IISE Transactions

Taylor & Francis
Taylor & Francis Group

IISE Transactions

ISSN: 2472-5854 (Print) 2472-5862 (Online) Journal homepage: https://www.tandfonline.com/loi/uiie21

Weakly correlated profile monitoring based on sparse multi-channel functional principal component analysis

Chen Zhang, Hao Yan, Seungho Lee & Jianjun Shi

To cite this article: Chen Zhang, Hao Yan, Seungho Lee & Jianjun Shi (2018) Weakly correlated profile monitoring based on sparse multi-channel functional principal component analysis, IISE Transactions, 50:10, 878-891, DOI: 10.1080/24725854.2018.1451012

To link to this article: https://doi.org/10.1080/24725854.2018.1451012

IISE TRANSACTIONS
2018, VOL. 50, NO. 10, 878–891
https://doi.org/10.1080/24725854.2018.1451012

Taylor & Francis
Taylor & Francis Group

Weakly correlated profile monitoring based on sparse multi-channel functional principal component analysis

Chen Zhang[a], Hao Yan[b], Seungho Lee[c] and Jianjun Shi[d]

[a]Department of Industrial Engineering, Tsinghua University, Beijing, P.R. China; [b]Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ, USA; [c]Samsung Electronics, Suwon, South Korea; [d]Department of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA

ABSTRACT

Although several works have been proposed for multi-channel profile monitoring, two additional challenges are yet to be addressed: (i) how to model complex correlations of multi-channel profiles when different profiles have different features (i.e., weakly or sparsely correlated); (ii) how to efficiently detect sparse changes occurring in only a small segment of a few profiles. To fill this research gap, our contributions are twofold. First, we propose a novel Sparse Multi-channel Functional Principal Component Analysis (SMFPCA) to model multi-channel profiles. SMFPCA can not only flexibly describe the correlation structure of multiple, or even high-dimensional, profiles with distinct features, but also achieve sparse PCA scores which are easily interpretable. Second, we propose an efficient convergence-guaranteed optimization algorithm to solve SMFPCA in real time based on the block coordinate descent algorithm. Third, as the SMFPCA scores can naturally identify sparse out-of-control (OC) patterns, we use the scores to construct a monitoring scheme which provides increased sensitivity to sparse OC changes. Numerical studies together with a real case study in a manufacturing system demonstrate the effectiveness of the developed methodology.

ARTICLE HISTORY
Received 18 April 2017
Accepted 7 February 2018

KEYWORDS
Dimension reduction; EWMA; functional PCA; multi-channel profiles; sparse PCA; statistical process control

1. Introduction

Profile data are widely used in manufacturing and service systems to evaluate system performance over time or space. Thus, it is important to develop effective methods to monitor profile data and detect their changes. So far, profile monitoring in the statistical process control framework has been extensively studied in the literature. See Noorossana *et al.* (2011) for a detailed literature review. Most of the literature focuses on developing monitoring techniques for a univariate profile. These techniques can generally be divided into two categories, the one based on regressions and the one based on Functional Data Analysis (FDA):

1. For regression-type methods, they assume that profile data are related to some explanatory variables, and a regression model can be used to explain this relation. The regression coefficients can then be used to construct the monitoring scheme. Based on the complexity of the profile data, these methods can be further categorized as parametric regression-based monitoring (Mahmoud and Woodall, 2004; Jensen *et al.*, 2008; Zou *et al.*, 2012) and nonparametric regression-based monitoring (Zou *et al.*, 2008; Qiu *et al.*, 2012).
2. For FDA-based methods, they treat profile data as continuous functions and analyze the functional features by transforming the data into certain feature spaces for feature extraction, such as wavelet transformation (Zhou *et al.*, 2006; Paynabar and Jin, 2011), B-spline approximation (Chang and Yadama, 2010), etc. These extracted features are then used for monitoring purposes.

In addition to these two kinds of methods, dimension reduction techniques are also used for feature extraction and process monitoring, including Principal Component Analysis (PCA) (Yu *et al.*, 2012), and independent component analysis (Ding *et al.*, 2006).

The literature mentioned above only focuses on process monitoring of a univariate profile. However, due to advances in sensing technology, a group of sensors are often installed in a manufacturing system to simultaneously collect profile data of different process variables, from which the collected data are called multi-channel profiles. These multi-channel profile data may have complex correlation structures. We demonstrate this point using a concrete example below.

We analyze signals of 15 sensors from a manufacturing system, and denote them as S1 to S15. These sensors measure different process variables of the system, such as reactor temperatures, pressures, gas flows, electric current, environment setting parameters, etc. The detailed information about the sensor signals will be discussed in Section 5. Figure 1 shows profile data of six selected sensors. We can see that every profile is smooth to some degree. This smoothness indicates that the signals of different time points are autocorrelated. Furthermore, for some

CONTACT Hao Yan HaoYan@asu.edu
Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/uiie.
Supplemental data for this article can be accessed on the publisher's website.

