)} - y_{j_2 t}^{(i)}|^2 / T_i\right)$. This cluster information can be used to derive a desirable $\boldsymbol{C}$. In particular, we assume each column of the observation matrix $\boldsymbol{c}_{:k} (k = 1...q)$ independently has a Gaussian distribution prior (Zhou et al. 2012) as $p_0(\boldsymbol{C}) = \prod_{k=1}^{q} p_0(\boldsymbol{c}_{:k})$ where $\boldsymbol{c}_{:k} \sim \mathcal{N}(\boldsymbol{0}, (\lambda_1 \boldsymbol{L})^{-1})$, i.e.,

$$p_0\left(\boldsymbol{c}_{:k} | \boldsymbol{0}, (\lambda_1 \boldsymbol{L})^{-1}\right) = \frac{|\lambda_1 \boldsymbol{L}|^{\frac{1}{2}}}{(2\pi)^{\frac{p}{2}}} \exp\left(-\frac{\lambda_1 \boldsymbol{c}'_{:k} \boldsymbol{L} \boldsymbol{c}_{:k}}{2}\right), k = 1...q. \quad [3]$$

Here λ_1 is the tuning parameter. We define $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{D} = diag(d_1, ..., d_p)$ with $d_{j_1} = \sum_{j_2=1}^{p} w_{j_1 j_2}, j_1 = 1, ..., p$. With this design, the prior leads to a shrinkage on $\lambda_1 tr(\boldsymbol{C}'\boldsymbol{L}\boldsymbol{C}) = \lambda_1 \sum_{j_1=1}^{p}\sum_{j_2=1}^{p} \|\boldsymbol{c}_{j_1:} - \boldsymbol{c}_{j_2:}\|_2^2 w_{j_1 j_2}$ where $\dot{c}$ denotes the trace of a matrix. It indicates that for similar sensors j_1 and j_2 with larger $w_{j_1 j_2}$, their corresponding rows $\boldsymbol{c}_{j_1:}$ and $\boldsymbol{c}_{j_2:}$ on $\boldsymbol{X}^{(i)}$ should be similar. As such, it will lead to a clustered structure of $\boldsymbol{C}$. In fact, this is the graph Laplacian regularization commonly used for graph-based modeling (Ando and Zhang 2007). By defining different sensors as nodes, $\boldsymbol{W}$ as the adjacency matrix and $\boldsymbol{D}$ as the degree matrix, $\boldsymbol{L}$ is the graph Laplacian matrix. This penalty encourages smoothness of $\boldsymbol{c}_{j:}$ over the graph.

Feature 2. *$\boldsymbol{A} = [a_{k_1 k_2}]$ is a sparse matrix due to the following two reasons: (1) Different system characteristics identified as different states are likely to be sparsely correlated. In the previous example of diffusion process, thermal energy (state k_1) and circuit power (state k_2)*

do not correlate with each other and consequently their associated entry in $\boldsymbol{A}$*, i.e.,* $a_{k_1k_2}$*, is more likely to be zero; (2) A sparse* $\boldsymbol{A}$ *can also help remove redundant states, consequently avoid overfitting and guarantee model identifiability.*

To achieve sparsity on the transition matrix, we impose a Laplace prior independently on each element of $\boldsymbol{A}$, which is equivalent to the l_1 regularization (Liu and Hauskrecht 2013), i.e., $p_0(\boldsymbol{A}) = \prod_{k_1=1}^{q}\prod_{k_2=1}^{q} p_0(a_{k_1k_2})$, where $a_{k_1k_2} \sim Laplace\left(0, \frac{2}{\lambda_2}\right)$, i.e.,

$$p_0\left(a_{k_1k_2}|0, \frac{2}{\lambda_2}\right) = \frac{\lambda_2}{4}\exp\left(-\frac{\lambda_2}{2}|a_{k_1k_2}|\right), \quad k_1 = 1, ..., q; k_2 = 1, ..., q. \tag{4}$$

This prior distribution tends to shrink the magnitude of $\lambda_2\sum_{k_1=1}^{q}\sum_{k_2=1}^{q}|a_{k_1k_2}| = \lambda_2||\boldsymbol{A}||_1$, where λ_2 is the tuning parameter. As a result, it will induce zero entries in $\boldsymbol{A}$. So far, the basic two-layer representation of SSM (Eqs. [1] and [2]) and the two prior distributions (Eqs. [3] and [4]) formulate our proposed RSSM in a probabilistic perspective.

However, direct maximization of the posterior distribution $p(\boldsymbol{\Theta}|\boldsymbol{Y}^{(1:N)})$ is extremely difficult since its closed-form solution does not exist. Alternatively, the augmented complete posterior $p(\boldsymbol{\Theta}|\boldsymbol{X}^{(1:N)}, \boldsymbol{Y}^{(1:N)})$ has a simpler analytical expression, yet integrating out $\boldsymbol{X}^{(1:N)}$ proves to be a difficult task. To solve this, EM algorithm can be adopted. In particular, in the E step of the r^{th} iteration, the expectation of the augmented complete posterior is calculated using the observed data $\boldsymbol{Y}^{(1:N)}$ and the current best estimation $\boldsymbol{\Theta}^{r-1}$. In the M step, the parameters are updated as $\boldsymbol{\Theta}^r$ by maximizing the expected augmented complete posterior. These two steps iterate until convergence and obtain $\hat{\boldsymbol{\Theta}}$ by generating a sequence of parameters $\{\boldsymbol{\Theta}^r\}$ from an initial estimation $\boldsymbol{\Theta}^0$. The appealing property of EM algorithm is that the marginal distribution increases monotonically with the iteration, i.e., $p(\boldsymbol{Y}^{(1:N)}|\boldsymbol{\Theta}^r) \geq p(\boldsymbol{Y}^{(1:N)}|\boldsymbol{\Theta}^{r-1})$ given mild regularity conditions (Wu 1983). The specific E and M steps for the r^{th} iteration in our case are shown as follows:

E Step

Evaluate the conditional expectation of the log augmented complete posterior:

$$\begin{aligned}\mathcal{Q}(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{r-1}) &= E[\log(p(\boldsymbol{\Theta}|\boldsymbol{X}^{(1:N)}, \boldsymbol{Y}^{(1:N)}))|\boldsymbol{Y}^{(1:N)}, \boldsymbol{\Theta}^{r-1}] \\ &\propto \underbrace{E[\log(p(\boldsymbol{X}^{(1:N)}, \boldsymbol{Y}^{(1:N)}|\boldsymbol{\Theta}))|Y^{(1:N)}, \boldsymbol{\Theta}^{r-1}]}_{\text{expected log likelihood}} + \underbrace{[-\frac{\lambda_1}{2}\text{tr}(\boldsymbol{C}'\boldsymbol{L}\boldsymbol{C}) - \frac{\lambda_2}{2}\|\boldsymbol{A}\|_1]}_{\text{prior regularizations}},\end{aligned} \tag{5}$$

2.2. Model inference based on EM-MAP algorithm

Since $\boldsymbol{A}$ and $\boldsymbol{C}$ have prior distributions, we can use the MAP framework for parameter estimation. Our objective is to compute the MAP estimation of $\boldsymbol{\Theta}$ given N historical IC samples, i.e., $\hat{\boldsymbol{\Theta}} \triangleq \text{argmax}_{\boldsymbol{\Theta}} p(\boldsymbol{\Theta}|\boldsymbol{Y}^{(1:N)})$. where the last two terms are the prior regularizations and the first term is the expected log likelihood which can be derived following Ghahramani and Hinton (1996). Specifically, due to the Markovian structure of SSM, we have

$$\begin{aligned}&E[\log(p(\boldsymbol{X}^{(1:N)}, \boldsymbol{Y}^{(1:N)}|\boldsymbol{\Theta}))|Y^{(1:N)}, \boldsymbol{\Theta}^{r-1}] \\ &\propto -\frac{1}{2}\left\{\begin{array}{l}\sum_{i=1}^{N}\sum_{t=1}^{T_i}E[(\boldsymbol{y}_t^{(i)} - \boldsymbol{C}\boldsymbol{x}_t^{(i)})'\boldsymbol{R}^{-1}(\boldsymbol{y}_t^{(i)} - \boldsymbol{C}\boldsymbol{x}_t^{(i)})|Y^{(i)}, \boldsymbol{\Theta}^{r-1}] + \sum_{i=1}^{N}T_i\log|\boldsymbol{R}| \\ +\sum_{i=1}^{N}E[(\boldsymbol{x}_1^{(i)} - \boldsymbol{\mu}_1)'\boldsymbol{V}_1^{-1}(\boldsymbol{x}_1^{(i)} - \boldsymbol{\mu}_1)|Y^{(i)}, \boldsymbol{\Theta}^{r-1}] + N\log|\boldsymbol{V}_1| \\ +\sum_{i=1}^{N}\sum_{t=2}^{T_i}E[(\boldsymbol{x}_t^{(i)} - \boldsymbol{A}\boldsymbol{x}_{t-1}^{(i)})'\boldsymbol{Q}^{-1}(\boldsymbol{x}_t^{(i)} - \boldsymbol{A}\boldsymbol{x}_{t-1}^{(i)})|Y^{(i)}, \boldsymbol{\Theta}^{r-1}] + \sum_{i=1}^{N}(T_i - 1)\log|\boldsymbol{Q}|\end{array}\right\}.\end{aligned}$$

By polynomial expansion of the above equation, Eq. [5] can be explicitly expressed as a linear function of the following three terms (For notation convenience, we omit the sample superscript i here):

$$\begin{aligned}\hat{\boldsymbol{x}}_t &\equiv E\left[\boldsymbol{x}_t|\boldsymbol{y}_1,...,\boldsymbol{y}_T,\boldsymbol{\Theta}^{r-1}\right] \equiv \boldsymbol{x}_{t|T},\\ \boldsymbol{P}_t &\equiv E\left[\boldsymbol{x}_t\boldsymbol{x}_t'|\boldsymbol{y}_1,...,\boldsymbol{y}_T,\boldsymbol{\Theta}^{r-1}\right] \equiv \boldsymbol{V}_{t|T}+\boldsymbol{x}_{t|T}\boldsymbol{x}_{t|T}',\\ \boldsymbol{P}_{t,t-1} &\equiv E\left[\boldsymbol{x}_t\boldsymbol{x}_{t-1}'|\boldsymbol{y}_1,...,\boldsymbol{y}_T,\boldsymbol{\Theta}^{r-1}\right] \equiv \boldsymbol{V}_{t,t-1|T}+\boldsymbol{x}_{t|T}\boldsymbol{x}_{t-1|T},\end{aligned} \quad [6]$$

where $\boldsymbol{x}_{t|T}$ is the expectation of $\boldsymbol{x}_t$, $\boldsymbol{V}_{t|T}$ is the variance of $\boldsymbol{x}_t$, and $\boldsymbol{V}_{t,t-1|T}$ is the covariance of $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$, given all the observations $\boldsymbol{y}_1,...,\boldsymbol{y}_T$ in the current profile. These variables can be obtained via Kalman filter (KF) and Kalman smoother (KS). Specifically, KF is a forward recursion which estimates $\boldsymbol{x}_{t|t},\boldsymbol{V}_{t|t},\boldsymbol{V}_{t,t-1|t}$ using the past t observations $\boldsymbol{y}_1,...,\boldsymbol{y}_t$, i.e.,

$$\boldsymbol{x}_{t|t-1} = \boldsymbol{A}\boldsymbol{x}_{t-1|t-1}, \quad [7a]$$

$$\boldsymbol{V}_{t|t-1} = \boldsymbol{A}\boldsymbol{V}_{t-1|t-1}\boldsymbol{A}'+\boldsymbol{Q}, \quad [7b]$$

$$\boldsymbol{K}_t = \boldsymbol{V}_{t|t-1}\boldsymbol{C}'(\boldsymbol{C}\boldsymbol{V}_{t|t-1}\boldsymbol{C}'+\boldsymbol{R})^{-1}, \quad [7c]$$

$$\boldsymbol{x}_{t|t} = \boldsymbol{x}_{t|t-1}+\boldsymbol{K}_t(\boldsymbol{y}_t-\boldsymbol{C}\boldsymbol{x}_{t|t-1}), \quad [7d]$$

$$\boldsymbol{V}_{t|t} = \boldsymbol{V}_{t|t-1}-\boldsymbol{K}_t\boldsymbol{C}\boldsymbol{V}_{t|t-1}, \quad [7e]$$

where the initial expectation and covariance matrix are $\boldsymbol{x}_{1|0}=\boldsymbol{\mu}_1$ and $\boldsymbol{V}_{1|0}=\boldsymbol{V}_1$. Based on the results of KF, KS is a backward recursion which modifies the estimates of KF using the future observations, i.e.,

$$\boldsymbol{J}_{t-1} = \boldsymbol{V}_{t-1|t-1}\boldsymbol{A}'(\boldsymbol{V}_{t|t-1})^{-1}, \quad [8a]$$

$$\boldsymbol{x}_{t-1|T} = \boldsymbol{x}_{t-1|t-1}+\boldsymbol{J}_{t-1}(\boldsymbol{x}_{t|T}-\boldsymbol{A}\boldsymbol{x}_{t-1|t-1}), \quad [8b]$$

$$\boldsymbol{V}_{t-1|T} = \boldsymbol{V}_{t-1|t-1}+\boldsymbol{J}_{t-1}(\boldsymbol{V}_{t|T}-\boldsymbol{V}_{t|t-1})\boldsymbol{J}_{t-1}', \quad [8c]$$

$$\begin{aligned}\boldsymbol{V}_{t-1,t-2|T} = {}& \boldsymbol{V}_{t-1|t-1}\boldsymbol{J}_{t-2}'\\ &+\boldsymbol{J}_{t-1}(\boldsymbol{V}_{t,t-1|T}-\boldsymbol{A}\boldsymbol{V}_{t-1|t-1})\boldsymbol{J}_{t-2}',\end{aligned} \quad [8d]$$

where the initial covariance of $\boldsymbol{x}_T$ and $\boldsymbol{x}_{T-1}$ is $\boldsymbol{V}_{T,T-1|T}=(\boldsymbol{I}-\boldsymbol{K}_T\boldsymbol{C})\boldsymbol{A}\boldsymbol{V}_{T-1|T-1}$. After these two steps, $\hat{\boldsymbol{x}}_t,\boldsymbol{P}_t,\boldsymbol{P}_{t,t-1}$ can be obtained by substituting the computed $\boldsymbol{x}_{t|T},\boldsymbol{V}_{t|T},\boldsymbol{V}_{t,t-1|T}$ into Eq. [6], followed by the expectation of the log augmented complete posterior computed via Eq. [5]. It should be noted that KF and KS can only handle stationary profile data, i.e., $\boldsymbol{A}$ has all stationary roots. Fortunately, this can be generally satisfied for profile data from manufacturing systems since machines are usually stable when in-control. When profiles are nonstationary, it is still solvable with a trivially extended form of KF, e.g., the diffuse Kalman filter (Jong 1991).