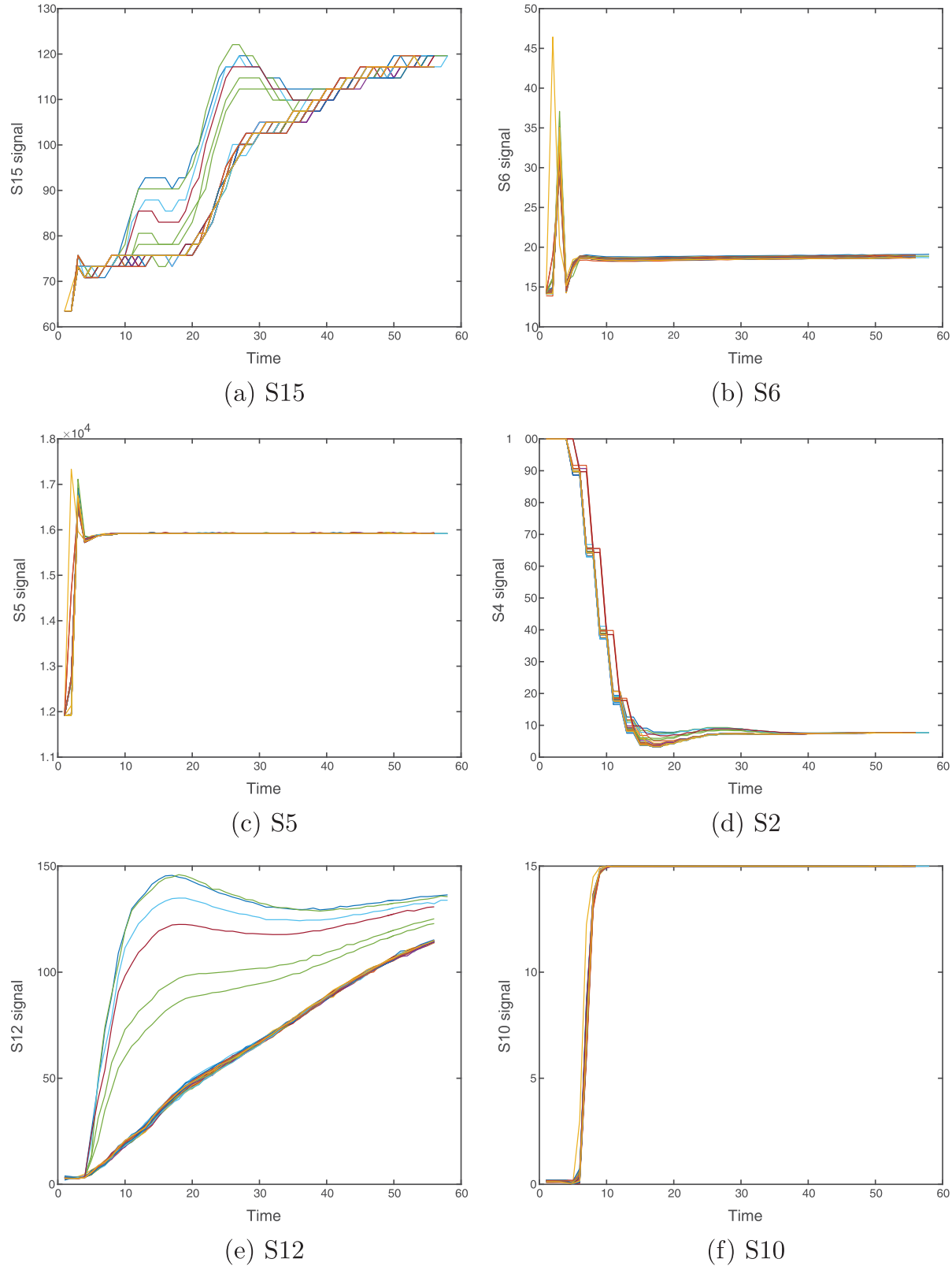


Figure 1. The profile variables over 46 samples.

sensors, they demonstrate strong inter-profile correlations with each other and share similar profile features (such as S6 and S5). It is due to these sensors are measuring physically-related process variables or they are located close to each other. However, for some other sensors, they have quite different features and weak inter-profile correlations (such as S5 and S10). This is due to these sensors being located in different places or measuring different process variables. Consequently, when the number of sensors is large, the chance that all sensors are strongly correlated is quite small. In other words, the features (variation patterns) of each sensor can only be sparsely shared by other sensors (e.g., only S6 and S5 share the same features). In this regard, a

good model should not only be able to describe the inner-profile correlations, but also be able to handle the weak inter-profile correlations of multiple sensors, especially for high-dimensional processes. Another challenge is that if a system change occurs, only a small segment of a few profiles will be influenced. As shown in Figure 1, the six out-of-control (OC) samples deviate from the in-control (IC) ones only in a small time segment of S15 and S12, with respect to a higher slope. This illustrates that in multi-channel profiles, change patterns are generally sparse, and a good monitoring scheme should be capable of detecting them.

In the above example, if we still apply univariate profile monitoring methods to monitor each profile individually and then combine their monitoring results for decision making, we will completely ignore the correlation structure of multi-channel profile data. As a result, these methods are ineffective in detecting changes of the correlation structure of different profiles. In the literature, many studies focus on jointly monitoring multi-channel profiles. In particular, for linear profiles with explanatory variables, Noorossana *et al.* (2010) proposed a control scheme based on the ordinary least square approach for Phase I monitoring. Zou *et al.* (2012) constructed a multivariate linear regression model for profiles with the LASSO penalty and used the regression coefficients for Phase II monitoring.

However, as Figure 1 shows, in most cases, multi-channel profiles are nonlinear. Then dimension reduction methods are usually adopted. We summarize some typical ones below. The simplest way is to vectorize all the profiles to a long univariate profile, and then to use the ordinary PCA on the vectorized profile for feature extraction and monitoring. This method is called Vectorized PCA (VPCA), which was first proposed by Nomikos and MacGregor (1995). However, this method sacrifices detection power, as the vectorization ignores the inter-profile correlations of multi-channel profiles. Kim *et al.* (2006) used the principal curve to extract key features or patterns from multi-channel profiles and monitored the distance between the testing sample and this principal curve. Later, other PCA techniques were introduced to directly use the original tensor representation of multi-channel profiles for monitoring. These techniques usually assume that there exist certain strong correlation structures among multi-channel profiles, and use the tensor decomposition to learn these structures. If the structure assumptions are satisfied, these methods can extract reasonable features more efficiently than VPCA, and consequently better describe the data interrelationship. In particular, Paynabar *et al.* (2013) proposed to use the Uncorrelated Multi-linear PCA (UMPCA) for fault detection in Phase I. Grasso *et al.* (2014) introduced the multi-linear PCA (MPCA) for Phase II monitoring and compared the results with VPCA. Yan *et al.* (2015) proposed several Phase II monitoring schemes based on UMPCA, MPCA, VPCA and tensor rank-one decomposition, and compared their performance for image monitoring. Recently, Paynabar *et al.* (2016) extended the Functional PCA (FPCA) to multi-channels using its proposed Multi-channel Functional PCA (MFPCA), and then constructed a change-point model for Phase I monitoring.

All the PCA-related methods mentioned above have a common limitation in that they assume each profile is a linear combination of all the extracted loadings and the multi-channel profiles are strongly correlated. Usually, this assumption

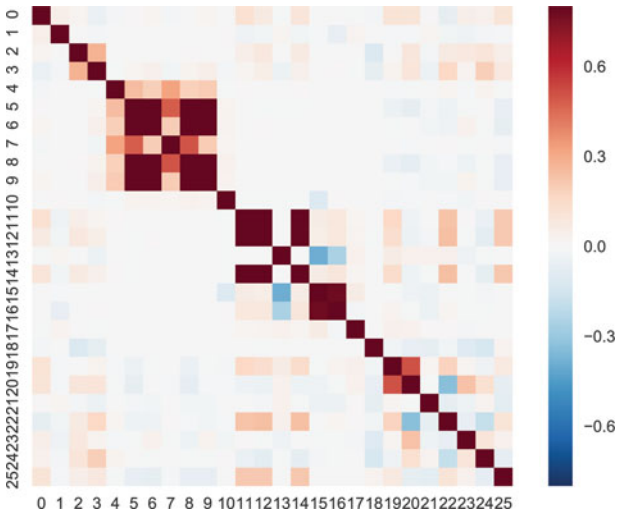


Figure 2. The correlation structure of all sensors.

cannot be satisfied. For example, Figure 2 shows the correlation structure among all the selected sensors in the manufacturing process. It can be seen that the sensors are weakly correlated, and strong correlation only exists within several sensor clusters. Therefore, the weakly correlated structure violates the strong correlation assumption of the above PCA methods, which leads to the failure of recovering the correct profile features (PCA loadings) and modeling the multi-channel profiles accurately. As a commonly acknowledged solution, incorporating thresholding or shrinkage techniques can filter out unrelated information and select granular-related features in high-dimensional data analysis. Zou *et al.* (2006) modified the traditional PCA by enforcing LASSO (elastic net) constraints on the PCA loadings. Similarly, Chen and Lei (2015) revised the FPCA by penalizing the l_1-norm of the eigen-functions to find the ones with localized support regions that explained most of the sample variance. Both methods make a trade-off between the sample variance captured by the Principal Components (PCs) and the sparsity of the PCA loadings. In this way, the achieved sparse PCA loadings can extract sparse features, filter out unrelated noise, and achieve easily interpretable results. However, in our cases when different features exist in different profiles, it is not the PCA loadings that are sparse, but the scores. As the above example shows, each feature (PCA loading) may only be shared by a few profiles, indicating that the PCA scores should be regularized.

Furthermore, the methods mentioned above tend to directly monitor all the extracted PCA scores without filtering or selection. When OC changes are sparse, these PCA scores that include IC noise may dilute OC signals and consequently compromise the detection power. With this in mind, thresholding and shrinkage techniques can also be used to smooth the noisy signals of the IC data that do not provide information about the OC state. For example, Jeong *et al.* (2006) imposed hard thresholding into the Hotelling T^2 statistic for univariate profile monitoring. Zou *et al.* (2012) applied LASSO in linear regression coefficients for multi-channel profile monitoring. Wang *et al.* (2016) adopted soft thresholding for the T^2 statistic constructed by MFPCA for multi-channel profile monitoring. However, these methods only use thresholding or shrinkage from Phase II monitoring perspective.

In summary, a unified methodology is needed to address the challenges from both perspectives, i.e., modeling weakly correlated multi-channel profiles with different features, and monitoring sparse changes. This article is targeted at these two perspectives, with the twofold contributions as below. First, inspired by the MFPCA of Paynabar *et al.* (2016) and the sparse PCA of Zou *et al.* (2006), we propose the Sparse Multi-Channel Functional PCA (SMFPCA) to extract features from multi-channel profiles. Instead of regularizing the sparsity of the PCA loadings (Zou *et al.*, 2006), the proposed SMFPCA adds the LASSO penalty on the scores. In this way, the model allows each profile to be a sparse combination of only some of the extracted loadings (features). Since the nonzero scores of different profiles can be significantly different, profiles are allowed to present different features. In this way, SMFPCA can model the weak inter-profile correlation structure. Furthermore, since the SMFPCA scores with small magnitudes, which represent noise of the IC state, are forced to be zero, the selected nonzero scores are more representative of the variation patterns of the OC state. Consequently, these scores can be used to predict sparse OC changes. Therefore, we design a monitoring scheme by projecting the profile data only to the nonzero scores to exclude IC noise, which is expected to have better detection power for sparse OC changes. Both numerical studies and a real example are presented to illustrate the advantage of the proposed methodology.

The remainder of this article is organized as follows. Section 2 introduces the proposed SMFPCA and its estimation algorithm in detail. Section 3 presents multi-channel profile monitoring scheme based on SMFPCA. Section 4 investigates the performance of the proposed monitoring scheme in detail. Section 5 applies the proposed SMFPCA and the corresponding monitoring scheme to a real-data example. Finally, Section 6 concludes this article with remarks. Some technical details are provided in the online supplement.

2. Methodology development

This section presents the SMFPCA methodology. In Subsection 2.1, we first review the original MFPCA of Paynabar *et al.* (2016), based on which, we introduce the definition of our proposed SMFPCA. Then in Subsection 2.2, we discuss the estimation algorithm of SMFPCA. We also talk about its tuning parameter selection criterion.

2.1. SMFPCA

In this subsection, we present the proposed SMFPCA in detail. To make the paper self-contained, we first introduce MFPCA proposed by Paynabar *et al.* (2016).

Assume that we have N independent and identical distributed multi-channel profile samples $\{\mathbf{Y}_i(t), i = 1, \ldots, N\}(t \in \mathcal{T})$. Each sample $\mathbf{Y}_i(t)(t \in \mathcal{T})$ contains p-channel profiles which are all square integrable random processes, denoted as $\mathbf{Y}_i(t) = [Y_{i1}(t), \ldots, Y_{ip}(t)](t \in \mathcal{T})$. Without loss of generality, we assume that $\mathcal{T} = [a, b]$, $-\infty < a < b < \infty$. Define $\mu_j(t)(t \in \mathcal{T}, j = 1, \ldots, p)$ as the mean (template) profile of the jth profile with $\boldsymbol{\mu}(t) = [\mu_1(t), \ldots, \mu_p(t)]$ $(t \in \mathcal{T})$, then we have:

$$Y_{ij}(t) = \mu_j(t) + \epsilon_{ij}(t), \quad t \in \mathcal{T}; \quad i = 1, \ldots, N; \quad j = 1, \ldots, p,$$

where $\epsilon_{ij}(t)$ is the stochastic error with $\mathrm{E}(\epsilon_{ij}(t)) = 0$.

In particular, define $\Gamma(t, s) = \mathrm{Cov}(\mathbf{Y}_i(t), \mathbf{Y}_i(s))$ and the covariance operator $(\Gamma f)(t) = \int_a^b f(s)\Gamma(t, s)ds$. Under the assumption that $\Gamma(t, s)$ is continuous over (a, b), this operator Γ has orthogonal eigen-functions, $v_k(t)(t \in \mathcal{T}, k = 1, 2, \ldots)$, with non-increasing eigenvalues λ_k, satisfying $\Gamma v_k = \lambda_k v_k$. Then according to the Karhunen–Loeve expansion, $\mathbf{Y}_i(t)$ is assumed to be represented as

$$\mathbf{Y}_i(t) = \boldsymbol{\mu}(t) + \sum_{k=1}^{\infty} v_k(t)\boldsymbol{\xi}'_{ik}, \tag{1}$$

where $\boldsymbol{\xi}_{ik} \in \mathcal{R}_{p\times 1}$ follows a p-dimensional distribution with mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Phi}_k$, and has an explicit representation that $\boldsymbol{\xi}_{ik} = \int_a^b (\mathbf{Y}_i(t) - \boldsymbol{\mu}(t))' v_k(t)dt$.

Generally, the majority of variation in the data is contained in the subspace spanned by the first few eigen-functions (i.e., PCA loadings, features or variation patterns) of Equation (1). Furthermore, in practice, every underlying sample $\mathbf{Y}_i(t)(t \in \mathcal{T})$ is usually recorded at a grid of points. In particular, here we assume that the grid points are the same for all the samples, and dense and equally spaced at $\{t_l, 1 \le l \le n\}$. Then we may reformulate Equation (1) and get the following rank-d MFPCA model:

$$\mathbf{Y}_i(t_l) = \boldsymbol{\mu}(t_l) + \sum_{k=1}^{d} v_k(t_l)\boldsymbol{\xi}'_{ik} + \mathbf{e}_i(t_l), \quad l = 1, \ldots, n. \tag{2}$$

In Equation (2), $\mathbf{Y}_i$, $\boldsymbol{\mu} \in \mathcal{R}_{n\times p}$, and $\mathbf{Y}_i(t_l)$ and $\boldsymbol{\mu}(t_l)$ are the lth row (grid point) of $\mathbf{Y}_i$ and $\boldsymbol{\mu}$, respectively; $v_k \in \mathcal{R}_{n\times 1}$, and $v_k(t_l)$ is the lth component of v_k; $\mathbf{e_i}(t_l) = [e_{i1}(t_l), \ldots, e_{ip}(t_l)] \in \mathcal{R}_{1\times p}$, and $\mathbf{e}_i(t_l)$ is independent noise with mean $\mathbf{0}$ and constant covariance matrix $\sigma^2\mathbf{I}_{p\times p}$, for $l = 1, \ldots, n$.