M Step

Update each parameter separately:

$$\boldsymbol{\Theta}^r = \arg\max_{\boldsymbol{\Theta}} \mathcal{Q}(\boldsymbol{\Theta},\boldsymbol{\Theta}^{r-1}). \quad [9]$$

1. Updating $\boldsymbol{A}$: Take out all the relevant terms w.r.t $\boldsymbol{A}$ in $Q(\boldsymbol{\Theta},\boldsymbol{\Theta}^{r-1})$, then the optimization problem is equivalent to $\boldsymbol{A}^r=\text{argmin}_{\boldsymbol{A}} f(\boldsymbol{A})$, where

$$f(\boldsymbol{A}) = \underbrace{\frac{\lambda_2}{2}\|\boldsymbol{A}\|_1}_{f_1(\boldsymbol{A})} + \underbrace{\frac{1}{2}\text{tr}\left\{\boldsymbol{Q}^{-1}\sum_{i=1}^{N}\sum_{t=2}^{T_i}\left[\boldsymbol{P}_t^{(i)}-\boldsymbol{P}_{t,t-1}^{(i)}\boldsymbol{A}'-\boldsymbol{A}\boldsymbol{P}_{t,t-1}^{(i)'}+\boldsymbol{A}\boldsymbol{P}_{t-1}^{(i)}\boldsymbol{A}'\right]\right\}}_{f_2(\boldsymbol{A})}.$$

It can be decomposed into a non-differentiable part $f_1(\boldsymbol{A})$ and a differentiable part $f_2(\boldsymbol{A})$. We can then apply the proximal gradient descent method with iterative soft-thresholding algorithm (ISTA) to minimize $f(\boldsymbol{A})$. The update rule in the m^{th} iteration of the gradient descent algorithm is

$$\begin{aligned}\boldsymbol{A}_m^r &= \mathcal{S}_{\lambda_2,\eta}\left[\boldsymbol{A}_{m-1}^r-\eta\nabla f_2(\boldsymbol{A}|\boldsymbol{A}=\boldsymbol{A}_{m-1}^r)\right]\\ &= \begin{cases}\boldsymbol{A}_{m-1}^r-\eta\nabla f_2(\boldsymbol{A}|\boldsymbol{A}=\boldsymbol{A}_{m-1}^r)-\lambda_2\eta & \text{if } \boldsymbol{A}_{m-1}^r-\eta\nabla f_2(\boldsymbol{A}|\boldsymbol{A}=\boldsymbol{A}_{m-1}^r)>\lambda_2\eta,\\ \boldsymbol{A}_{m-1}^r-\eta\nabla f_2(\boldsymbol{A}|\boldsymbol{A}=\boldsymbol{A}_{m-1}^r)+\lambda_2\eta & \text{if } \boldsymbol{A}_{m-1}^r-\eta\nabla f_2(\boldsymbol{A}|\boldsymbol{A}=\boldsymbol{A}_{m-1}^r)<-\lambda_2\eta,\\ 0 & \text{otherwise,}\end{cases}\end{aligned} \quad [10]$$

where $\boldsymbol{A}_0^r = \boldsymbol{A}^{r-1}$ is the initial value, and $\mathcal{S}_{\lambda_2,\eta}$ is a soft-thresholding function with a thresholding of $\lambda_2\eta$. η is the fixed step size set as $\eta = \frac{1}{\|\boldsymbol{Q}^{-1}\|_F \cdot \|\sum_{i=1}^{N}\sum_{t=2}^{T_i}\boldsymbol{P}_{t-1}^{(i)}\|_F}$ which proves to guarantee a fast convergence rate (Liu and Hauskrecht 2013). $\nabla f_2(\boldsymbol{A}|\boldsymbol{A} = \boldsymbol{A}_{m-1}^r) = \sum_{i=1}^{N}\sum_{t=2}^{T_i}\boldsymbol{Q}^{-1}\boldsymbol{A}_{m-1}^r\boldsymbol{P}_{t-1}^{(i)} - \sum_{i=1}^{N}\sum_{t=2}^{T_i}\boldsymbol{Q}^{-1}\boldsymbol{P}_{t,t-1}^{(i)}$ is the gradient of the differentiable part $f_2(\boldsymbol{A})$ at value $\boldsymbol{A}_{m-1}^r$. Suppose this loop converges in the M^{th} iteration, then we have $\boldsymbol{A}^r = \boldsymbol{A}_M^r$.

2. Updating $\boldsymbol{C}$: Similarly, the original optimization problem on $\boldsymbol{C}$ can be converted to $\boldsymbol{C}^r = \text{argmin}_{\boldsymbol{C}} g(\boldsymbol{C})$ where

$$g(\boldsymbol{C}) = \frac{\lambda_1}{2}tr(\boldsymbol{C}'\boldsymbol{L}\boldsymbol{C}) + \frac{1}{2}tr\left\{\boldsymbol{R}^{-1}\sum_{i=1}^{N}\sum_{t=1}^{T_i}\left[\boldsymbol{y}_t^{(i)}\boldsymbol{y}_t^{(i)'} - \boldsymbol{C}\hat{\boldsymbol{x}}_t^{(i)}\boldsymbol{y}_t^{(i)'} - \boldsymbol{y}_t^{(i)}\hat{\boldsymbol{x}}_t^{(i)'}\boldsymbol{C}' + \boldsymbol{C}\boldsymbol{P}_t^{(i)}\boldsymbol{C}'\right]\right\}.$$

Set $\nabla g(\boldsymbol{C}|\boldsymbol{C} = \boldsymbol{C}^r)$ to zero. We have $\lambda_1\boldsymbol{R}\boldsymbol{L}\boldsymbol{C}^r + \boldsymbol{C}^r\sum_{i=1}^{N}\sum_{t=1}^{T_i}\boldsymbol{P}_t^{(i)} = \sum_{i=1}^{N}\sum_{t=1}^{T_i}\boldsymbol{y}_t^{(i)}\hat{\boldsymbol{x}}_t^{(i)'}$, which is a Sylvester equation. We adopt the Bartels-Stewart algorithm to get the following closed-form solution (Bartels and Stewart 1972):

$$vec(\boldsymbol{C}^r) = \left(\boldsymbol{I}_q \otimes \lambda_1\boldsymbol{R}\boldsymbol{L} + \sum_{i=1}^{N}\sum_{t=1}^{T_i}\boldsymbol{P}_t^{(i)} \otimes \boldsymbol{I}_p\right)^{-1} \times vec\left(\sum_{i=1}^{N}\sum_{t=1}^{T_i}\boldsymbol{y}_t^{(i)}\hat{\boldsymbol{x}}_t^{(i)'}\right), \quad [11]$$

where $\boldsymbol{I}_p$ and $\boldsymbol{I}_q$ represent the identity matrices with dimension p and q, and $\otimes$ is the Kronecker product. When p or q is large, conjugate gradient (CG) algorithm in Rao et al. (2015) can be alternatively adopted to avoid inverse operations.

3. Updating $\boldsymbol{Q}$, $\boldsymbol{R}$, $\boldsymbol{\mu}_1$ and $\boldsymbol{V}_1$: All these parameters have closed-form solutions and can be solved via the following.

$$\boldsymbol{Q}^r = \frac{1}{\sum_{i=1}^{N}T_i - N}\sum_{i=1}^{N}\sum_{t=2}^{T_i}\left[\boldsymbol{P}_t^{(i)} - \boldsymbol{P}_{t,t-1}^{(i)}\boldsymbol{A}^{r\prime} - \boldsymbol{A}^r\boldsymbol{P}_{t,t-1}^{(i)'} + \boldsymbol{A}^r\boldsymbol{P}_{t-1}^{(i)}\boldsymbol{A}^{r\prime}\right], \quad [12]$$

$$\boldsymbol{R}^r = \frac{1}{\sum_{i=1}^{N}T_i}\sum_{i=1}^{N}\sum_{t=2}^{T_i}\left[\boldsymbol{y}_t^{(i)}\boldsymbol{y}_t^{(i)'} - \boldsymbol{C}^r\hat{\boldsymbol{x}}_t^{(i)}\boldsymbol{y}_t^{(i)'} - \boldsymbol{y}_t^{(i)}\hat{\boldsymbol{x}}_t^{(i)'}\boldsymbol{C}^{r\prime} + \boldsymbol{C}^r\boldsymbol{P}_t^{(i)}\boldsymbol{C}^{r\prime}\right], \quad [13]$$

$$\boldsymbol{\mu}_1^r = \frac{1}{N}\sum_{i=1}^{N}\hat{\boldsymbol{x}}_1^{(i)}, \quad [14]$$

$$\boldsymbol{V}_1^r = \frac{1}{N}\sum_{i=1}^{N}\left[\boldsymbol{P}_1^{(i)} - \hat{\boldsymbol{x}}_1^{(i)}\hat{\boldsymbol{x}}_1^{(i)'}\right]. \quad [15]$$

Algorithm 1. EM-MAP algorithm for model inference

Input: $\boldsymbol{Y}^{(i)}, i = 1, ..., N$
Result: Estimated $\hat{\Theta} = \{\hat{\boldsymbol{A}}, \hat{\boldsymbol{C}}, \hat{\boldsymbol{Q}}, \hat{\boldsymbol{R}}, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{V}}_1\}$
Initialization For $r = 0$, set $\boldsymbol{\Theta}^0$ by subspace identification
Estimation *(by EM algorithm)*
while $|Q(\boldsymbol{\Theta}^r, \boldsymbol{\Theta}^{r-1}) - Q(\boldsymbol{\Theta}^{r-1}, \boldsymbol{\Theta}^{r-2})| \geq 10^{-4} \times |Q(\boldsymbol{\Theta}^{r-1}, \boldsymbol{\Theta}^{r-2})|$ **do**
  Set $r = r + 1$
  E step:
  for $i = 1, \ldots, N$ **do**
    Update $\boldsymbol{x}_{t|T_i}^{(i)}, \boldsymbol{V}_{t|T_i}^{(i)}, \boldsymbol{V}_{t,t-1|T_i}^{(i)}$ using the current $\boldsymbol{\Theta}^{r-1}$ via Eqs. [7] and [8]
    Calculate $\hat{\boldsymbol{x}}_t^{(i)}, \boldsymbol{P}_t^{(i)}, \boldsymbol{P}_{t,t-1}^{(i)}$ via Eq. [6]
  end
  M step:
  while $\|\boldsymbol{A}_m^r - \boldsymbol{A}_{m-1}^r\|_F \geq 10^{-3} \times \|\boldsymbol{A}_{m-1}^r\|_F$ **do**
    Set $m = m + 1$
    Update $\boldsymbol{A}_m^r$ via Eq. [10]
  end
  Update $\boldsymbol{C}^r, \boldsymbol{Q}^r, \boldsymbol{R}^r, \boldsymbol{\mu}_1^r, \boldsymbol{V}_1^r$ via Eqs. [11] to [15]
  Calculate $Q(\boldsymbol{\Theta}^r, \boldsymbol{\Theta}^{r-1})$ via Eq. [5]
end

Denote the converged estimation results as $\hat{\Theta} = \{\hat{\boldsymbol{A}}, \hat{\boldsymbol{C}}, \hat{\boldsymbol{Q}}, \hat{\boldsymbol{R}}, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{V}}_1\}$. Considering that EM algorithm is sensitive to initial values, we implement the parameter initialization process via subspace identification, which is commonly used for model inference of SSM. Its details are shown in Appendix A. With this initialization algorithm, the convergence of EM can be guaranteed and the convergent speed is significantly expedited (Overschee and Moor 1996). Algorithm 1 summarizes the EM-MAP algorithm for model inference.

2.3. In-profile monitoring (INPOM)

Denote the estimated model from historical IC samples as $\boldsymbol{\Theta}_0$. We formulate the monitoring problem as an in-profile change-point model where anomalies occur since time point $\tau_0 + 1$ of sample $m_0 + 1$. Define the accumulative time-point index since the first time point of the first sample as u, then the accumulative change point is $u_0 = T_1 + ... + T_{m_0} + \tau_0$. The null hypothesis and the alternative hypothesis are established in the following form:

$$\begin{cases} \mathrm{H}_0 : \boldsymbol{\Theta} = \boldsymbol{\Theta}_0, & \text{for } u = 1, \ldots, u_0, \\ \mathrm{H}_1 : \boldsymbol{\Theta} \neq \boldsymbol{\Theta}_0, & \text{for } u = u_0 + 1, \ldots. \end{cases}$$

Notably, compared with the traditional change-point formulations of online monitoring, we have two levels of change points, τ_0 in time-point level, and m_0 in sample level. Consequently, here we define two criteria for evaluating the detection delay. Suppose the monitoring scheme triggers an OC alarm at u_1, i.e., time point τ_1 of sample m_1, we then define its OC profile run length (PRL_1) and OC time run length (TRL_1) as

$$\mathrm{PRL}_1 = m_1 - m_0, \quad \mathrm{TRL}_1 = u_1 - u_0, \tag{16}$$

where PRL_1 is the same as the OC run length of traditional control charts, which measures the detection delay in sample level. It is mainly used for comparison when shift magnitudes are small. When shift magnitudes are larger, RSSM-INPOM may successfully detect anomalies before the current OC sample m_1 is complete, and PRL_1 can no longer accurately measure the detection delay. Consequently, TRL_1 can be adopted to evaluate the detection delay of RSSM-INPOM in time-point level. For the traditional profile monitoring methods, TRL_1 is defined as $\sum_{i=m_0+1}^{m_1} T_i - \tau_0$.

Now we aim to construct a monitoring statistic for oncoming sequential data. When the observations $\boldsymbol{y}_1^{(i)}, \ldots, \boldsymbol{y}_{t-1}^{(i)}$ come sequentially for a certain online sample i, we predict its observation one step ahead by $\boldsymbol{y}_{t|t-1}^{(i)} = \boldsymbol{C}\boldsymbol{x}_{t|t-1}^{(i)}$, where $\boldsymbol{x}_{t|t-1}^{(i)} = \boldsymbol{A}\boldsymbol{x}_{t-1|t-1}^{(i)}$ can be obtained via KF in Eq. [7]. Furthermore, when the next observation at time point t has been collected, the corresponding one-step-ahead prediction error (OSPE) can be calculated by $\boldsymbol{e}_t^{(i)} = \boldsymbol{y}_t^{(i)} - \boldsymbol{y}_{t|t-1}^{(i)}$. OSPE reflects the untraceable signal of $\boldsymbol{y}_t^{(i)}$, which includes system inherent fluctuations and anomaly information that cannot be described by the IC model with $\boldsymbol{\Theta}_0$. As such, we can transform monitoring $\boldsymbol{\Theta}$ into monitoring OSPE (Xiang and Tsung 2008).

Under H_0, we can derive that $\boldsymbol{e}_t^{(i)}$ follows a p-dimensional multivariate Gaussian distribution with mean $\boldsymbol{0}$ and covariance matrix $\boldsymbol{S}_t^{(i)} = \boldsymbol{C}\boldsymbol{V}_{t|t-1}^{(i)}\boldsymbol{C}' + \boldsymbol{R}$ (Durbin and Koopman 2012). Therefore, we construct the monitoring statistic based on the Hotelling T^2 test of $\boldsymbol{e}_t^{(i)}$. To handle anomalies with small shift, we incorporate the monitoring scheme with exponentially weighted moving average (EWMA) technique. Different from existing works for profile monitoring where EWMA is conducted in sample level, our exponential weights are imposed in time-point level and do not break across different samples. Denote $\boldsymbol{e}_u = \boldsymbol{e}_t^{(i)}, \boldsymbol{S}_u = \boldsymbol{S}_t^{(i)}$, then the EWMA monitoring statistic can be defined as

$$T_u^2 = \boldsymbol{Z}_u' \boldsymbol{\Phi}_u^{-1} \boldsymbol{Z}_u, \tag{17}$$

where

$$\boldsymbol{Z}_u = \gamma \boldsymbol{e}_u + (1-\gamma)\boldsymbol{Z}_{u-1}, \quad \boldsymbol{\Phi}_u = \mathrm{Var}(\boldsymbol{Z}_u),$$

where $\boldsymbol{Z}_0$ is initialized as $\boldsymbol{0}$, and $\gamma \in [0, 1]$ is the EWMA smoothing parameter. To compute $\boldsymbol{\Phi}_u$, we first note that $\boldsymbol{e}_t^{(i)}$ is orthogonal to $\boldsymbol{y}_1^{(i)}, \ldots, \boldsymbol{y}_{t-1}^{(i)}$ along with $\boldsymbol{y}_{t|t-1}^{(i)}$ according to the projection theorem of KF (Anderson, Moore, and Eslami 1982). Thus, we have $Cov(\boldsymbol{e}_t^{(i)}, \boldsymbol{e}_s^{(i)}) = E[\boldsymbol{e}_t^{(i)}(\boldsymbol{y}_s^{(i)} - \boldsymbol{y}_{s|s-1}^{(i)})'] = \boldsymbol{0}$, for $s = 1, \ldots, t-1$. Furthermore, with the independence assumption among different samples, we can derive that $\boldsymbol{e}_u$ is independent over different accumulative time points u. Therefore, we have

$$\boldsymbol{\Phi}_u = \mathrm{Var}[\gamma \boldsymbol{e}_u + (1-\gamma)\boldsymbol{Z}_{u-1}] = \gamma^2 \boldsymbol{S}_u + (1-\gamma)^2 \boldsymbol{\Phi}_{u-1}.$$

It should be pointed out that the forward estimation algorithm of $\boldsymbol{V}_{t|t-1}^{(i)}$ contains only Eqs. [7b], [7c], and [7e], which are irrelevant to the observations $\boldsymbol{y}_t^{(i)}$. Therefore, for better monitoring efficiency, $\boldsymbol{S}_u$ and $\boldsymbol{\Phi}_u^{-1}$ can be estimated in advance for preparation of online monitoring.