However, the form of Equation (1) shows that every profile is a linear combination of all the d eigen-functions $v_k(k = 1, \ldots, d)$, indicating that all the p profiles share a common set of features and their inter-profile correlations are essentially described by the correlations of $\boldsymbol{\xi}_{ik}(k = 1, \ldots, d)$. Therefore, this model is only valid when the multi-channel profiles exhibit strong correlations or share similar patterns. However, when the number of profiles p is large, multi-channel profiles are usually weakly correlated. In other words, data may come from different sources with quite different features. Then for a profile, it is not necessary to have nonzero scores on all the d features. With this in mind, it is desirable to set $\boldsymbol{\xi}_{ik}$ to be sparse to force its scores on unrelated eigen-functions to be zero. One intuitive example is when the p profiles come from s clusters, which have strong inner-cluster correlations but no inter-cluster correlations (i.e., the profiles in the same cluster have similar features and those in different clusters have different features). Therefore, directly using MFPCA for all the p profiles will achieve poor PCA loadings, which are contaminated by the mixed features of different clusters together. Consequently, the scores would also lose their meanings for monitoring. However, by adding sparsity on $\boldsymbol{\xi}_{ik}$, we represent a profile using only a few selected loadings (e.g., the loadings within the same cluster), and therefore force the algorithm to learn features of all the profiles. Consequently, the

algorithm involves all the features of different clusters together in $v_k(k = 1, \ldots, d)$, but only describes a profile using features that particularly belong to this profile's cluster. Furthermore, this sparsity also makes the extracted PCA scores easier for interpretation. With this in mind, we add the LASSO penalty on $\boldsymbol{\xi}_{ik}$ in Equation (1) and come up with SMFPCA as follows:

$$\min_{\boldsymbol{\Xi}_i, \mathbf{V}} \sum_{i=1}^{N} ||\mathbf{Y}_i - \boldsymbol{\mu} - \mathbf{V}\boldsymbol{\Xi}_i'||_F^2 + \rho \sum_{k=1}^{d} \sum_{i=1}^{N} ||\boldsymbol{\xi}_{ik}||_1, \quad (3)$$
$$\text{subject to} \quad \mathbf{V}'\mathbf{V} = \mathbf{I}_{q\times q},$$

where $\mathbf{V} = [v_1, \ldots, v_d]$, $\boldsymbol{\Xi}_i = [\boldsymbol{\xi}_{i1}, \ldots, \boldsymbol{\xi}_{id}]$, and ρ is the tuning parameter on $\boldsymbol{\xi}_{ik}(k = 1, \ldots, d)$; $||\cdot||_F^2$ is the square of the Frobenius norm of the matrix. It should be noted that here we set the same tuning parameter ρ for all the scores. However, in practice, like the traditional PCA, the magnitude of the d_1^{th} SMFPCA score is bigger than that of the d_2^{th} SMFPCA score, if $d_1 < d_2$. Therefore, it seems that ρ_{d_1} should be larger than ρ_{d_2}. However, this leads to a larger number of regularity parameters to be tuned, which may be computational intractable in practice. Furthermore, from another perspective, it is still reasonable to set a constant ρ for all the scores, with the purpose of penalizing higher-order SMFPCA scores to a greater extent. This is because the low-order SMFPCA loadings contribute more to the data variation explanation, and can be regarded as being more important than the high-order loadings. By penalizing high-order PCs more, we will focus more on low-order PCs.

It should be noted that Equation (3) is different from several SFPCA methods in Zou *et al.* (2006), Allen (2013), and Chen and Lei (2015), where it is the PCA loadings, instead of the scores, that are assumed to be sparse. Their sparsity is built on the assumption that the PCA loadings have nonzero values only in certain local regions of $\mathcal{T}$ based on some prior information, which, however, does not always hold. Instead, our method describes the data of every profile using sparsely selected loadings that are related to the profile. Our l_1- sparsity regularization makes a trade-off between signals with small PCA scores and noise. If adding a nonzero PCA score can reduce the fitting error more significantly than the penalty that the score increases, the score is more likely to represent signals and should be reserved. On the contrary, if a nonzero PCA score only reduce the fitting error slightly, it means the score is more likely to represent "insignificant" noise, and should be removed. Of course, to avoid mis-preserving noise or mis-discarding signals, we need to tune the regularization parameter ρ carefully, as introduced in the next section. In this way, our method can describe weakly correlated multi-channel profiles. The idea of SMFPCA is inspired by some other commonly used sparse dictionary learning methods (such as sparse coding (Lee *et al.*, 2007) and sparse representation (Wright *et al.*, 2009)), which aim to represent input data as linear combinations of basis elements and typically want to use as few elements as possible for every input.

2.2. Estimation of SMFPCA parameters

This section proposes an estimation algorithm for $\mathbf{V}$ and $\boldsymbol{\Xi}_i(i = 1, \ldots, N)$, to minimize the SMFPCA objective function of Equation (3). More specifically, we apply the Block Coordinate Descent (BCD) algorithm to update $\mathbf{V}$ and $\boldsymbol{\Xi}_i(i = 1, \ldots, N)$ iteratively until convergence. This algorithm can be decomposed into the following two subproblems:

1. Estimate $\boldsymbol{\Xi}_i$ given $\mathbf{V}$: For each sample i, we have:

$$\hat{\boldsymbol{\Xi}}_i = \arg\min_{\boldsymbol{\Xi}_i} ||\mathbf{Y}_i - \boldsymbol{\mu} - \mathbf{V}\boldsymbol{\Xi}_i'||_F^2 + \rho \sum_{k=1}^{d} ||\boldsymbol{\xi}_{ik}||_1. \quad (4)$$

   Let $\mathbf{Z}_i = \mathbf{V}'(\mathbf{Y}_i - \boldsymbol{\mu}) \in \mathcal{R}_{q\times p}$. Then with trivial derivation, Equation (4) can be rewritten as the exact form of the LASSO problem with the soft thresholding solution as

$$\hat{\xi}_{ijk} = \text{sgn}(Z_{ijk})(|Z_{ijk}| - \rho)^+. \quad (5)$$

   In Equation (5), Z_{ijk} is the (j, k) component of matrix $\mathbf{Z}_i$; $\hat{\xi}_{ijk}$ is the (j, k) component of matrix $\hat{\boldsymbol{\Xi}}_i$:

$$\text{sgn}(Z_{ijk}) = \begin{cases} 1 & Z_{ijk} > 0 \\ 0 & Z_{ijk} = 0. (|Z_{ijk}| - \rho)^+ \text{ is defined as} \\ & (|Z_{ijk}| - \rho)^+ = \max(0, |Z_{ijk}| - \rho) \\ -1 & Z_{ijk} < 0 \end{cases}.$$

2. Estimate $\mathbf{V}$ given $\boldsymbol{\Xi}_i$: If $\boldsymbol{\Xi}_i(i = 1, \ldots, N)$ are fixed, we can ignore the penalty part since it only depends on $\boldsymbol{\Xi}_i(i = 1, \ldots, N)$. We only try to minimize $\sum_{i=1}^{N} ||\mathbf{Y}_i - \boldsymbol{\mu} - \mathbf{V}\boldsymbol{\Xi}_i'||_F^2 = ||\mathbf{X} - \mathbf{V}\boldsymbol{\Psi}'||_F^2$ in terms of $\mathbf{V}$, subject to $\mathbf{V}'\mathbf{V} = \mathbf{I}_{q\times q}$, with $\mathbf{X} = [\mathbf{Y}_1 - \boldsymbol{\mu}, \ldots, \mathbf{Y}_N - \boldsymbol{\mu}]$ and $\boldsymbol{\Psi} = [\boldsymbol{\Xi}_1', \ldots, \boldsymbol{\Xi}_N']'$. According to the reduced rank form of the *Procrustes rotation* in Theorem 4 of Zou *et al.* (2006), we can get the solution by computing the singular value decomposition of $\mathbf{X}\boldsymbol{\Psi}$ as $\mathbf{X}\boldsymbol{\Psi} = \mathbf{U}\mathbf{D}\mathbf{W}'$, and setting $\hat{\mathbf{V}} = \mathbf{U}\mathbf{W}'$.