We set an upper control limit (UCL) for Eq. [17] and define once $T_u^2 >$UCL, the monitoring system will trigger an OC alarm. The detailed monitoring scheme is summarized in Algorithm 2. It is noted that the training phase requires full-length profiles, while the monitoring phase only requires past observations since it does not consider the backward recursion, which gives the feasibility of INPOM. Owe to the advantage of offline computation of $\boldsymbol{\Phi}_u^{-1}$, the computational complexity of calculating T_u^2 is reduced to $O(p^2 + pq)$, and the complexity of calculating OSPE by KF is $\mathcal{O}(p^{2.376} + q^2)$ (Thrun 2002). Overall, the computational complexity of our RSSM-INPOM control chart for one time point is $O(p^{2.376} + p^2 + q^2 + pq)$. This ensures fast computation for online monitoring and quick detection of anomalies.

Algorithm 2. In-profile monitoring scheme based on RSSM

Input: Data streams $\boldsymbol{y}_t^{(i)}$, IC model parameter $\boldsymbol{\Theta}_0$
Estimated $\boldsymbol{V}_{t|t}^{(i)}$ via Eqs. [7b], [7c], and [7e]
Estimated $\boldsymbol{S}_t^{(i)}$ via $\boldsymbol{S}_t^{(i)} = Var(\boldsymbol{e}_t^{(i)}) = \boldsymbol{C}\boldsymbol{V}_{t|t-1}^{(i)}\boldsymbol{C}' + \boldsymbol{R}$
Estimated $\boldsymbol{\Phi}_u^{-1}$ via $\boldsymbol{\Phi}_u = \gamma^2 \boldsymbol{S}_u + (1-\gamma)^2 \boldsymbol{\Phi}_{u-1}$
Result: whether trigger an OC alarm, if yes, output PRL_1 and TRL_1 via Eq. [16]

```
for u = 1, ..., T_1 + ··· + T_{i-1} + t, ... do
    Predict y_{t|t-1}^{(i)} = C x_{t|t-1}^{(i)} via Eq. [7] given y_1^{(i)}, ..., y_{t-1}^{(i)}
    Calculate e_t^{(i)} = y_t^{(i)} - y_{t|t-1}^{(i)} given y_t^{(i)}
    Calculate T_u^2 via Eq. [17]
    if T_u^2 > UCL then
        Trigger an OC alarm and return u as u_1, i as m_1, t as τ_1
        Break
    end
end
```

2.4. Guidelines for design and implementation

This section provides guidelines on designing and implementing the proposed RSSM-INPOM framework.

2.4.1. On choosing the Phase I sample size N

Generally, the larger sample size N and the larger average profile length $\bar{T}$ are, the more accurate estimation and the better performance of RSSM would be expected. Hence, as $\bar{T}$ increases, the required N decreases. This can be shown from the results in Section 2.3, As long as $N \times \bar{T} > 10\rho$, the residuals almost reach the minimum and are quite stable, where ρ is the number of parameters in $\hat{\boldsymbol{\Theta}}$. Hence, we suggest collecting at least $10\rho\bar{T}$ IC samples before Phase II monitoring. When sufficient IC data are unavailable, self-starting methods can be implemented by simultaneously updating parameter estimates and checking for OC conditions (Hawkins and Maboudou-Tchao 2007).

2.4.2. On determining the tuning parameters $\{\lambda_1, \lambda_2\}$

Small values of $\{\lambda_1, \lambda_2\}$ may lose the effect of regularizations, while large ones may induce more similar rows in $\boldsymbol{C}$ and more zero entries in $\boldsymbol{A}$, leading to an easier model which may not be able to describe the system. To search the optimal $\{\lambda_1, \lambda_2\}$ corresponding to a specific q (the state dimension), we suggest the Bayesian information criterion (BIC) (Durbin and Koopman 2012) to find proper values of $\{\lambda_1, \lambda_2\}$:

$$\{\hat{\lambda}_1, \hat{\lambda}_2\} = \arg\min_{\lambda_1, \lambda_2} \left[-2\log h\left(\boldsymbol{Y}^{(1:N)} | \hat{\boldsymbol{\Theta}}^{q}_{\lambda_1,\lambda_2}, \hat{\boldsymbol{X}}^{(1:N)}\right) + \rho^{q}_{\lambda_1,\lambda_2}\bar{T}\log N\right], \quad [18]$$

where $-2\log h(\boldsymbol{Y}^{(1:N)}|\hat{\boldsymbol{\Theta}}^{q}_{\lambda_1,\lambda_2}, \hat{\boldsymbol{X}}^{(1:N)}) = \sum_{i=1}^{N}\sum_{t=1}^{T_i}(\boldsymbol{y}_t^{(i)} - \hat{\boldsymbol{C}}\hat{\boldsymbol{x}}_t^{(i)})'\hat{\boldsymbol{R}}^{-1}(\boldsymbol{y}_t^{(i)} - \hat{\boldsymbol{C}}\hat{\boldsymbol{x}}_t^{(i)}) + \sum_{i=1}^{N} T_i \log|\hat{\boldsymbol{R}}|$ is the conditional log likelihood of all the observable variables given the estimated $\hat{\boldsymbol{\Theta}}^{q}_{\lambda_1,\lambda_2}$ via Algorithm 1 and the latent state variables $\hat{\boldsymbol{X}}^{(1:N)}$ estimated by KF and KS. $\rho^{q}_{\lambda_1,\lambda_2}$ is the number of nonzero components in $\hat{\boldsymbol{\Theta}}^{q}_{\lambda_1,\lambda_2}$. For searching convenience, we use grid search method for λ_1 and λ_2 on a predefined two-dimensional grid, where the grid range is proportional to the sample size which is determined case by case. Binary search in the grid can be adopted to efficiently get the best combination of $\{\lambda_1, \lambda_2\}$ for each alternative q (Zhang et al. 2021).

2.4.3. On selecting the number of states q

The selection of q shares a similar idea with $\{\lambda_1, \lambda_2\}$, since a small value of q may not be able to describe the system, while a large one may lead to overfitting. To keep a balance between model complexity and fitting result, we first estimate a range S of reasonable values for q based on singular value decomposition from our initialization algorithm in Appendix A. Apparently, the squared components of the diagonal matrix $\boldsymbol{E}$ indicate the variances explained by each state. Using this information, we can compute a range S for alternative q based on an interval of proportion of total explained variance, such as $(80\%, 99\%)$. Then we also follow BIC to find the optimal q for a desired modeling performance:

$$\hat{q} = \arg\min_{q\in S} \left[-2\log h\left(\boldsymbol{Y}^{(1:N)} | \hat{\boldsymbol{\Theta}}^{q}_{\lambda_1,\lambda_2}, \hat{\boldsymbol{X}}^{(1:N)}\right) + \rho^{q}_{\lambda_1,\lambda_2}\bar{T}\log N\right], \quad [19]$$

where $\{\lambda_1, \lambda_2\}$ are the selected best tuning parameters based on Eq. [18].

2.4.4. On determining the EWMA smoothing parameter γ

A small γ means we have a strong ability of remembering old observations, which helps accumulate small shift and therefore improves the detection ability. On the contrary, a large γ puts more emphases on recent observations and is able to detect large shift more quickly. To make a tradeoff between both sides, we usually set $\gamma \in [0.05, 0.15]$.

2.4.5. On determining the upper control limit

In the case that the sequential OSPEs are exactly independent, the EWMA statistic follows a χ^2(p) distribution in the asymptotic sense. Therefore, UCL can be set in accordance with a desired Type I error α. However, the independence assumption may not be generally satisfied since $\boldsymbol{e}_u$ is not precisely independent with its covariance $\boldsymbol{S}_u$. As such, an alternative way is to adjust the UCL numerically case by case. In particular, based on a prespecified average IC PRL ($APRL_0$), or average IC TRL ($ATRL_0$) through a large number of simulation replications (e.g., 5000 replications in our numerical studies), we set UCL by Monte Carlo simulation with Bisection searching algorithm (Qiu 2008).

3. Numerical studies

Now we use numerical experiments to evaluate our proposed method in two aspects: (1) how well RSSM describes multi-channel profiles, especially their WPC and BPC structures; (2) how fast and robust RSSM-INPOM is for anomaly detection. Specifically, in Section 2.3, we generate synthetic data from three designed models. The performance of our RSSM in modeling is investigated by four evaluation metrics. In

Section 2.3, the sensitivity and efficiency of RSSM-INPOM are demonstrated by evaluating the average PRL_1 ($APRL_1$) and average TRL_1 ($ATRL_1$), along with their standard deviations, i.e., $SDPRL_1$ and $SDTRL_1$, under several representative OC patterns with different shift magnitudes.

We also consider six baselines for comparison as follows: (1) SSM without regularizations; (2) First-order vector autoregressive model (VAR(1)) (Pan and Jarrett 2007); (3) VPCA (Nomikos and MacGregor 1995); (4) MPCA (Grasso, Colosimo, and Pacella 2014); (5) MFPCA (Paynabar, Zou, and Qiu 2016); (6) SMFPCA (Zhang et al. 2018b). The first two time series models are used to demonstrate the superiority of our RSSM-based dynamic modeling approach for unsynchronized samples, and the four PCA-based methods are used to demonstrate the superiority of our INPOM scheme.

3.1. Modeling performance evaluation

We first evaluate the modeling performance of different methods. Synthetic data are generated from the following three models, i.e., VAR(1), SSM and Fourier bases, which correspond to the modeling mechanisms of our baselines, and are also commonly used for numerical studies in many works.

i. Model (I): $\boldsymbol{Y}^{(i)} = \{\boldsymbol{y}_1^{(i)}, ..., \boldsymbol{y}_t^{(i)}, ..., \boldsymbol{y}_{T_i}^{(i)}\} \in \mathbb{R}^{6\times T_i}$ with $\boldsymbol{y}_t = \boldsymbol{b} + \boldsymbol{B}\boldsymbol{y}_{t-1} + \boldsymbol{\epsilon}_t$, where T_i is randomly generated by setting $T_i = \lfloor S_i + \frac{1}{2} \rfloor$ with $S_i \sim \mathcal{N}(40, 5)$. $\boldsymbol{\epsilon}_t \in \mathbb{R}^{6\times 1}$ is the white noise with every component $\epsilon_{jt} \overset{iid}{\sim} \mathcal{N}(0, 10^{-3})$. The other parameters of VAR(1) are set as below.

$$\boldsymbol{b} = \begin{bmatrix} 1 \\ 1 \\ 2 \\ 2 \\ -1 \\ -1 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} 0.6 & 0.2 & -0.2 & -0.2 & 0 & 0 \\ 0.2 & 0.6 & -0.2 & -0.2 & 0 & 0 \\ 0 & 0 & 0.6 & 0.2 & 0.2 & 0.2 \\ 0 & 0 & 0.2 & 0.6 & 0.2 & 0.2 \\ 0 & 0 & -0.1 & -0.1 & 0.6 & 0.2 \\ 0 & 0 & -0.1 & -0.1 & 0.2 & 0.6 \end{bmatrix}.$$

ii. Model (II): $\boldsymbol{Y}^{(i)} \in \mathbb{R}^{6\times T_i}$ driven by Eqs. [1] and [2] with latent states $\boldsymbol{X}^{(i)} \in \mathbb{R}^{3\times T_i}$, where T_i is randomly generated by setting $T_i = \lfloor S_i + \frac{1}{2} \rfloor$ with $S_i \sim \mathcal{N}(100, 10)$. The parameters of SSM are set as below.

$$\boldsymbol{A} = \begin{bmatrix} 0.95 & 0 & 0 \\ 0.9 & 0.9 & 0 \\ 0.5 & 0 & 1 \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0.5 \end{bmatrix}',$$
$$\boldsymbol{Q} = 10^{-5} \times \boldsymbol{I}_3, \boldsymbol{R} = 10^{-3} \times \boldsymbol{I}_6, \boldsymbol{\mu}_1 = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}', \boldsymbol{V}_1 = 10^{-3} \times \boldsymbol{I}_3.$$

iii. Model (III): $\boldsymbol{Y}^{(i)} \in \mathbb{R}^{6\times 200}$ with $\boldsymbol{Y}^{(i)} = \sum_{k=1}^{3} \boldsymbol{\xi}_{ik}\boldsymbol{v}_k' + \boldsymbol{\varepsilon}_i$, where $\boldsymbol{\varepsilon}_i \in \mathbb{R}^{6\times 200}$ is the white noise with every component $\varepsilon_{ijn} \overset{iid}{\sim} \mathcal{N}(0, 10^{-3})$. $\boldsymbol{v}_k \in \mathbb{R}^{200\times 1} (k = 1, ..., 3)$ are the first three non-constant Fourier bases, i.e., $v_{kl} = \cos(kt_l + k\pi)$, with a grid of $n = 200$ equally spaced sensing points $t_l (l = 1, ..., n)$ in $[0, 2\pi]$. $\boldsymbol{\xi}_{ik} (k = 1, ..., 3)$ is the score set to be clustered which follows a six-dimensional normal distribution as $\boldsymbol{\xi}_{ik} \sim \mathcal{N}(\boldsymbol{\zeta}_k, \boldsymbol{\Xi}_k)$, where $\boldsymbol{\zeta}_k$ and $\boldsymbol{\Xi}_k$ are set as below.

$$\begin{bmatrix} \boldsymbol{\zeta}_1 & \boldsymbol{\zeta}_2 & \boldsymbol{\zeta}_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -1 & -1 & 0.1 & 0.1 \\ 1 & 1 & 1 & 1 & 0.5 & 0.5 \\ 1 & 1 & -1 & -1 & 1 & 1 \end{bmatrix}',$$
$$\boldsymbol{\Xi}_k = 10^{-3} \times \boldsymbol{I}_6, k = 1, ..., 3.$$

Figure 2 shows an intuitive illustration of 30 samples generated from the above three IC models. Note that we focus on cluster-correlated data, the synthetic data all have clustered structures. For each model, we generate 50 training samples for offline estimation of $\boldsymbol{\Theta}_0$ and test each model on 50 testing samples. In particular, we learn our RSSM by Algorithm 1, which converges within 50 EM iterations. For the SSM-based methods, we determine the state dimension q via BIC criterion. For fair competition, we select the number of PCs for the PCA-based methods to be q as well. Notably, as for the PCA-based models, they require that all samples should be synchronized. Therefore, for Model (I) and Model (II), when different samples are misaligned, we first remove the non-synchronization effect of different samples by dynamic time-warping (DTW) (Berndt and Clifford 1994).

The residuals of different fitted models are analyzed for performance evaluation. We recall that residuals are computed as the difference between the observed data and predictions for time series models (or the reconstruction curves for the PCA-based methods). If the residuals are found neither autocorrelated nor cross-correlated, then the model can successfully capture the features of multi-channel profiles. Statistical tests can be used for this purpose, such as Durbin-Watson test for autocorrelation, and Pearson correlation analysis for cross-correlation. Consequently, we test the modeling performance by comparing the

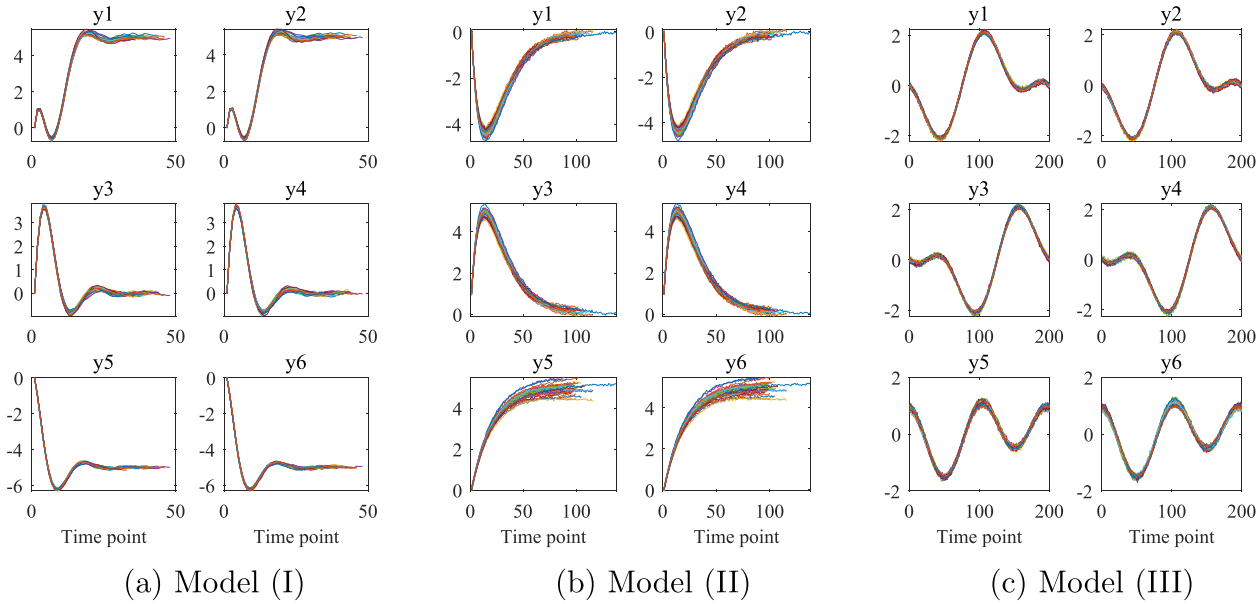


Figure 2. Illustration of the synthetic data from the three IC models.