Proposition 1. *The BCD algorithm converges to a stationary point of Equation* (3).

Proof. The convergence of the BCD algorithm can be actually implied by the monotonic decrease of the cost function of Equation (3) during the iterations of the algorithm. In specific, in each iteration of the algorithm, step 1 and step 2 optimize the column vector $v_k(k = 1, \ldots, d)$ of $\mathbf{V}$ or $\boldsymbol{\xi}_{ik}(k = 1, \ldots, d)$ of $\boldsymbol{\Xi}_i$, with all of the others fixed, respectively. Since the objective function of Equation (3) is evidently lower bounded (> 0), the algorithm is guaranteed to be convergent. To evaluate where the algorithm converges, we can define the objective function of Equation (3) as $f(\mathbf{V}, \boldsymbol{\Xi}_1, \ldots, \boldsymbol{\Xi}_N)$. We can easily see that $f(\mathbf{V}, \boldsymbol{\Xi}_1, \ldots, \boldsymbol{\Xi}_N)$ has an unique minimum in $\mathbf{V}$ and $\boldsymbol{\Xi}_i(i = 1, \ldots, N)$ with the others fixed. Then according to Theorem 4.1 of Tseng (2001), the estimated sequence generated by the BCD method converges to a stationary point of $f(\cdot)$. ■

In this way, we can iterate these two steps until convergence to get the estimate of $\boldsymbol{\Xi}_i(i = 1, \ldots, N)$ and $\mathbf{V}$. Furthermore, when $\boldsymbol{\mu}$ is unknown in practice, we can substitute $\boldsymbol{\mu}$ by its estimate $\hat{\boldsymbol{\mu}} = \sum_{i=1}^{N} \mathbf{Y}_i/N$.

In the model, the tuning parameter ρ is used to control the sparsity of $\boldsymbol{\xi}_{ik}(k = 1, \ldots, d)$, and can be derived based on the model selection criteria of LASSO. In particular, here we adopt the Bayesian Information Criterion (BIC) and propose an iterative selection method that updates the tuning parameter in each

iteration of Equation (4). In every iteration for the updated $\mathbf{V}$, we select $\rho^\star$ by

$$\rho^\star = \min_{\rho} \sum_{i=1}^{N} ||\mathbf{Y}_i - \boldsymbol{\mu} - \mathbf{V}\boldsymbol{\Xi}_i^{\rho\prime}||_F^2 + \log(n)\hat{\sigma}^2 \sum_{i=1}^{N}\sum_{k=1}^{d} ||\boldsymbol{\xi}_{ik}^{\rho}||_0, \quad (6)$$

where $\boldsymbol{\Xi}_i^{\rho}(i = 1, \ldots, N)$ are estimated based on Equation (5) given the current ρ and $\mathbf{V}$. $\hat{\sigma}^2$ is the estimated variance of $\mathbf{e}_i$ based on the full model with $\rho = 0$, and $||\cdot||_0$ is the number of nonzero components in the vector. Algorithm 1 summarizes the estimation algorithm of SMFPCA outlined above. From many empirical studies, we can guarantee Algorithm 1 with the embedded BIC is still convergent, though its convergence rate is a bit slower than the BCD algorithm without BIC.

Algorithm 1 Estimation of SMFPCA

1. Initialization: Set $\hat{\mathbf{V}}^{(0)}$ as the first d loadings of the ordinary PCA based on $\mathbf{X}'$. $\hat{\boldsymbol{\Psi}}^{(0)} = [\hat{\boldsymbol{\Xi}}_1^{(0)\prime}, \ldots, \hat{\boldsymbol{\Xi}}_N^{(0\prime)}]$ are the corresponding PCA scores.
2. Set $k = 1$. Estimate $\hat{\boldsymbol{\Psi}}^{(k)} = [\hat{\boldsymbol{\Xi}}_1^{(k)\prime}, \ldots, \hat{\boldsymbol{\Xi}}_N^{(k)\prime}]'$ according to (5) with $\mathbf{V} = \hat{\mathbf{V}}^{(k-1)}$.
3. Compute the SVD of $\mathbf{X}\hat{\boldsymbol{\Psi}}^{(k)} = \mathbf{UDW}'$, and then update $\hat{\mathbf{V}}^{(k)} = \mathbf{UW}'$.
4. Select $\rho^\star$ based on the BIC of (6) with $\mathbf{V} = \hat{\mathbf{V}}^{(k)}$.
5. If $||\hat{\boldsymbol{\Psi}}^{(k)} - \hat{\boldsymbol{\Psi}}^{(k-1)}||_F^2 < \epsilon$ and $||\hat{\mathbf{V}}^{(k)} - \hat{\mathbf{V}}^{(k-1)}||_F^2 < \epsilon$, where ϵ is the tolerance precision, terminate the iteration; Otherwise go back to Step 2.

3. Phase II multi-channel profile monitoring

In this section, we construct a Phase II monitoring scheme for multi-channel profiles based on SMFPCA. Without loss of generality, we assume that the IC template profile $\boldsymbol{\mu}_0 = \mathbf{0}$. Suppose the SMFPCA loadings $\mathbf{V}_0$ have been estimated using m_0 reference samples $\{\mathbf{Y}_{-m_0+1}, \ldots, \mathbf{Y}_0\}$ with the corresponding scores $\{\boldsymbol{\Xi}_{-m_0+1}, \ldots, \boldsymbol{\Xi}_0\}$ in the Phase I analysis. Then for the ith on-line testing data $\mathbf{Y}_i$, we project it to $\mathbf{V}_0$, i.e., $\mathbf{Z}_i = \mathbf{V}_0'\mathbf{Y}_i$, and get its PCA scores $\boldsymbol{\Xi}_i$ according to Equation (5).