Table 1. MAE, MSE, MDW and MPCC comparison of RSSM, SSM, VAR(1), SMFPCA, MFPCA, MPCA and VPCA on 50 test samples of Models (I-III) (numbers in parentheses are the corresponding standard deviations).

		RSSM	SSM	VAR(1)	SMFPCA	MFPCA	MPCA	VPCA
Model (I)	MAE($\times 10^{-2}$)	**1.94**(0.04)	2.15(0.05)	2.83(0.24)	4.05(0.57)	4.03(0.56)	4.98(0.43)	4.63(0.64)
	MSE($\times 10^{-3}$)	**0.59**(0.02)	0.78(0.03)	1.25(0.21)	3.01(0.97)	2.99(0.95)	4.07(0.80)	3.66(1.11)
	MDW	1.29(0.13)	1.14(0.11)	**1.54**(0.12)	0.76(0.12)	0.77(0.13)	0.53(0.09)	0.65(0.10)
	MPCC	**0.01**(0.01)	0.04(0.01)	0.02(0.04)	0.15(0.06)	0.14(0.06)	0.17(0.08)	0.17(0.05)
Model (II)	MAE($\times 10^{-2}$)	**2.51**(0.13)	2.55(0.15)	9.34(1.39)	7.41(0.45)	7.41(0.45)	9.10(2.45)	7.93(0.64)
	MSE($\times 10^{-3}$)	**1.00**(0.10)	1.03(0.14)	14.3(2.00)	9.80(1.21)	9.82(1.24)	14.6(7.02)	10.8(1.79)
	MDW	**1.93**(0.10)	1.90(0.08)	1.65(0.52)	0.34(0.04)	0.34(0.04)	0.28(0.05)	0.33(0.05)
	MPCC	**0.01**(0.02)	0.02(0.02)	0.03(0.03)	0.06(0.03)	0.06(0.03)	0.11(0.03)	0.07(0.02)
Model (III)	MAE($\times 10^{-2}$)	**2.77**(0.38)	2.99(0.40)	7.04(0.46)	4.91(0.37)	4.93(0.38)	5.07(0.42)	5.14(0.43)
	MSE($\times 10^{-3}$)	**1.23**(0.35)	1.46(0.39)	9.79(0.92)	3.72(0.51)	3.74(0.53)	3.91(0.58)	4.00(0.59)
	MDW	**1.47**(0.21)	1.35(0.20)	1.08(0.06)	0.78(0.03)	0.77(0.03)	0.70(0.04)	0.71(0.03)
	MPCC	**0.03**(0.02)	0.05(0.03)	0.03(0.02)	0.16(0.02)	0.16(0.02)	0.17(0.04)	0.17(0.02)

[1] $MAE = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{p\bar{T}_i}\sum_{j=1}^{p}\sum_{t=1}^{T_i}|e_{j,t}^{(i)}|$, $MSE = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{p\bar{T}_i}\sum_{j=1}^{p}\sum_{t=1}^{T_i}(e_{j,t}^{(i)} - \bar{e}_j^{(i)})^2$,

$MDW = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{p}\sum_{j=1}^{p}\frac{\sum_{t=1}^{T_i-1}(e_{j,t+1}^{(i)} - e_{j,t}^{(i)})^2}{\sum_{t=1}^{T_i}(e_{j,t}^{(i)})^2}$, $MPCC = \frac{1}{N}\sum_{i=1}^{N}\frac{2}{p(p-1)}\sum_{j_1=1}^{p-1}\sum_{j_2=j_1+1}^{p}\text{coeff}(\boldsymbol{e}_{j_1,:}^{(i)}, \boldsymbol{e}_{j_2,:}^{(i)})$.

residuals in terms of four evaluation metrics, i.e., mean absolute error (MAE), mean standard error (MSE), mean Durbin–Watson (MDW) statistic, and mean Pearson correlation coefficient (MPCC). Their definitions are given in the footnotes of Table 1. Generally, MAE, MSE, MPCC which are smaller and MDW which is more closed to two are preferred. The MAE, MSE, MDW and MPCC of different models are shown in Table 1, where the best results are all shown in bold face. The results can be summarized as the following points.

1. The proposed RSSM has the smallest MAEs and MSEs, followed by SSM. Also, RSSM has larger MDWs and smaller MPCCs than SSM, which shows it can capture WPC and the clustered BPC better, and hence provides a more accurate approximation to the underlying process. In addition, the standard deviations of the SSM-based methods are significantly smaller than the PCA-based models, especially for Model (I) and Model (II). This is because the SSM-based methods can capture the dynamic evolution mechanism inside the data better.
2. SMFPCA and MFPCA are less desirable than the SSM-based models. This is because they do not consider WPC and the structured BPC very carefully, which can be demonstrated by the undesirable MDWs and MPCCs. As to MPCA and VPCA, they perform worse than SMFPCA and MFPCA. This makes sense since the BPC structure assumed by MPCA is not satisfied by the

data, while VPCA ignores BPC by stacking up channels into a high-dimensional vector. VAR(1) performs the worst except for Model (I), since it can only describe limited WPC by such an easy model. Though it has quite large values of MDW, it ignores BPC and has poor performance in tracking signals. These can be further demonstrated by the monitoring results later.

3. To further investigate the effect of the regularizations, we also compare the estimated parameters of RSSM and SSM for Model (II). We simulate different circumstances by scaling the transition matrix as $s\boldsymbol{A}$, where s is a scale parameter. Table 2 shows the true transition matrix $s\boldsymbol{A}$ and observation matrix $\boldsymbol{C}$, and their estimations by RSSM and SSM when $s = 0.8$. As expected, the l_1 regularization can successfully induce zero entries in $\boldsymbol{A}$, and the graph Laplacian regularization tends to encourage a clustered $\boldsymbol{C}$, while SSM fails to estimate $\boldsymbol{A}$ and $\boldsymbol{C}$ accurately. The superiority of RSSM is more evident as s increases.
4. Finally, a study of the required sample size N for Phase I modeling is conducted on the above three models with different average profile lengths of 40, 100 and 200. For each model, we compute the MSEs under different sample sizes by 100 replications. As shown in Figure 3, 50, 20 and 10 samples can achieve stable and quite small MSEs for the three models respectively.

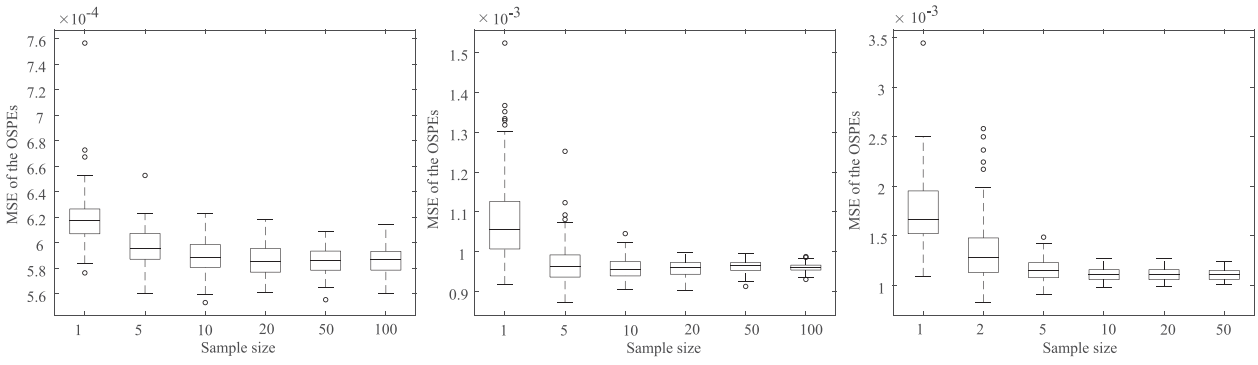
Table 2. The true transition matrix $s\boldsymbol{A}$ and observation matrix $\boldsymbol{C}$, and their estimations by RSSM and SSM when $s = 0.8$.

	True			RSSM			SSM		
$s\boldsymbol{A}$	0.76	0	0	0.76	0.00	0	0.94	−0.70	0.09
	0.72	0.72	0	0.73	0.71	0.02	0.06	0.56	0.03
	0.4	0	0.8	0.43	0	0.82	−0.00	0.12	0.75
$\boldsymbol{C}$	1	−1	0	1.00	−1.00	0.00	0.23	0.48	−0.13
	1	−1	0	1.00	−1.00	0.00	0.23	0.49	−0.14
	0	1	0	0.06	0.96	0.07	−0.34	−0.11	0.09
	0	1	0	0.06	0.96	0.07	−0.34	−0.11	0.08
	0	0	0.5	−0.02	0.01	0.48	−0.08	0.20	0.27
	0	0	0.5	−0.02	0.01	0.48	−0.08	0.20	0.26

3.2. Monitoring performance evaluation

Now we test and compare the monitoring performance of RSSM-INPOM together with the six baselines for the above three models. Since for multi-channel profiles, OC scenarios can be numerous and it is impossible to comprehensively study the performance of all OC scenarios, here we choose the following six representative scenarios.

i. Scenario (I): step shift in the j^{th}-channel profiles since sample $m_0 + 1 = 1$ and time point $\tau_0 + 1 = 21$, i.e., $y_{jt}^{(i)}(OC) = y_{jt}^{(i)}(IC) + \delta\mathcal{I}_1(t \geq \tau_0), \forall i, t$, where $\mathcal{I}_1$ is the indicator function which equals 1 when its condition is true and equals 0 otherwise. This shift simulates the sensor measurement drift.
ii. Scenario (II): mean shift in the j^{th}-channel profiles since sample $m_0 + 1 = 1$ and time point $\tau_0 + 1 = 1$, i.e., $y_{jt}^{(i)}(OC) = y_{jt}^{(i)}(IC) \times (1 - \delta) + \frac{t}{T_i} \times y_{jT_i}^{(i)}(IC) \times \delta, \forall i, t$. This shift simulates the mechanical degradation of a system component.
iii. Scenario (III): slope shift in the j^{th}-channel profiles since sample $m_0 + 1 = 1$ and time point $\tau_0 + 1 = 1$, i.e., $y_{jt}^{(i)}(OC) = y_{jt}^{(i)}(IC) - t \times \delta, \forall i, t$. This shift simulates the mechanical degradation of a system component.
iv. Scenario (IV): spike shift in the j^{th}-channel profiles since sample $m_0 + 1 = 1$ and time point $\tau_0 + 1 = 21$, i.e., $y_{jt}^{(i)}(OC) = y_{jt}^{(i)}(IC) + \delta\mathcal{I}_2(\tau_0 \leq t \leq \tau_0 + 4), \forall i, t$, where $\mathcal{I}_2 = [0.33, 0.67, 1.00, 0.67, 0.33]$ for $\tau_0 \leq t \leq \tau_0 + 4$ and equals 0 otherwise. This shift simulates the abnormal input from the external environment.



(a) Model (I) (b) Model (II) (c) Model (III)

Figure 3. MSEs of 100 replications under different sample sizes.

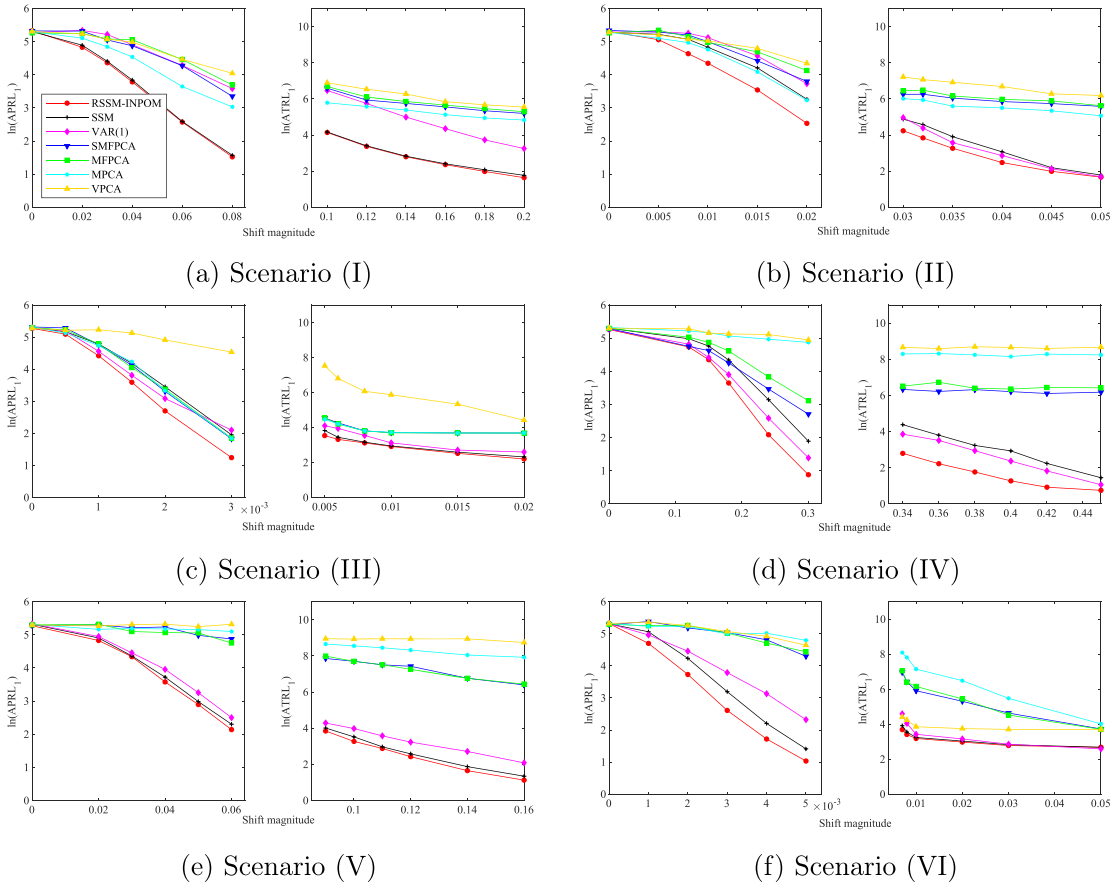


Figure 4. $APRL_1$ and $ATRL_1$ comparison in logarithmic scale of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA and VPCA with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (I-VI) of Model (I).

v. Scenario (V): random shift in the j^{th}-channel profiles since sample $m_0 + 1 = 1$ and time point $\tau_0 + 1 = 1$, i.e., $y_{jt}^{(i)}(OC) = y_{jt}^{(i)}(IC) + \Delta, \Delta \sim \mathcal{N}(0, \delta), \forall i, t$. This shift simulates the random measurement error.

vi. Scenario (VI): parameter shift of a specific model parameter θ since sample $m_0 + 1 = 1$ and time point $\tau_0 + 1 = 1$, i.e., $\theta_{OC} = \theta_{IC} - \delta$. We choose different θ for different IC models as specified in Tables B1–B6 of Appendix B. This shift simulates the changed dynamics inside the system.

For fair comparison, the monitoring statistics for SSM and VAR(1) are also constructed by T^2 statistic based on OSPEs, so they are INPOM-based methods. The monitoring statistics for the PCA-based models are constructed according to their corresponding works, which have to wait for monitoring until the whole profile is available. For all the methods, we add EWMA with $\gamma = 0.1$ and find UCL via Monte Carlo simulation such that $APRL_0$ equals 200. We consider 12 shift magnitudes of δ under different scenarios for each model. For the first six shift magnitudes, $APRL_1$s and $SDPRL_1$s based on 1000 simulation replications are compared. When the shift magnitudes are larger, we compare their $ATRL_1$s and $SDTRL_1$s to evaluate the in-profile detection delay in time-point level. Figures 4–6 show the $APRL_1$s and $ATRL_1$s under the above six OC scenarios of the three models. See Tables B1–B6 for the detailed results with the corresponding standard deviations, which can be summarized as the following points.