Since $\mathbf{Z}_i$ is the linear transform of $\mathbf{Y}_i$, $\mathbf{Z}_i$ directly reflects the change of $\mathbf{Y}_i$. With this in mind, we can equivalently transform monitoring the mean of $\mathbf{Y}_i$ into monitoring the mean of $\mathbf{Z}_i$. In particular, we assume that every row of $\mathbf{Z}_i$, i.e., $\mathbf{z}_{ik}$, for $k = 1, \ldots, d$, follows a p-dimensional multivariate normal distribution with mean $\mathbf{u}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. Then we formulate the sequential monitoring problem as a conventional change-point model where the mean shift occurs since sample $\tau + 1$, that is

$$\begin{cases} \mathbf{u}_k = \mathbf{u}_k^0, & \text{for } i = 1, \ldots, \tau, \\ \mathbf{u}_k = \mathbf{u}_k^1, & \text{for } i = \tau + 1, \ldots, \end{cases}$$

with $\mathbf{u}_k^1 \neq \mathbf{u}_k^0$. Without loss of generality, we assume $\mathbf{u}_k^0 = \mathbf{0}(k = 1, \ldots, d)$. As mentioned earlier, in a high-dimensional case, the probability that all features or channels change simultaneously is rather low. Instead, as mentioned in Wang and Jiang (2009) and Zou *et al.* (2012), an OC change is more likely to be caused by a hidden source that affects only one or a small set of features or channels. Thus, it is reasonable to assume that only a few components of $\mathbf{u}_k$ change to be nonzero in the OC state. Fortunately, similarly to Wang and Jiang (2009) and Zou *et al.* (2012), our sparse PCA score $\boldsymbol{\xi}_{ik}$ filters out IC noise, and correctly represents the OC direction of $\mathbf{z}_{ik}$. Therefore, $\boldsymbol{\xi}_{ik}$ is a good and convenient estimate of the changed $\mathbf{u}_k$, i.e., $\hat{\mathbf{u}}_k^1 = \boldsymbol{\xi}_{ik}$. With this in mind, we propose to construct the monitoring scheme by $\mathbf{z}_{ik}$ and $\boldsymbol{\xi}_{ik}$ based on the likelihood ratio test as

$$T_i = \sum_{k=1}^{d} \left(2\mathbf{z}_{ik}\boldsymbol{\Sigma}_k^{-1}\boldsymbol{\xi}_{ik} - \boldsymbol{\xi}_{ik}'\boldsymbol{\Sigma}_k^{-1}\boldsymbol{\xi}_{ik}\right). \quad (7)$$

The derivation of Equation (7) is shown in online supplement A. When $\boldsymbol{\Sigma}_k$ is unknown, we can estimate it as $\hat{\boldsymbol{\Sigma}}_k = \sum_{i=-m_0+1}^{0} \boldsymbol{\xi}_{ik}\boldsymbol{\xi}_{ik}'/m_0$, in the Phase I analysis. Furthermore, to handle cases with small shift magnitudes (e.g., smaller than the system's standard deviation), we integrate the monitoring scheme with the EWMA technique. In this way, we can place more emphases on the recent samples and lower emphases on the old ones. Consequently, it leads to the EWMA-based monitoring scheme. In particular, define:

$$\tilde{\mathbf{Y}}_i = (1 - \gamma)\tilde{\mathbf{Y}}_{i-1} + \gamma\mathbf{Y}_i,$$

with the initial $\tilde{\mathbf{Y}}_0 = \mathbf{0}$ and the EWMA tuning parameter $0 < \gamma < 1$. Then we have the corresponding $\tilde{\mathbf{Z}}_i = \mathbf{V}_0'\tilde{\mathbf{Y}}_i$, $\tilde{\xi}_{ijk} = \text{sgn}(\tilde{Z}_{ijk})(\tilde{Z}_{ijk} - \rho)^+$, and the final monitoring statistic as

$$T_i^{\gamma} = \frac{2 - \gamma}{\gamma[1 - (1 - \gamma)^{2i}]} \sum_{k=1}^{d} \left(2\tilde{\mathbf{z}}_{ik}\hat{\boldsymbol{\Sigma}}_k^{-1}\tilde{\boldsymbol{\xi}}_{ik} - \tilde{\boldsymbol{\xi}}_{ik}'\hat{\boldsymbol{\Sigma}}_k^{-1}\tilde{\boldsymbol{\xi}}_{ik}\right). \quad (8)$$

Correspondingly, we chose a control limit $L > 0$ for Equation (8) and define if $T_i^{\gamma} > L$, the monitoring scheme triggers an OC alarm at sample i. Specifically, given γ, m_0, and a pre-specified IC Average Run Length (ARL), we search the control limit L by simulation. In particular, we first choose an initial value for L, and then compute the IC ARL of the monitoring statistic in Equation (8) based on a large number of simulation replications (say, 10 000 in this article), where the IC samples are generated from the IC distribution of the process. If the computed IC ARL is smaller than the nominal one, we increase the value of L. Otherwise we decrease it. We repeat this process until the nominal IC ARL is achieved with a desired precision. In particular, in the search procedure, we may use some numerical search algorithms, such as the bisection search algorithm (Qiu, 2008). Due to the closed-form solution of Equation (5), the computation involved in searching L is not time-consuming. For instance, when the nominal IC ARL = 200, $\gamma = 0.1$, $m_0 = 200$ for the process with $p = 20$, $n = 50$, and the selected $d = 6$, it takes 20 minutes to complete the bisection search procedure based on 10 000 simulation replications, on a single-core personal computer. With the help of high-performance computing, the computation time can be further reduced. Hereinafter, we define the proposed monitoring scheme as the SMFPCA chart. In the SMFPCA chart, the selection of γ depends mainly on the expected

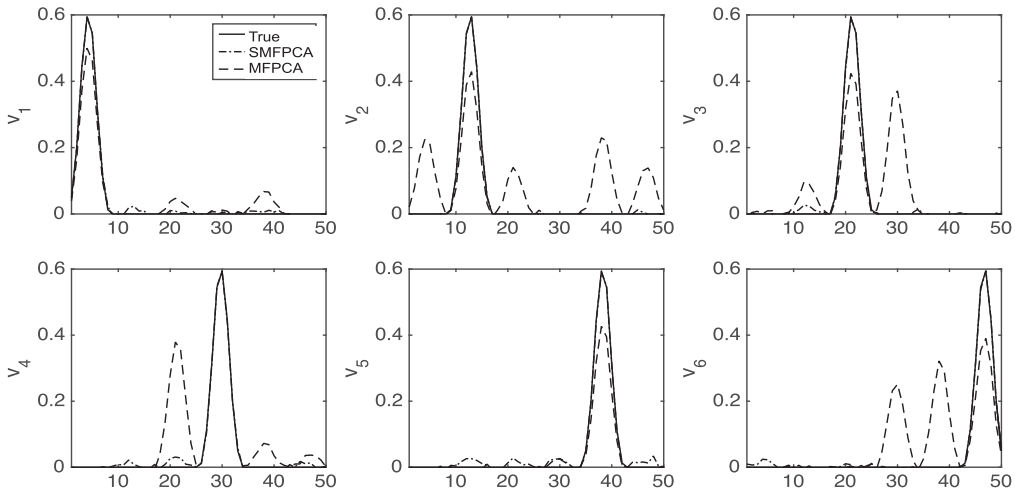
true mean shift. Similar to the traditional EWMA control charts, a large γ is good at detecting large shifts, whereas a small γ is helpful to detect small ones. Usually we set $\gamma \in [0.05, 0.2]$.