1. RSSM-INPOM control chart can generally achieve the best performance for different kinds of data and different OC patterns, especially when the shift magnitudes are large. The merit is not only owed to our RSSM's accurate modeling, but also to its finer granular INPOM mechanism.
2. As shown in Tables B1–B6, $SDPRL_1$s and $SDTRL_1$s of the PCA-based control charts are larger than the time series-based methods for Model (I) and Model (II), which means they are not suitable for dynamic system. On the contrary, $SDPRL_1$s and $SDTRL_1$s of RSSM-INPOM are still stable for Model (III), indicating that it is generally more robust in profile monitoring.

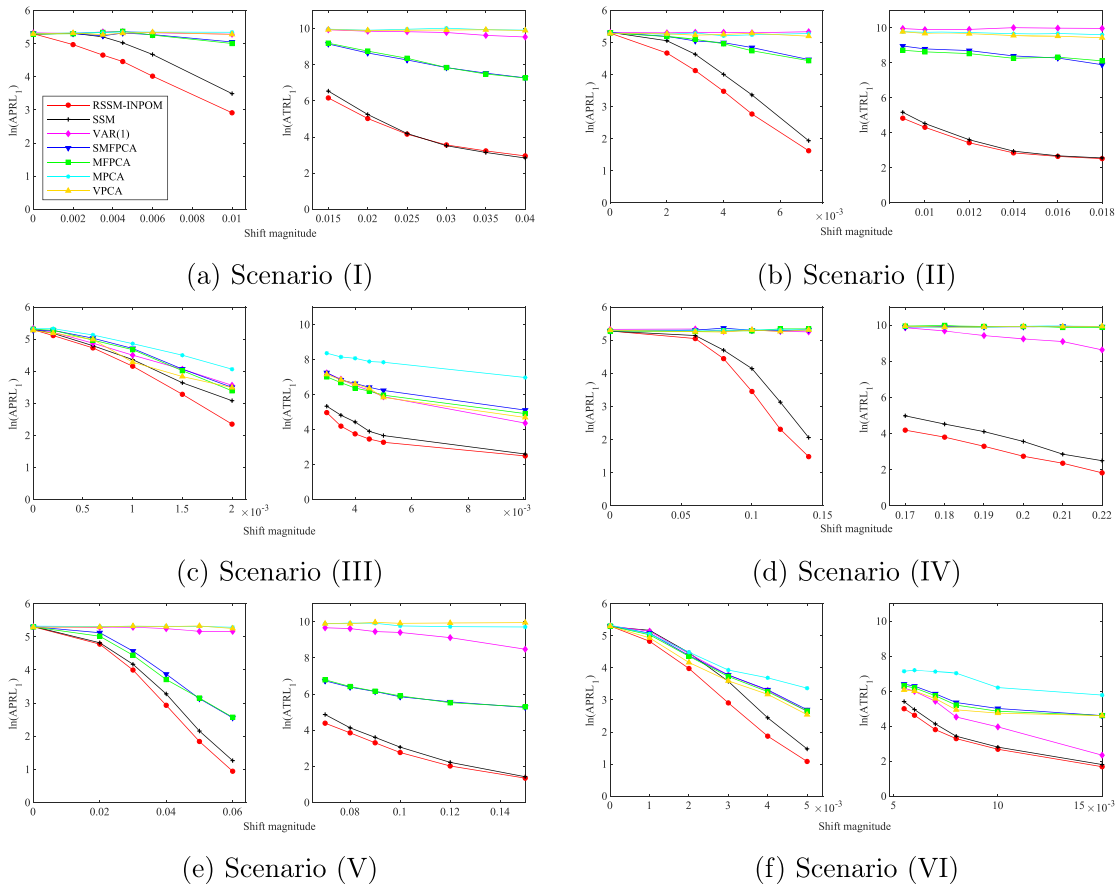


Figure 5. $APRL_1$ and $ATRL_1$ comparison in logarithmic scale of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA and VPCA with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (I-VI) of Model (II).

3. For the other six baselines, their monitoring performance not only depends on their modeling performance, but also relates to the OC scenario. Take Model (II) for instance, the PCA-based methods perform better than VAR(1) in most cases, however, they show lower detection precision under Scenario (IV), as shown in Figure 5. This is because the alignment step of the PCA-based methods weakens the pattern of sparse shifts significantly.
4. For different models, our RSSM-INPOM shows greater superiority to other control charts when the time length of each sample is longer. For instance, under Scenario (III), RSSM-INPOM reduces TRL_1 of SMFPCA from 94.2 to 34.5 for Model (I), from 1410 to 144 for Model (II), and from 3300 to 235 for Model (III), respectively. This is reasonable since the PCA-based algorithms cannot be implemented until the whole profile has been collected, which causes longer detection delay when the profile length is longer.

4. Case studies

In this section, to investigate the validity of the proposed RSSM-INPOM control chart in real applications, we apply it into two real cases from advanced manufacturing systems, i.e., the diffusion process in semiconductor manufacturing and the pipe-casing tightening process. Briefly, we summarize the step-by-step implementation of RSSM-INPOM as follows: (1) Preprocess the signals by real-time denoising and normalization; (2) Learn RSSM using historical IC samples according to Algorithm 1; (3) Set UCL by Monte Carlo simulation; (4) Run RSSM-INPOM control chart for online samples according to Algorithm 2.

4.1. Diffusion process

In this diffusion process of semiconductor manufacturing, in-situ sensors located around a diffusion furnace are used for real-time data collection of each wafer (Zhang, Zhang, and Chen 2017). We analyze nine representative sensors and denote them as S1-S9, with S1-S6 shown in Figure 1(a). The time lengths of

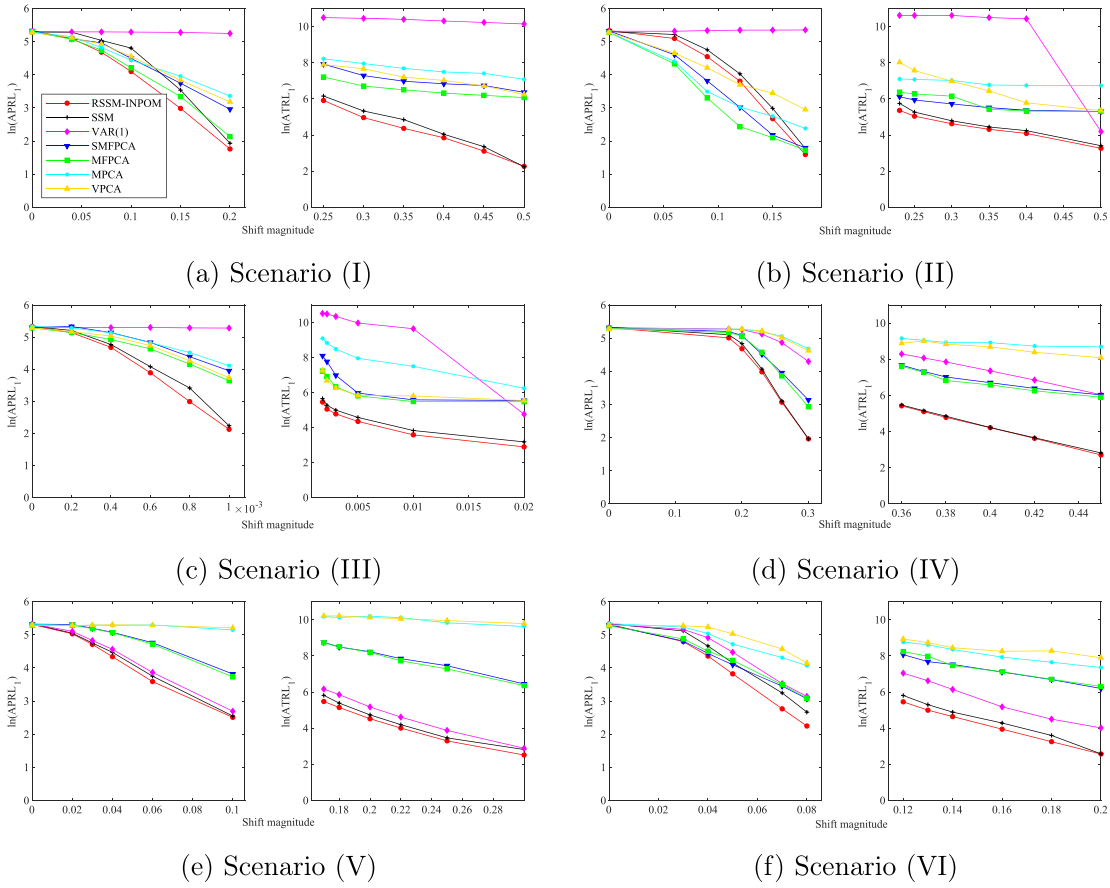
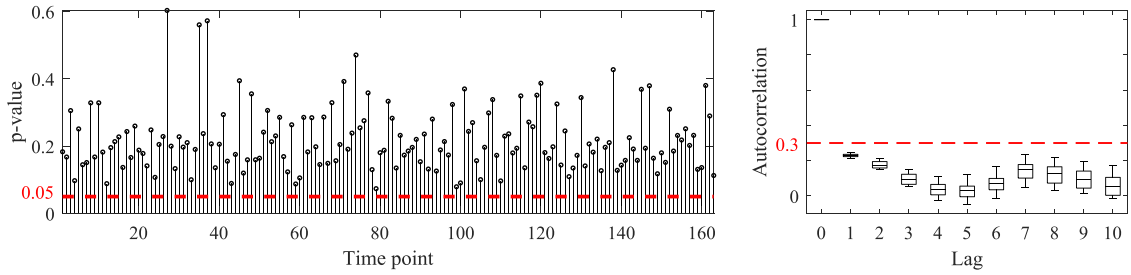


Figure 6. $APRL_1$ and $ATRL_1$ comparison in logarithmic scale of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA and VPCA with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (I-VI) of Model (III).

different samples vary from 190 to 240. Clearly, there exists clustered BPC structure in this case. For example, voltage sensors (S1 and S2) are jointly influenced by circuit power and local load; temperature sensors (S3, S4, S7-S9) are all depended on thermal energy; while pressure sensors (S5 and S6) are both affected by gas flow rate. Consequently, these sensors can be divided into three groups, and the four identified latent states can be regarded as circuit power, load, thermal energy, and gas flow rate. We have 28 IC samples in total. Since the real OC data are confidential, we generate OC signals to mimic the real OC situation as Figure 1(a) shows. These OC patterns are designed by engineers based on the true anomaly patterns in the manufacturing process. In particular, a slope shift occurs in early sensing points of S1 and S2. Also, both S5 and S6 have an earlier drop since time point $\tau_0 + 1 = 75$.

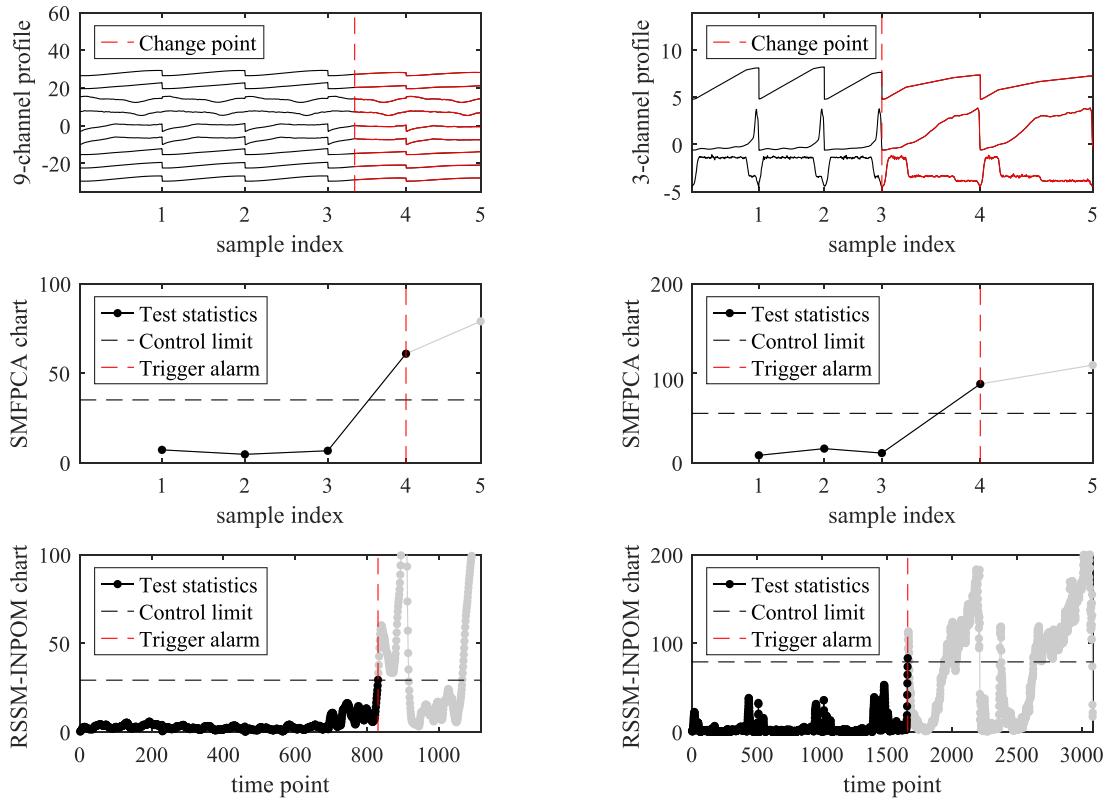
The first 25 samples are used as historical IC set to train our RSSM, and the rest three IC samples followed by two OC samples are used for monitoring performance evaluation. We select the dimension of latent state variable as $q = 4$ and tuning parameters as $\lambda_1 = 0.5, \lambda_2 = 2$ based on BIC. The EM algorithm converges in the 77^{th} iteration. To test the goodness-of-fit of RSSM on real data, Henze–Zirkler multivariate normality test is performed on the OSPEs at each time point (Henze and Zirkler 1990). As shown in Figure 7(a), there is no sufficient evidence to reject the null hypothesis that OSPEs are normally distributed, since almost all p-values associated with the tests are larger than the predetermined significance level of 0.05. In addition, Figure 7(b) shows the lag-a ($a = 1, \ldots, 10$) autocorrelation of OSPEs, and it can be observed that OSPEs almost have no significant autocorrelation. Therefore, both the normality and independence assumptions of OSPEs can be validated, which are required for in-profile monitoring.

Now we run RSSM-INPOM control chart for monitoring. Figure 8(a) presents the comparison of RSSM-INPOM control chart and the best SMFPCA control chart selected from the six baselines, where the dash red lines mark the time when OC alarms are triggered. It can be seen that SMFPCA control chart cannot detect anomalies until the current OC sample with a time length of 216 has been entirely obtained, while RSSM-INPOM control chart has a quick response to the OC signals and successfully exceeds the UCL at time point $\tau_1 = 139$ of sample $m_1 = 4$. Consequently, we get the corresponding profile run length $PRL_1^{SMFPCA} =$



(a) Normality test (b) Autocorrelation test

Figure 7. Goodness-of-fit test on OSPEs of the diffusion process.



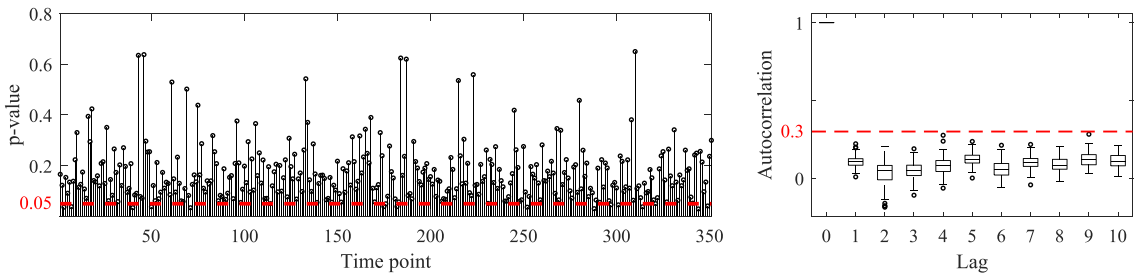
(a) The diffusion process (b) The pipe-casing tightening process

Figure 8. Five sequential samples and their corresponding monitoring statistics of SMFPCA control chart and RSSM-INPOM control chart with $\gamma = 0.1$.

$\mathrm{PRL}_1^{\mathrm{RSSM-INPOM}} = 1$, and the time run length $\mathrm{TRL}_1^{\mathrm{SMFPCA}} = 216 - 74 = 143$ and $\mathrm{TRL}_1^{\mathrm{RSSM-INPOM}} = 139 - 74 = 65$. It is noted that in practice, all the control charts stop running once an alarm is triggered, though for illustrative purpose, we show the whole monitoring statics for the five sequential samples.

4.2. Pipe-casing tightening process

In this case study, a motor drives a threaded pipe to tighten into a casing, where a spring is installed for buffering. Three sensors located on a backup tong are used to collect key process variables during the screw turning process, i.e., the number of turns, torque signal, and rotating speed signals. The first two sensors (S1 and S2) measure the mechanical deformation of the spring, which can be theoretically represented by the linear relationship between the number of turns (S1) and the torque (S2). Therefore, the intercept and slope of the fitted curve by S1 w.r.t S2 can be considered as two identified latent states of the process. In addition, it is the system power that influences the



(a) Normality test (b) Autocorrelation test

Figure 9. Goodness-of-fit test on the OSPEs of the pipe-casing tightening process.

rotating speed (S3) which records the electrical property, and thus the system power is the third latent state. According to this domain knowledge, the state dimension is set to be three. In the actual manufacturing process, anomalies happen occasionally due to stripping of threads or slipping of pressure joints. This will influence the physical dynamics of the spring, which is further reflected in sensor signals. We have 98 IC samples and three OC samples, with slope shifts in S1 and S2, and step shifts in S3 since $\tau_0 + 1 = 1$, as Figure 1(b) shows. It should be noted that for different samples, their profile lengths vary from 350 to 650 due to unavoidable variations from the pre-tightening stage. For more detailed information about the background, please refer to Du et al. (2018).

Now we use 95 IC samples to learn our RSSM, and the rest three IC samples followed by two OC samples are used for monitoring. Since there is no clustered BPC structure in this case, we set λ_1 as zero for the graph Laplacian regularization. λ_2 is set to be 0.34 for the l_1 regularization. The EM algorithm converges in the 126$^{\text{th}}$ iteration. Further, goodness-of-fit test is performed before monitoring. As shown in Figure 9, we can draw the same conclusion as in Section 2.3. In Phase II monitoring, we construct RSSM-INPOM control chart and the best MFPCA control chart from the baselines, as shown in Figure 8(b). We get the corresponding profile run length $\text{PRL}_1^{\text{SMFPCA}} = \text{PRL}_1^{\text{RSSM-INPOM}} = 1$, and the time run length $\text{TRL}_1^{\text{SMFPCA}} = 757$ and $\text{TRL}_1^{\text{RSSM-INPOM}} = 199$. Clearly, RSSM-INPOM control chart has superior monitoring efficiency to the current control charts, which is more desirable for processes that need long time to complete.

5. Concluding remarks

Although profile monitoring has been extensively studied, the challenges concerning how to design in-profile monitoring schemes for multi-channel profiles with clustered BPC are to be addressed. This article is the first to combine profile monitoring with time series-based model, which we believe will bring significant impact in more efficient monitoring schemes. In particular, we propose a RSSM-INPOM control chart to dynamically monitor multi-channel profiles on the fly. RSSM aims to describe the dynamic evolution mechanism of each sensing point inside a profile, thus giving the feasibility of INPOM and being extremely effective for misaligned samples. Moreover, by incorporating graph similarity and sparsity on the model parameters, it can not only describe the clustered BPC of multi-channel profiles, but also help avoid overfitting. The model parameters are efficiently learned via an EM-incorporated MAP framework. Considering that OSPE generated by this RSSM naturally indicates the system deviation from its normal state, a T^2 monitoring statistic with EWMA technique is constructed based on OSPE. Extensive numerical studies and two real cases demonstrate that our RSSM-INPOM control chart possesses a better monitoring performance than literature methods.

Along with this research direction, there are several potential valuable extensions. First, by adding an input variable, RSSM-INPOM control chart can be extended to describe systems with feedback control. Second, systems may be time varying in some multi-stage manufacturing applications, and an extension of the current static RSSM to a switched RSSM would be desirable. Another potential venue is to consider nonlinear cases by further extending the current linear model to a nonlinear model, which would be able to describe more general manufacturing systems.

About the authors

Peiyao Liu is a PhD student in the Department of Industrial Engineering at Tsinghua University. Her email is liupy20@mails.tsinghua.edu.cn.

Juan Du is an Assistant Professor in the Smart Manufacturing Thrust and an Affiliate Assistant Professor

in the Department of Mechanical and Aerospace Engineering at The Hong Kong University of Science and Technology. Her email is juandu@ust.hk.

Yangyang Zang is an engineer at China Aero Polytechnology Establishment. She received her PhD in the Department of Industrial Engineering from Tsinghua University. Her email is zangyy12@163.com.

Chen Zhang is an Associate Professor in the Department of Industrial Engineering at Tsinghua University. Her email is zhangchen01@tsinghua.edu.cn.

Kaibo Wang is a Professor in the Department of Industrial Engineering at Tsinghua University. His email is kbwang@tsinghua.edu.cn.

Acknowledgments

We thank the editor and two anonymous referees for their valuable comments and suggestions for improving our work.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the NSFC Grant 71901131, 71932006, 72001139, 71731008, the BNSF Grant 9222014, the ASFC Grant 2020Z063058001, the SASTIND Grant JSZL2021205A001, and the Tsinghua University Intelligent Logistics & Supply Chain Research Center Grant THUCSL20182911756-001.

ORCID

Peiyao Liu http://orcid.org/0000-0003-3674-9163
Juan Du http://orcid.org/0000-0002-6018-2972
Yangyang Zang http://orcid.org/0000-0002-4837-6440
Chen Zhang http://orcid.org/0000-0002-4767-9597
Kaibo Wang http://orcid.org/0000-0001-9888-4323

Data availability statement

The data that support the findings of this study are openly available in Mendeley Data at https://data.mendeley.com/datasets/sfv654hbnd/1.

References

Anderson, B. D. O., J. B. Moore, and M. Eslami. 1982. Optimal filtering. *IEEE Transactions on Systems, Man, and Cybernetics* 12 (2):235–6. doi: 10.1109/TSMC.1982.4308806.

Ando, R. K, and T. Zhang. 2007. Learning on graph with Laplacian regularization. In Advances in *neural information processing systems*, 25–32.

Barber, D. 2012. *Bayesian reasoning and machine learning*. Cambridge: Cambridge University Press.

Bartels, R. H, and G. W. Stewart. 1972. Solution of the matrix equation ax + xb = c [f4]. *Communications of the ACM* 15 (9):820–6. doi: 10.1145/361573.361582.

Berndt, D. J, and J. Clifford. 1994. Using dynamic time warping to find patterns in time series. in KDD workshop, Seattle, WA, vol. 10, 359–70.

Du, J., X. Zhang, X. Xu, and J. Shi. 2018. A novel critical point detection method for mechanical deformation in tightening processes. *Journal of Manufacturing Systems* 48:157–65. doi: 10.1016/j.jmsy.2018.07.007.

Durbin, J, and S. J. Koopman. 2012. *Time series analysis by state space methods*, vol. 38, Oxford: Oxford University Press.

Eyvazian, M., R. Noorossana, A. Saghaei, and A. Amiri. 2011. Phase II monitoring of multivariate multiple linear regression profiles. *Quality and Reliability Engineering International* 27 (3):281–96. doi: 10.1002/qre.1119.

Gevers, M., and V. Wertz. 1984. Uniquely identifiable state-space and ARMA parametrizations for multivariable linear systems. *Automatica* 20 (3):333–47. doi: 10.1016/0005-1098(84)90048-7.

Ghahramani, Z., and G. E. Hinton. 1996. Parameter estimation for linear dynamical systems. Technical report, Technical Report CRG-TR-96-2, University of Totronto, Department of Computer Science.

Grasso, M., B. Colosimo, and M. Pacella. 2014. Profile monitoring via sensor fusion: the use of PCA methods for multichannel data. *International Journal of Production Research* 52 (20):6110–35. doi: 10.1080/00207543.2014.916431.

Grasso, M., A. Menafoglio, B. M. Colosimo, and P. Secchi. 2016. Using curve-registration information for profile monitoring. *Journal of Quality Technology* 48 (2):99–127. doi: 10.1080/00224065.2016.11918154.

Hawkins, D. M., and E. M. Maboudou-Tchao. 2007. Self-starting multivariate exponentially weighted moving average control charting. *Technometrics* 49 (2):199–209. doi: 10.1198/004017007000000083.

Henze, N, and B. Zirkler. 1990. A class of invariant consistent tests for multivariate normality. *Communications in Statistics - Theory and Methods* 19 (10):3595–617. doi: 10.1080/03610929008830400.

Jong, P. D. 1991. The diffuse Kalman filter. *The Annals of Statistics* 19 (2):1073–83. doi: 10.1214/aos/1176348139.

Khedmati, M, and S. T. A. Niaki. 2016. Phase II monitoring of general linear profiles in the presence of between-profile autocorrelation. *Quality and Reliability Engineering International* 32 (2):443–52. doi: 10.1002/qre.1762.

Li, Z., N. D. Sergin, H. Yan, C. Zhang, and F. Tsung. 2020. Tensor completion for weakly-dependent data on graph for metro passenger flow prediction. *Proceedings of the AAAI Conference on Artificial Intelligence* 34 (04): 4804–10. doi: 10.1609/aaai.v34i04.5915.

Liu, Z, and M. Hauskrecht. 2013. Sparse linear dynamical system with its application in multivariate clinical time series.

Liu, J., J. Shi, and S. J. Hu. 2009. Quality-assured setup planning based on the stream-of-variation model for multi-stage machining processes. *IIE Transactions* 41 (4): 323–34. doi: 10.1080/07408170802108526.

Nomikos, P, and J. F. MacGregor. 1995. Multivariate SPC charts for monitoring batch processes. *Technometrics* 37 (1):41–59. doi: 10.1080/00401706.1995.10485888.

Noorossana, R., M. Eyvazian, A. Amiri, and M. A. Mahmoud. 2010. Statistical monitoring of multivariate multiple linear regression profiles in Phase I with

calibration application. *Quality and Reliability Engineering International* 26 (3):291–303. doi: 10.1002/qre.1066.

Overschee, P. V, and B. D. Moor. 1996. *Subspace identification for linear systems: theory, implementation, applications*. New York: Kluwer.

Pan, X, and J. Jarrett. 2007. Using vector autoregressive residuals to monitor multivariate processes in the presence of serial correlation. *International Journal of Production Economics* 106 (1):204–16. doi: 10.1016/j.ijpe.2006.07.002.

Paynabar, K., J. J. Jin, and M. Pacella. 2013. Monitoring and diagnosis of multichannel nonlinear profile variations using uncorrelated multilinear principal component analysis. *IIE Transactions* 45 (11):1235–47. doi: 10.1080/0740817X.2013.770187.

Paynabar, K., C. Zou, and P. Qiu. 2016. A change-point approach for Phase-I analysis in multivariate profile monitoring and diagnosis. *Technometrics* 58 (2):191–204. doi: 10.1080/00401706.2015.1042168.

Qiu, P. 2008. Distribution-free multivariate process control based on log-linear modeling. *IIE Transactions* 40 (7): 664–77. doi: 10.1080/07408170701744843.

Qiu, P, and D. Xiang. 2014. Univariate dynamic screening system: an approach for identifying individuals with irregular longitudinal behavior. *Technometrics* 56 (2): 248–60. doi: 10.1080/00401706.2013.822423.

Qiu, P., X. Zi, and C. Zou. 2018. Nonparametric dynamic curve monitoring. *Technometrics* 60 (3):386–97. doi: 10.1080/00401706.2017.1361340.

Ramsay, J, and B. Silverman. 2005. *Functional data analysis*. New York: Springer.

Rao, N., H.-F. Yu, P. K. Ravikumar, and I. S. Dhillon. 2015. Collaborative filtering with graph information: consistency and scalable methods. in Advances in neural information processing systems, 2107–15.

Shi, J. 2006. *Stream of variation modeling and analysis for multistage manufacturing processes*. Boca Raton: CRC Press.

Thrun, S. 2002. Probabilistic robotics. *Communications of the ACM* 45 (3):52–7. doi: 10.1145/504729.504754.

Wu, C. J. 1983. On the convergence properties of the EM algorithm. *The Annals of Statistics* 11 (1):95–103. doi: 10.1214/aos/1176346060.

Xiang, L, and F. Tsung. 2008. Statistical monitoring of multistage processes based on engineering models. *IIE Transactions* 40 (10):957–70. doi: 10.1080/07408170701880845.

Zhang, Y., Z. He, C. Zhang, and W. H. Woodall. 2014. Control charts for monitoring linear profiles with within-profile correlation using Gaussian process models. *Quality and Reliability Engineering International* 30 (4): 487–501. doi: 10.1002/qre.1502.

Zhang, C., H. Yan, S. Lee, and J. Shi. 2018a. Multiple profiles sensor-based monitoring and anomaly detection. *Journal of Quality Technology* 50 (4):344–62. doi: 10.1080/00224065.2018.1508275.

Zhang, C., H. Yan, S. Lee, and J. Shi. 2018b. Weakly correlated profile monitoring based on sparse multi-channel functional principal component analysis. *IISE Transactions* 50 (10):878–91. doi: 10.1080/24725854.2018.1451012.

Zhang, C., H. Yan, S. Lee, and J. Shi. 2021. Dynamic multivariate functional data modeling via sparse subspace learning. *Technometrics* 63 (3):370–83. doi: 10.1080/00401706.2020.1800516.

Zhang, C., L. Zhang, and N. Chen. 2017. Spectral network approach for multi-channel profile data analysis with applications in advanced manufacturing. In *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pp. 1709–1713. doi: 10.1109/IEEM.2017.8290183.

Zhou, T., H. Shan, A. Banerjee, and G. Sapiro. 2012. Kernelized probabilistic matrix factorization: exploiting graphs and side information. In *Proceedings of the SIAM International Conference on Data Mining, Society for Industrial and Applied Mathematics,* 403.

Zou, C., X. Ning, and F. Tsung. 2012. LASSO-based multivariate linear profile monitoring. *Annals of Operations Research* 192 (1):3–19. doi: 10.1007/s10479-010-0797-8.