4. Numerical studies

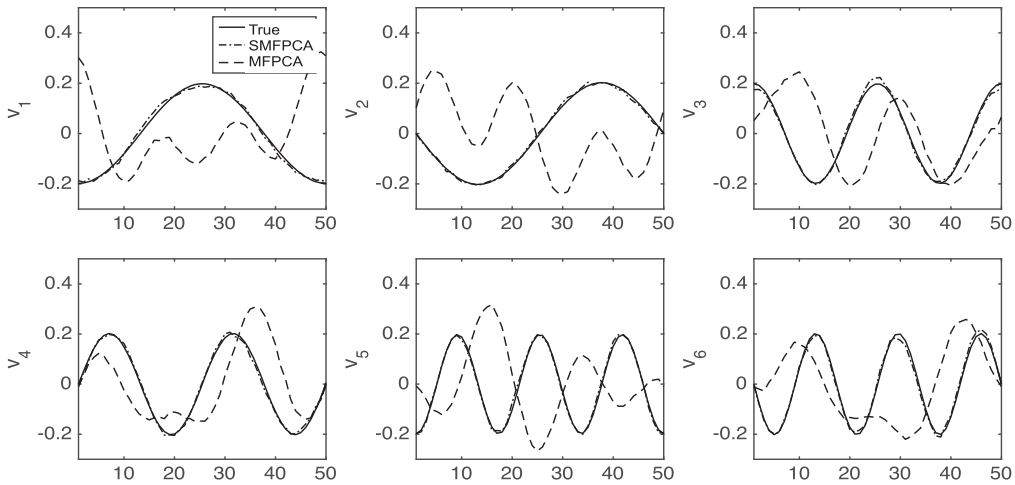
In this section, using some numerical studies, we evaluate the performance of the SMFPCA chart thoroughly and compare it with some state-of-the-art methods. In particular, we consider three different IC models as below:

1. Model (I): $\mathbf{Y}_i \in \mathcal{R}^{50\times 20}$ with $\mathbf{Y}_i = \sum_{k=1}^{6} \upsilon_k \boldsymbol{\xi}'_{ik} + \mathbf{e}_i$, where $\upsilon_k (k = 1, \ldots, 6)$ are the 1st, 4th, 7th, 10th, 13th, and 16th B-spline basis functions of order 3 with a grid of 50 equally spaced knots in $[0, 1]$. $t_l (l = 1, \ldots, n)$ with $n = 50$ are at the same locations as the knots. $\boldsymbol{\xi}_{ik} (k = 1, \ldots, 6)$ are set to be sparse vectors whose lth component is generated by $\xi_{kl} = \beta_{kl}\mathcal{I}(|\beta_{kl}| > 1.5)$ for $l = 1, \ldots, 20$, where $\mathcal{I}(\cdot)$ is the indicator function which equals one when its condition is true and equals zero otherwise. Here $\boldsymbol{\beta}_k = [\beta_{k1}, \ldots, \beta_{k20}]$ follows a 20-dimensional multivariate normal distribution with mean $\mathbf{b}_k = \mathbf{0}$ and covariance matrix $\mathbf{B}_k$. We set $(\mathbf{B}_k)_{lh} = (0.5)^{|l-h|}, l, h = 1, \ldots, 20$. $\mathbf{e}_i \in \mathcal{R}^{50\times 20}$ is noise with every component e_{ijn} following a normal distribution with mean 0 and variance $\sigma^2 = 0.04$.
2. Model (II): $\mathbf{Y}_i \in \mathcal{R}^{50\times 20}$ with $\mathbf{Y}_i = \sum_{k=1}^{6} \upsilon_k \boldsymbol{\xi}'_{ik} + \mathbf{e}_i$, where $\upsilon_k (k = 1, \ldots, 6)$ are the first six non-constant Fourier bases, i.e., $\upsilon_k = \cos(kt + k\pi)$, with a grid of 50 equally spaced sensing points $t_l (l = 1, \ldots, n)$ in $[0, 2\pi]$. $\boldsymbol{\xi}_{ik} (k = 1, \ldots, 6)$ and $\mathbf{e}_i$ are generated in the same way as Model (I).
3. Model (III): $\mathbf{Y}_i \in \mathcal{R}^{64\times 50}$ with $\mathbf{Y}_i = \sum_{k=1}^{10} \upsilon_k \boldsymbol{\xi}'_{ik} + \mathbf{e}_i$, where $\upsilon_k (k = 1, \ldots, 10)$ are the first 10 Vaidyanathan wavelet bases with a grid of 64 equally spaced sensing points in $[0, 1]$. $\boldsymbol{\xi}_{ik} (k = 1, \ldots, 10)$ are sparse vectors whose lth component is generated by $\xi_{kl} = \beta_{kl}\mathcal{I}(|\beta_{kl}| > 1.5)$ for $l = 1, \ldots, 50$. Here $\boldsymbol{\beta}_k = [\beta_{k1}, \ldots, \beta_{k50}]$ follows a 50-dimensional multivariate normal distribution with mean $\mathbf{b}_k = \mathbf{0}$ and covariance matrix $\mathbf{B}_k$. We set $(\mathbf{B}_k)_{lh} = (0.5)^{|l-h|}, l, h = 1, \ldots, 50$. $\mathbf{e}_i \in \mathcal{R}^{64\times 50}$ is generated in the same way as in Model (I).

For every model, in each simulation replication, m_0 IC samples are generated from the corresponding model and used for estimating $\mathbf{V}_0$, $\boldsymbol{\xi}_{ik} (i = -m_0 + 1, \ldots, 0, k = 1, \ldots, d)$, and the

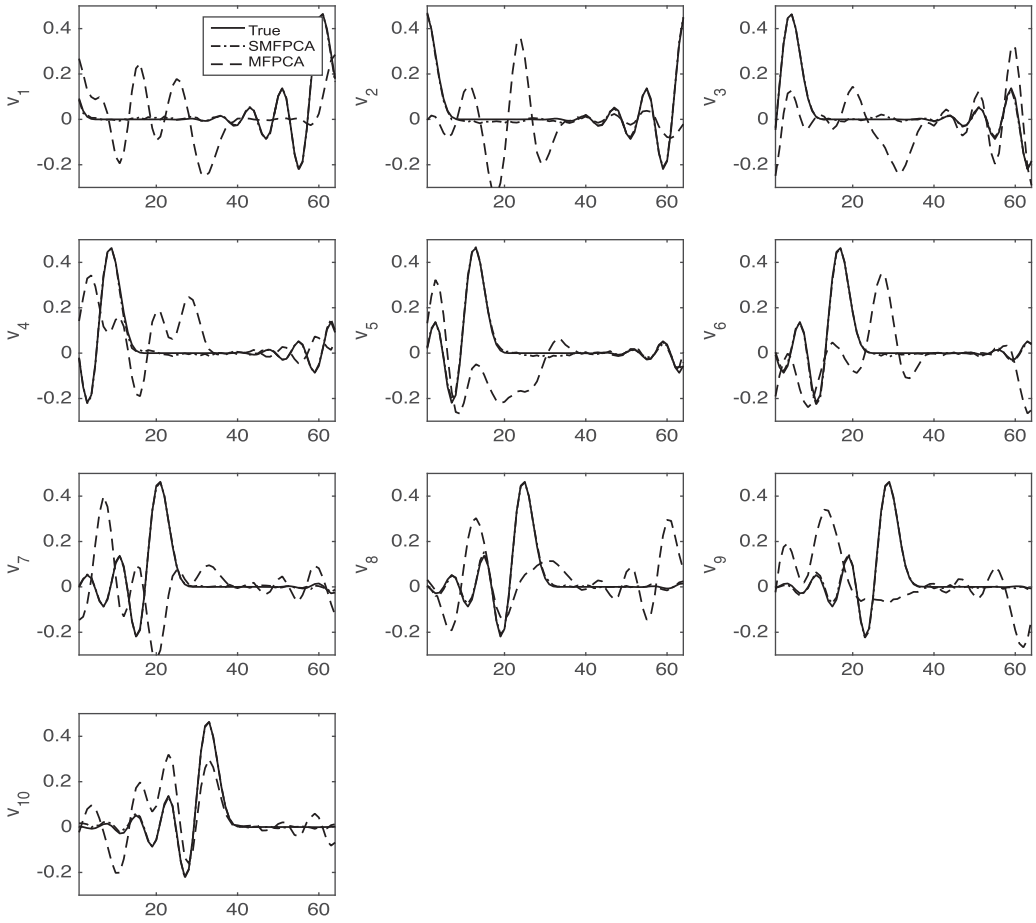


(a) The estimated PCA loadings by SMFPCA and MFPCA for Model (I).