Appendix A. Subspace identification for parameter initialization

Subspace identification assumes observations can be projected into a subspace which is consisted of latent states, i.e., $\boldsymbol{y}_t^{(i)} = \boldsymbol{C}\boldsymbol{x}_t^{(i)} = \boldsymbol{C}\boldsymbol{A}\boldsymbol{x}_{t-1}^{(i)} = \boldsymbol{C}\boldsymbol{A}^{t-1}\boldsymbol{x}_1^{(i)}$. Based on this, we further implement singular value decomposition (SVD) on the Hankel matrix stacked by the observations as follows. Let $\boldsymbol{U}_{1:p,\,1:q}^{(i)}$ be the upper left $q \times p$ submatrix of $\boldsymbol{U}$, $\boldsymbol{E}_{1:q,\,1:q}$ be the upper left $q \times q$ submatrix of $\boldsymbol{E}$, and $\boldsymbol{V}_{:,\,1:q}$ be the first q columns of $\boldsymbol{V}$. We initialize the observation matrix as $\boldsymbol{C}^0 = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{U}_{1:p,\,1:q}^{(i)}$ and the latent states as $(\boldsymbol{x}_1^{(i)}, \boldsymbol{x}_2^{(i)}, \ldots, \boldsymbol{x}_{T_1-2}^{(i)}) = \boldsymbol{E}_{1:q,\,1:q}\boldsymbol{V}'_{:,\,1:q}$.

$$\begin{bmatrix} \boldsymbol{y}_1^{(i)} & \boldsymbol{y}_2^{(i)} & & \boldsymbol{y}_{T_i-2}^{(i)} \\ \boldsymbol{y}_2^{(i)} & \boldsymbol{y}_3^{(i)} & \cdots & \boldsymbol{y}_{T_i-1}^{(i)} \\ \boldsymbol{y}_3^{(i)} & \boldsymbol{y}_4^{(i)} & & \boldsymbol{y}_{T_i}^{(i)} \end{bmatrix} = \boldsymbol{U}\boldsymbol{E}\boldsymbol{V}' \doteq \begin{bmatrix} \boldsymbol{C}^{(i)} \\ \boldsymbol{C}^{(i)}\boldsymbol{A} \\ \boldsymbol{C}^{(i)}\boldsymbol{A}^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1^{(i)} & \boldsymbol{x}_2^{(i)} & \cdots & \boldsymbol{x}_{T_i-2}^{(i)} \end{bmatrix}.$$

Given the estimated state variables $\boldsymbol{X}^{(i)}$, $\boldsymbol{A}^0$ can be initialized based on the least squares method, followed by the other parameters estimated via their mathematical definitions.

$$\begin{aligned} \boldsymbol{A}^0 &= \operatorname{argmin} \sum_{i=1}^{N}\sum_{t=2}^{T_i-1} \left(\boldsymbol{x}_t^{(i)} - \boldsymbol{A}\boldsymbol{x}_{t-1}^{(i)}\right)^2 \\ &= \frac{\sum_{i=1}^{N}\sum_{t=2}^{T_i-1}\left(\boldsymbol{x}_t^{(i)}\boldsymbol{x}_{t-1}^{(i)'}\right)}{\sum_{i=1}^{N}\sum_{t=2}^{T_i-1}\left(\boldsymbol{x}_{t-1}^{(i)}\boldsymbol{x}_{t-1}^{(i)'}\right)} \\ \boldsymbol{Q}^0 &= \frac{1}{N}\sum_{i=1}^{N} cov\left[\left(\boldsymbol{X}_{2:T_i}^{(i)} - \boldsymbol{A}^0\boldsymbol{X}_{1:T_i-1}^{(i)}\right)'\right] \\ \boldsymbol{R}^0 &= \frac{1}{N}\sum_{i=1}^{N} cov\left[\left(\boldsymbol{Y}_{1:T_i}^{(i)} - \boldsymbol{C}^0\boldsymbol{X}_{1:T_i}^{(i)}\right)'\right] \\ \boldsymbol{\mu}_1^0 &= \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_1^{(i)} \\ \boldsymbol{V}_1^0 &= \frac{1}{N}\sum_{i=1}^{N}\left[\left(\boldsymbol{x}_1^{(i)} - \boldsymbol{\mu}_1\right)\left(\boldsymbol{x}_1^{(i)} - \boldsymbol{\mu}_1\right)'\right] \end{aligned}$$

Appendix B. PRL_1 and TRL_1 comparison for numerical studies

Table B1. $APRL_1$ and $ATRL_1$ comparison of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA, and VPCA control charts with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (I-III) for Model (I) (numbers in parentheses are the corresponding $SDPRL_1$ and $SDTRL_1$ values).

Scenario		δ	RSSM-INPOM	SSM	VAR(1)	SMFPCA	MFPCA	MPCA	VPCA
I ($j = 1, 2; \delta \times 10^{-2}$)	APRL	0	206(201)	201(196)	195(189)	205(362)	193(332)	200(355)	201(343)
		2	125(123)	133(133)	208(203)	204(353)	189(332)	165(318)	188(333)
		3	78.4(73.6)	82.0(78.0)	184(173)	156(309)	161(306)	127(280)	160(306)
		4	44.0(44.1)	46.5(45.7)	135(131)	131(281)	157(305)	93.5(226)	146(293)
		6	13.1(12.0)	13.3(12.4)	72.3(72.1)	71.8(200)	86.5(210)	38.3(128)	86.9(208)
		8	4.58(3.61)	4.80(3.96)	36.0(34.7)	28.5(89.6)	40.1(119)	20.7(83.1)	57.0(167)
	ATRL	10	62.8(54.1)	64.4(58.6)	641(652)	726(2520)	798(2150)	325(326)	984(2840)
		12	29.1(23.9)	30.0(25.3)	312(289)	377(212)	448(297)	264(241)	686(2260)
		14	16.4(13.5)	16.9(13.2)	147(141)	313(164)	349(189)	218(185)	528(2330)
		16	10.6(8.26)	11.1(7.35)	77.6(73.9)	259(104)	285(140)	168(103)	345(1540)
		18	7.26(5.20)	7.98(5.16)	41.7(36.8)	209(79.7)	235(102)	140(87.6)	290(1320)
		20	5.15(3.74)	5.86(3.39)	25.8(21.8)	180(69.0)	198(89.8)	125(92.7)	257(1350)
II ($j = 3, 4; \delta \times 10^{-3}$)	APRL	0	202(196)	196(193)	198(198)	208(369)	193(333)	196(347)	197(340)
		5	157(164)	185(183)	199(194)	198(348)	209(353)	164(318)	182(328)
		8	103(104)	159(153)	191(182)	181(328)	172(319)	144(298)	159(304)
		10	77.2(79.5)	125(128)	167(157)	148(298)	145(269)	117(268)	152(295)
		15	34.5(32.4)	67.1(62.3)	96.4(97.1)	83.1(212)	109(249)	59.6(171)	121(261)
		20	12.6(13.0)	26.1(25.3)	41.7(42.3)	44.7(137)	62.1(166)	25.3(84.2)	77.0(196)
	ATRL	30	68.9(98.0)	131(154)	143(206)	518(874)	630(843)	409(1360)	1360(4510)
		32	46.5(69.6)	97.0(117)	80.2(135)	521(529)	654(564)	379(1270)	1170(3520)
		35	26.2(38.1)	49.8(64.8)	35.7(63.4)	421(396)	473(318)	270(1070)	1010(3320)
		40	11.9(18.6)	21.5(31.9)	17.5(24.5)	347(145)	393(270)	245(872)	799(2820)
		45	7.32(7.29)	8.97(14.0)	8.43(9.11)	311(127)	360(127)	209(220)	534(1120)
		50	5.30(5.79)	5.96(3.32)	5.42(3.60)	266(98.6)	277(140)	158(152)	485(1250)
III ($j = 5, 6; \delta \times 10^{-3}$)	APRL	0	195(196)	200(192)	199(188)	204(364)	200(338)	203(360)	196(338)
		0.5	163(158)	176(176)	181(184)	198(352)	186(335)	169(320)	184(330)
		1	83.1(80.8)	120(119)	95.6(95.0)	117(267)	120(264)	114(260)	187(339)
		1.5	36.4(35.6)	65.7(64.1)	45.2(44.3)	62.0(181)	57.5(162)	69.1(194)	169(311)
		2	14.9(14.3)	31.6(29.9)	21.9(21.2)	27.4(56.0)	29.3(67.0)	28.0(107)	136(288)
		3	3.47(2.83)	7.07(6.51)	8.20(8.01)	6.25(2.96)	6.38(4.66)	6.34(8.18)	93.4(241)
	ATRL	5	34.5(12.7)	46.1(26.1)	60.5(33.2)	94.2(34.6)	94.6(44.6)	87.0(53.3)	1870(6560)
		6	27.8(5.54)	31.4(9.96)	52.0(27.8)	71.2(24.6)	67.0(28.8)	67.1(54.0)	905(4280)
		8	22.7(4.07)	23.7(5.82)	34.8(11.9)	45.3(14.1)	45.0(14.0)	45.2(16.2)	434(2210)
		10	18.5(4.83)	19.1(3.42)	22.8(5.95)	40.9(7.95)	40.4(3.78)	41.7(11.4)	356(2270)
		15	12.5(3.12)	13.4(2.67)	15.1(3.72)	40.1(2.53)	40.1(3.79)	40.0(1.27)	208(1480)
		20	9.08(2.43)	10.2(2.14)	13.5(2.96)	40(0)	40(0)	40.1(2.09)	83.4(133)

Table B2. $APRL_1$ and $ATRL_1$ comparison of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA, and VPCA control charts with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (IV-VI) for Model (I) (numbers in parentheses are the corresponding $SDPRL_1$ and $SDTRL_1$ values).

Scenario		δ	RSSM-INPOM	SSM	VAR(1)	SMFPCA	MFPCA	MPCA	VPCA
IV: ($j = 1, 3; \delta \times 10^{-2}$)	APRL	0	193(199)	201(198)	194(185)	199(351)	200(342)	205(354)	201(344)
		12	114(111)	146(140)	124(119)	116(249)	153(299)	185(336)	197(337)
		15	77.6(78.0)	117(114)	83.1(82.4)	102(239)	131(268)	175(324)	174(321)
		18	38.4(40.0)	76.9(76.0)	49.5(48.8)	70.3(185)	101(229)	159(307)	169(314)
		24	8.06(7.47)	23.2(22.8)	13.3(12.6)	32.0(99.1)	46.1(124)	144(294)	166(309)
		30	2.41(2.12)	6.64(6.76)	4.01(3.51)	15.0(17.4)	22.5(58.1)	131(181)	141(293)
	ATRL	34	16.4(32.5)	80.2(116)	47.4(65.0)	563(1200)	676(1300)	4010(9770)	5800(11600)
		36	9.24(20.6)	44.8(68.1)	33.5(51.5)	504(1070)	837(1050)	4100(9760)	5390(11200)
		38	5.85(14.7)	25.4(43.6)	19.0(32.5)	557(670)	602(643)	3810(8940)	5960(11800)
		40	3.57(8.59)	18.8(35.5)	10.7(24.0)	499(537)	576(517)	3480(8850)	5770(11600)
		42	2.52(4.53)	9.38(22.0)	6.21(15.9)	450(352)	628(390)	3980(9260)	5440(11500)
		45	2.12(2.19)	4.26(10.6)	2.89(9.37)	484(259)	615(261)	3800(9350)	5830(11800)
V: ($j = 4, 6; \delta \times 10^{-2}$)	APRL	0	195(184)	204(201)	207(199)	197(364)	200(343)	203(363)	200(352)
		2	125(128)	135(131)	141(141)	200(356)	204(356)	175(326)	194(335)
		3	76.6(76.0)	78.4(75.8)	86.1(87.4)	184(336)	164(314)	180(335)	202(343)
		4	35.8(36.3)	41.4(40.7)	52.3(53.4)	188(344)	159(305)	177(330)	205(343)
		5	18.1(18.4)	19.8(19.9)	25.9(25.7)	146(294)	157(305)	174(331)	190(336)
		6	8.49(8.41)	10.0(10.0)	12.2(12.4)	131(277)	116(257)	164(317)	204(358)
	ATRL	9	46.8(60.1)	55.2(68.5)	72.8(79.5)	2590(7090)	2950(7190)	5750(11800)	7730(13600)
		10	26.3(35.1)	33.8(48.0)	53.5(55.3)	2210(6410)	2210(6160)	5180(11100)	7570(13300)
		11	17.8(25.9)	19.5(30.2)	35.8(44.8)	1820(5540)	1820(5080)	4680(10600)	7700(13400)
		12	11.3(18.3)	13.2(20.6)	25.3(38.6)	1680(5333)	1420(3980)	4120(9610)	7660(13600)
		14	5.22(8.84)	6.52(11.6)	15.1(19.3)	866(2390)	864(1880)	3130(8060)	7680(13300)
		16	3.09(4.65)	3.84(6.44)	8.03(8.32)	598(882)	627(1440)	2770(7480)	6260(11900)
VI ($\theta = b(1, 1); \delta \times 10^{-2}$)	APRL	0	204(200)	202(206)	204(201)	198(366)	199(355)	205(370)	204(358)
		0.1	110(113)	158(159)	143(142)	217(367)	191(335)	187(335)	214(358)
		0.2	41.7(42.4)	68.9(65.3)	86.3(81.6)	178(324)	192(331)	186(339)	192(334)
		0.3	13.6(12.6)	24.4(24.4)	44.2(44.5)	154(305)	152(300)	148(303)	159(312)
		0.4	5.58(4.90)	9.07(8.22)	22.9(22.0)	123(271)	111(254)	151(301)	137(283)
		0.5	2.81(2.25)	4.10(3.27)	10.2(9.37)	73.5(194)	84.6(210)	121(260)	104(247)
	ATRL	0.7	39.9(23.9)	50.0(34.2)	98.8(80.6)	1060(3490)	1160(3520)	3270(8440)	82.1(213)
		0.8	30.3(12.1)	34.9(17.4)	57.2(41.9)	617(1601)	602(2726)	2480(7240)	69.0(202)
		1	24.1(4.53)	25.5(6.14)	30.6(13.7)	369(237)	471(1510)	1270(4610)	47.2(150)
		2	19.6(3.61)	20.8(3.97)	23.5(6.49)	202(162)	231(752)	656(2620)	42.4(35.6)
		3	16.1(1.28)	16.9(1.33)	17.4(2.10)	103(57.6)	92.2(47.0)	239(1030)	40.6(11.8)
		5	14.6(0.79)	14.6(1.01)	13.3(1.06)	41.6(13.3)	40.6(14.8)	55.4(18.5)	40(0)

Table B3. $APRL_1$ and $ATRL_1$ comparison of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA, and VPCA control charts with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (I-III) for Model (II) (numbers in parentheses are the corresponding $SDPRL_1$ and $SDTRL_1$ values).

Scenario		δ	RSSM-INPOM	SSM	VAR(1)	SMFPCA	MFPCA	MPCA	VPCA
I ($j = 1, 2; \delta \times 10^{-3}$)	APRL	0	200(196)	197(189)	205(198)	203(259)	198(237)	200(247)	199(241)
		2	144(140)	203(199)	202(195)	201(242)	200(256)	205(258)	204(243)
		3.5	105(110)	182(170)	211(201)	188(235)	208(258)	208(258)	194(242)
		4.5	86.4(83.4)	152(153)	213(214)	204(247)	215(251)	211(255)	203(245)
		6	55.6(59.4)	107(104)	206(190)	195(237)	192(247)	210(260)	209(257)
		10	18.4(17.2)	32.8(32.6)	197(196)	156(199)	149(197)	209(258)	196(238)
	ATRL	15	474(474)	692(683)	20600(19600)	9380(13700)	9800(14300)	21300(25900)	20700(24800)
		20	152(150)	190(191)	19000(18700)	5700(7760)	6330(9290)	20100(25300)	19900(24200)
		25	63.9(54.3)	66.3(59.5)	18500(18300)	3880(6030)	4250(6960)	21200(26400)	20100(24700)
		30	35.2(20.3)	33.7(22.4)	17700(17600)	2540(2700)	2550(2980)	22500(27400)	20300(23900)
		35	25.3(11.3)	23.2(12.2)	15100(15200)	1870(2010)	1780(1860)	20700(25500)	20600(25100)
		40	19.2(7.21)	17.1(6.88)	13800(12900)	1440(1420)	1430(1400)	20200(25100)	19800(23900)
II ($j = 3, 4; \delta \times 10^{-3}$)	APRL	0	199(198)	204(201)	203(194)	203(253)	200(232)	199(250)	197(242)
		2	107(106)	157(156)	201(190)	180(234)	179(229)	193(248)	195(238)
		3	62.0(65.0)	103(98.9)	204(201)	156(196)	164(215)	200(243)	187(241)
		4	32.5(32.2)	55.0(51.6)	204(194)	148(202)	143(197)	183(233)	191(229)
		5	16.0(14.7)	29.0(28.8)	201(195)	127(181)	115(181)	189(243)	198(243)
		7	5.09(4.46)	6.96(6.40)	208(204)	87.4(147)	84.4(148)	197(244)	181(225)
	ATRL	9	127(143)	178(205)	21300(20300)	7860(15300)	6170(12700)	17800(22900)	17800(23500)
		10	75.6(91.9)	93.7(117)	20000(19400)	6660(13700)	5630(11500)	17200(23200)	16100(21700)
		12	31.2(38.0)	37.1(48.2)	20300(21000)	6060(13700)	5090(12000)	17000(23600)	15900(20700)
		14	17.5(12.4)	19.3(17.4)	22400(20300)	4460(11600)	3900(11300)	15900(23100)	14300(19800)
		16	14.3(3.33)	14.8(6.36)	21800(20700)	3960(10500)	4160(11400)	16100(23200)	13800(18800)
		18	12.6(2.61)	13.1(2.43)	21500(20300)	2700(9520)	3360(10800)	15000(22200)	12600(18100)
III ($j = 5, 6; \delta \times 10^{-3}$)	APRL	0	199(185)	201(198)	207(207)	195(239)	202(251)	205(253)	195(241)
		0.2	166(153)	178(175)	184(181)	195(251)	196(232)	206(261)	179(235)
		0.6	113(113)	121(118)	131(138)	153(205)	144(193)	169(238)	147(205)
		1	63.9(64.1)	78.5(75.2)	90.1(89.7)	111(160)	107(153)	129(209)	72.4(117)
		1.5	26.6(25.1)	38.1(37.4)	58.1(57.6)	58.4(84.4)	55.6(96.7)	90.0(173)	46.3(74.7)
		2	10.5(9.60)	21.8(19.7)	35.2(37.0)	33.0(42.8)	29.7(51.4)	58.2(134)	32.8(36.8)
	ATRL	3	144(127)	208(186)	1390(1300)	1410(1260)	1110(1150)	4300(12400)	1280(1200)
		3.5	66.4(45.2)	124(117)	970(932)	933(632)	798(653)	3460(11400)	940(630)
		4	42.7(20.4)	84.2(75.1)	679(630)	761(519)	586(438)	3210(10100)	738(567)
		4.5	31.8(11.3)	49.4(15.5)	517(480)	612(375)	493(362)	2670(9880)	574(312)
		5	26.4(7.86)	38.7(8.00)	356(309)	511(307)	390(268)	2560(8700)	344(279)
		10	12.1(2.41)	13.5(2.51)	78.9(36.4)	166(72.5)	136(58.2)	1070(7280)	109(32.8)

Table B4. $APRL_1$ and $ATRL_1$ comparison of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA, and VPCA control charts with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (IV-VI) for Model (II) (numbers in parentheses are the corresponding $SDPRL_1$ and $SDTRL_1$ values).

Scenario		δ	RSSM-INPOM	SSM	VAR(1)	SMFPCA	MFPCA	MPCA	VPCA
IV ($j = 1, 3; \delta \times 10^{-2}$)	APRL	0	196(191)	195(187)	206(197)	195(242)	196(237)	200(256)	201(245)
		6	156(155)	171(160)	208(198)	201(246)	193(244)	201(252)	192(243)
		8	85.1(81.6)	110(111)	194(182)	214(249)	197(245)	201(257)	190(231)
		10	31.6(30.6)	63.0(63.0)	202(192)	200(248)	197(253)	203(254)	202(253)
		12	10.1(9.21)	22.9(22.2)	194(189)	198(246)	209(255)	203(246)	193(234)
		14	4.42(3.72)	7.87(7.36)	191(186)	203(243)	209(249)	203(234)	203(245)
	ATRL	17	66.1(105)	146(179)	19100(19000)	19800(22900)	20900(25300)	20400(25600)	20900(26300)
		18	44.9(75.3)	92.5(123)	16000(16100)	20400(25600)	21700(25900)	19400(24300)	19700(24600)
		19	27.1(54.1)	61.0(97.3)	14400(14200)	20000(24500)	20500(25500)	19600(24200)	20000(23600)
		20	15.6(34.8)	35.4(67.7)	10300(10800)	20100(24400)	20400(24900)	20700(26100)	20600(25200)
		21	10.6(27.1)	17.5(39.2)	8900(8200)	20800(24900)	19400(23700)	20900(25800)	20400(25300)
		22	6.23(17.1)	12.2(30.6)	5610(4920)	20100(23700)	19500(25000)	20500(26100)	20600(24600)
V ($j = 4, 6; \delta \times 10^{-2}$)	APRL	0	201(193)	203(198)	201(190)	201(239)	200(257)	205(256)	198(244)
		2	119(115)	124(122)	197(190)	168(209)	151(195)	202(258)	200(252)
		3	54.5(53.5)	64.8(64.3)	198(193)	96.5(135)	85.2(113)	204(250)	205(255)
		4	18.8(18.4)	26.5(25.5)	190(179)	48.1(69.4)	40.9(52.9)	204(258)	202(248)
		5	6.31(5.86)	8.65(8.15)	175(164)	23.0(32.8)	23.4(34.6)	203(250)	206(250)
		6	2.57(2.23)	3.54(2.87)	175(171)	12.9(17.7)	13.1(16.1)	198(248)	191(231)
	ATRL	7	80.5(91.0)	130(140)	15800(15600)	838(898)	889(1130)	19700(25100)	19800(22700)
		8	47.2(56.7)	62.3(74.9)	15100(14400)	591(622)	612(811)	20500(24800)	19800(23000)
		9	27.2(33.1)	36.7(44.1)	12800(12400)	465(635)	472(693)	20300(25600)	21300(24400)
		10	15.9(20.7)	21.4(27.4)	12200(11700)	347(413)	365(410)	17500(21300)	20200(23100)
		12	7.54(9.90)	9.22(12.4)	9150(8710)	262(251)	250(244)	16800(20600)	20600(24100)
		15	3.85(5.25)	4.16(5.46)	4820(4980)	193(165)	199(159)	16500(20700)	21000(23300)
VI ($\theta = a(1, 1); \delta \times 10^{-3}$)	APRL	0	201(185)	196(188)	202(198)	201(249)	201(241)	207(264)	200(243)
		1	125(129)	176(178)	171(172)	161(215)	153(208)	157(231)	139(175)
		2	53.5(53.5)	87.5(80.9)	85.0(85.4)	80.0(113)	78.2(86.7)	89.0(167)	64.5(65.8)
		3	18.3(17.9)	35.8(34.0)	43.7(42.7)	43.6(42.2)	41.7(41.1)	51.0(121)	36.4(35.0)
		4	6.49(6.27)	11.5(11.2)	27.0(26.8)	27.5(26.2)	26.0(24.6)	40.0(132)	23.9(22.6)
		5	2.95(2.32)	4.36(3.94)	14.1(13.8)	14.8(13.5)	14.2(13.2)	29.0(110)	12.7(11.6)
	ATRL	5.5	148(167)	223(237)	481(465)	604(420)	547(435)	1260(9370)	433(341)
		6	102(115)	141(158)	400(377)	538(326)	486(393)	1340(9440)	406(324)
		7	44.8(49.5)	62.1(77.7)	227(207)	347(255)	311(233)	1240(9190)	264(187)
		8	26.8(30.6)	30.8(36.9)	92.3(78.6)	211(132)	182(114)	1130(8950)	139(64.0)
		10	14.6(3.84)	16.5(8.71)	52.4(34.4)	150(78.8)	128(57.2)	496(5250)	115(39.1)
		15	5.38(0.79)	6.09(0.85)	10.4(1.07)	100(0)	100(0)	323(4480)	100(0)

Table B5. $APRL_1$ and $ATRL_1$ comparison of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA, and VPCA control charts with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (I-III) for Model (III) (numbers in parentheses are the corresponding $SDPRL_1$ and $SDTRL_1$ values).

Scenario		δ	RSSM-INPOM	SSM	VAR(1)	SMFPCA	MFPCA	MPCA	VPCA
I ($j = 1, 2; \delta \times 10^{-2}$)	APRL	0	200(197)	198(190)	202(188)	206(200)	200(203)	199(228)	195(196)
		4	163(156)	197(189)	199(193)	162(178)	159(153)	172(206)	167(168)
		7	108(110)	154(154)	199(193)	143(141)	115(115)	124(181)	138(135)
		10	60.2(61.6)	122(119)	198(192)	89.4(93.0)	67.1(69.0)	84.8(130)	94.7(94.2)
		15	19.8(19.8)	34.4(34.5)	196(192)	42.3(44.7)	28.3(20.7)	52.2(94.6)	46.1(44.1)
		20	5.82(5.07)	6.91(6.48)	190(188)	19.3(19.9)	8.47(8.44)	28.8(68.4)	24.1(24.8)
	ATRL	25	369(327)	472(434)	36100(36100)	2750(2800)	1350(1280)	3700(7730)	2700(2840)
		30	143(114)	206(188)	34700(34800)	1460(1530)	820(731)	2840(6830)	2130(2150)
		35	78.9(51.9)	128(101)	32800(33500)	1070(1120)	665(533)	2180(6080)	1340(1330)
		40	46.8(39.5)	56.9(49.4)	30000(30600)	929(931)	556(423)	1790(5170)	1120(1150)
		25	22.4(25.8)	28.6(32.0)	27700(26800)	845(827)	490(368)	1650(4350)	822(638)
		50	9.65(13.3)	9.44(13.0)	25400(23900)	579(481)	433(327)	1190(2790)	515(519)
II ($j = 3, 4; \delta \times 10^{-2}$)	APRL	0	207(199)	201(195)	203(201)	205(203)	196(196)	196(229)	195(199)
		6	163(156)	184(175)	203(200)	99.8(93.2)	75.7(74.6)	81.5(125)	105(108)
		9	94.2(94.9)	116(116)	207(202)	45.4(45.4)	27.1(29.5)	32.9(66.8)	67.4(68.2)
		12	44.9(43.3)	56.1(53.3)	210(202)	20.4(21.2)	11.4(13.1)	20.6(35.0)	40.4(41.7)
		15	14.5(14.4)	19.8(19.6)	210(202)	8.86(7.04)	8.17(7.03)	15.6(12.3)	31.4(35.6)
		18	4.91(4.45)	5.88(4.28)	211(204)	6.01(4.56)	5.61(5.47)	10.8(18.9)	19.1(12.1)
	ATRL	23	213(167)	314(275)	40800(39600)	454(358)	586(438)	1210(1560)	3040(3080)
		25	155(97.6)	194(147)	40700(39900)	379(301)	528(424)	1180(1460)	1940(1860)
		30	102(30.1)	120(46.7)	40700(38000)	305(258)	469(331)	1120(1380)	1070(926)
		35	74.8(9.24)	85.2(20.1)	36200(34700)	249(57.8)	228(64.4)	870(1060)	628(556)
		40	59.5(5.98)	69.5(16.1)	33900(32900)	215(39.5)	205(39.1)	848(950)	324(103)
		50	26.2(14.5)	30.2(12.4)	66.0(17.5)	201(12.6)	201(10.9)	841(947)	211(59.4)
III ($j = 5, 6; \delta \times 10^{-3}$)	APRL	0	207(199)	207(200)	202(202)	201(201)	198(198)	207(226)	198(201)
		0.2	171(167)	183(177)	201(201)	207(208)	171(176)	198(224)	176(178)
		0.4	108(106)	118(117)	200(201)	171(186)	138(137)	170(196)	154(157)
		0.6	48.9(46.1)	59.2(55.2)	201(201)	126(128)	103(108)	125(140)	114(119)
		0.8	19.9(19.8)	30.4(29.6)	198(200)	80.7(81.5)	64.1(67.2)	91.9(117)	71.1(72.5)
		1	8.40(7.55)	9.37(8.44)	197(199)	52.1(53.7)	38.1(31.6)	61.0(72.7)	41.9(46.6)
	ATRL	1.8	235(140)	286(191)	37100(38000)	3300(3490)	1380(1480)	8890(11300)	1420(1490)
		2.2	157(42.1)	196(81.2)	36000(36900)	2370(2450)	1010(1030)	6890(8400)	799(810)
		3	119(22.1)	148(32.0)	31300(31200)	1080(986)	558(430)	4850(5290)	523(303)
		5	77.5(15.5)	97.3(35.6)	21500(20900)	392(280)	334(264)	2860(3350)	342(228)
		10	36.0(7.08)	46.1(17.2)	15500(14400)	266(146)	241(117)	1810(1750)	333(279)
		20	18.1(2.18)	24.1(7.11)	117(8.90)	257(145)	239(120)	515(257)	256(163)

Table B6. $APRL_1$ and $ATRL_1$ comparison of RSSM-INPOM, SSM, VAR(1), SMFPCA, MFPCA, MPCA, and VPCA control charts with $\gamma = 0.1$ and $APRL_0 = 200$ under Scenarios (IV-VI) for Model (III) (numbers in parentheses are the corresponding $SDPRL_1$ and $SDTRL_1$ values).

Scenario		δ	RSSM-INPOM	SSM	VAR(1)	SMFPCA	MFPCA	MPCA	VPCA
IV ($j = 1, 3; \delta \times 10^{-2}$)	APRL	0	204(198)	207(199)	204(202)	204(203)	199(196)	197(226)	196(196)
		18	150(148)	165(154)	196(198)	182(180)	176(177)	195(225)	194(195)
		20	108(102)	126(111)	192(194)	161(168)	158(158)	196(225)	196(197)
		23	53.8(51.9)	58.4(54.6)	168(167)	91.4(90.6)	97.3(96.6)	180(207)	185(187)
		26	21.5(21.4)	22.3(22.2)	130(124)	52.2(42.1)	47.5(36.3)	157(186)	151(153)
		30	7.11(6.50)	7.14(6.46)	73.5(71.7)	23.0(11.0)	18.8(8.19)	108(126)	102(104)
	ATRL	36	227(312)	238(317)	4000(4100)	2160(2140)	2020(1940)	9480(13800)	7230(7250)
		37	165(239)	173(244)	3210(3320)	1520(1530)	1450(1410)	8500(11000)	8250(8190)
		38	120(187)	128(194)	2580(2760)	1120(1060)	923(895)	7560(9420)	6900(6230)
		40	67.9(129)	68.3(129)	1570(1660)	818(743)	723(620)	7490(8320)	5930(5240)
		42	37.6(88.4)	38.8(88.4)	948(1060)	596(342)	522(318)	6200(7320)	4400(3750)
		45	15.0(47.9)	16.8(48.6)	420(502)	417(279)	367(319)	5940(6290)	3260(2580)
V ($j = 4, 6; \delta \times 10^{-2}$)	APRL	0	203(202)	207(199)	206(200)	205(203)	199(196)	204(228)	197(198)
		2	153(151)	156(155)	165(163)	203(205)	198(196)	195(227)	197(198)
		3	111(115)	116(118)	125(114)	181(183)	179(177)	199(222)	199(198)
		4	76.7(78.8)	87.5(87.5)	95.9(99.0)	160(162)	158(156)	198(221)	201(202)
		6	36.2(36.4)	42.6(42.6)	47.7(46.7)	117(111)	112(117)	200(231)	198(199)
		10	12.3(11.5)	12.8(12.0)	14.8(14.3)	45.5(45.2)	41.7(39.4)	172(198)	183(187)
	ATRL	17	240(247)	337(341)	476(493)	6330(6320)	6190(6120)	26200(32000)	27200(27300)
		18	172(177)	219(203)	349(362)	4870(4820)	4990(4950)	25100(29400)	27300(27700)
		20	92.2(97.0)	114(117)	177(184)	3750(3720)	3610(3580)	26900(30200)	25000(25600)
		22	55.0(65.1)	66.5(74.1)	101(111)	2560(2530)	2270(2230)	24700(28100)	23300(23500)
		25	27.1(34.1)	32.0(34.0)	48.4(57.5)	1730(1650)	1450(1410)	18600(22700)	21100(21400)
		30	12.4(15.9)	16.8(18.4)	18.0(24.6)	632(590)	577(528)	15200(18100)	17800(18000)
VI ($\theta = \zeta_1 1; \delta \times 10^{-2}$)	APRL	0	204(198)	207(199)	204(202)	200(198)	195(196)	208(224)	197(198)
		3	121(119)	167(163)	175(176)	123(127)	131(134)	188(209)	195(196)
		4	78.2(74.9)	106(105)	136(134)	84.7(83.6)	90.6(86.5)	154(179)	188(191)
		5	45.7(45.5)	63.6(63.3)	87.7(86.0)	60.1(61.5)	68.7(65.8)	112(135)	153(154)
		7	15.9(14.9)	25.7(24.7)	34.0(32.9)	31.6(32.8)	32.9(30.5)	74.6(95.1)	96.8(95.4)
		8	9.47(8.75)	14.4(13.4)	23.2(22.5)	21.4(20.5)	22.0(20.7)	58.6(71.5)	63.3(65.1)
	ATRL	12	235(210)	334(308)	1150(1050)	3200(3240)	3840(3750)	6420(10500)	7600(7940)
		13	148(122)	201(174)	752(707)	2170(2150)	2920(2860)	5510(9570)	6200(6830)
		14	104(78.7)	133(108)	467(433)	1860(1870)	1750(1770)	4250(8060)	4580(4690)
		16	51.8(45.3)	72.8(65.3)	178(144)	1220(1250)	1240(1210)	2790(5870)	3850(3860)
		18	25.9(28.5)	36.6(39.1)	90.2(56.8)	806(761)	819(798)	2100(3450)	3950(3940)
		20	13.1(12.8)	13.2(13.3)	55.5(39.0)	491(350)	551(382)	1570(2870)	2690(2580)