(b) The estimated PCA loadings by SMFPCA and MFPCA for Model (II).

Figure 3. The estimated PCA loadings (features) by SMFPCA and MFPCA for the three models.



(c) The estimated PCA loadings by SMFPCA and MFPCA for Model (III).

Figure 3. (continue).

corresponding $\mathbf{\Sigma}_k (k = 1, \ldots, d)$ for the Phase I analysis. We set $m_0 = 200$ for Model (I) and Model (II), and $m_0 = 500$ for Model (III). In particular, we chose d such that in every replication the explained cumulative percentage of the sample variance by the first d loadings is 95% (other values rather than 95 can also be used, depending on the specified model accuracy and noise magnitude). To achieve this, we can first guess d by setting it as the number of PCs for which the traditional MFPCA can explain 95% of the data variation. Then we increase the value of d, and train the SMFPCA for every d, until the trained SMFPCA with a specific d can explain 95% of the data variation. Actually, in most replications, the chosen d equals the true number of eigenfunctions (i.e., $d = 6, 6, 10$ for Models (I) to (III), respectively) and the percentage of mis-specification of d in the simulation is rather low. The estimated bases $\upsilon_k (k = 1, \ldots, d)$ by SMFPCA in a certain simulation replication for these three models are shown in Figure 3. We can see that the estimated bases are quite close to (almost overlapping) the true ones, demonstrating the accuracy of the proposed estimation procedure. We also estimate and plot the bases of MFPCA (Paynabar *et al.*, 2016) for comparison in Figure 3. Unsurprisingly, MFPCA fails to extract the profile features for these three models, demonstrating its limited capability of modeling weakly correlated multi-channel profiles with different features.

For the estimated SMFPCA scores, we further evaluate their Mis-Identification Rate (MIR) and False Identification Rate (FIR). Suppose the nonzero support of the true $\boldsymbol{\xi}_{ik}$ as $\mathcal{S}(\boldsymbol{\xi}_{ik})$ and its complement set as $\mathcal{S}^c(\boldsymbol{\xi}_{ik})$. Correspondingly, we define the estimated ones as $\hat{\mathcal{S}}(\boldsymbol{\xi}_{ik})$ and $\hat{\mathcal{S}}^c(\boldsymbol{\xi}_{ik})$. Then the FIR and MIR can be defined as

$$\text{FIR} = \frac{\sum_i \sum_k \hat{\mathcal{S}}(\boldsymbol{\xi}_{ik}) \cap \mathcal{S}^c(\boldsymbol{\xi}_{ik})}{\sum_i \sum_k \mathcal{S}(\boldsymbol{\xi}_{ik})},$$

$$\text{MIR} = \frac{\sum_i \sum_k \hat{\mathcal{S}}^c(\boldsymbol{\xi}_{ik}) \cap \mathcal{S}(\boldsymbol{\xi}_{ik})}{\sum_i \sum_k \mathcal{S}^c(\boldsymbol{\xi}_{ik})}.$$

Table 1 shows the chosen optimal ρ, the FIRs, and MIRs of the three models. As observed, the chosen ρ of the three models almost ensure equivalent FIRs and MIRs.

To evaluate the Phase-II performance of the SMFPCA chart, we compare its detection power with some other monitoring schemes presented in the literature. Here we consider four charts that are particularly designed for multi-channel profile monitoring as benchmark methods: the MFPCA chart (Paynabar *et al.*, 2016), the UMPCA chart (Paynabar *et al.*, 2013), the MPCA

Table 1. The FIR and MIR of SMFPCA for Models (I) to (III).

	ρ	*FIR*	*MIR*
Model (I)	2.81	0.1633	0.1593
Model (II)	2.78	0.1652	0.1596
Model (III)	2.97	0.1621	0.1599

Table 2. OC ARL comparison in detecting mean shifts of Scenario (I) with $m_0 = 200$ for Model (I) and Model (II), and $m_0 = 500$ for Model (III) (numbers in parentheses are the run length standard deviations (SDRL)). The smallest ARLs are shown in bold.

	γ	δ	*SMFPCA*	*MFPCA*	*UMPCA*	*MPCA*	*VPCA*
Model (I)	0.05	0	195(218)	204(201)	200(224)	196(206)	202(206)
		0.25	**165**(190)	180(192)	205(206)	190(186)	180(187)
		0.75	**46.5**(41.7)	101(93.9)	191(200)	161(175)	80.4(72.5)
		1.25	**20.8**(14.3)	46.4(36.8)	172(189)	95.5(124)	31.8(23.1)
		1.75	**10.6**(6.92)	27.2(18.4)	150(171)	50.8(46.5)	18.9(12.5)
		2.75	**2.96**(1.79)	11.7(6.19)	99.5(112)	21.4(14.4)	8.21(4.71)
		3.75	**2.85**(1.40)	6.58(3.17)	72.5(94.1)	11.3(6.73)	4.84(2.52)
	0.1	0	201(218)	204(201)	200(224)	195(206)	201(205)
		0.25	**169**(188)	208(227)	197(213)	174(206)	179(180)
		0.75	**66.6**(67.5)	127(136)	193(208)	135(153)	104(103)
		1.25	**24.2**(18.8)	67.5(62.6)	174(179)	83.5(87.5)	46.7(41.6)
		1.75	**12.3**(7.60)	34.7(26.3)	154(162)	50.8(51.6)	22.8(15.8)
		2.75	**5.38**(2.98)	13.7(7.75)	116(131)	20.4(15.3)	9.82(4.99)
		3.75	**3.01**(1.42)	7.50(3.57)	81.4(94.6)	10.7(6.21)	5.52(2.40)
Model (II)	0.05	0	199(208)	197(209)	198(195)	203(219)	201(206)
		0.25	**171**(191)	191(202)	190(185)	192(208)	180(197)
		0.75	**68.3**(67.3)	120(122)	188(192)	149(172)	98.9(94.7)
		1.25	**24.7**(19.1)	63.0(56.4)	178(190)	90.0(97.4)	40.8(35.7)
		1.75	**12.2**(7.78)	33.5(26.1)	155(164)	53.1(56.0)	20.4(14.6)
		2.75	**5.27**(3.09)	13.1(7.38)	118(138)	20.7(15.6)	8.51(4.69)
		3.75	**2.98**(1.51)	7.39(3.55)	88.(109)	10.9(6.63)	5.02(2.43)
	0.1	0	205(223)	205(217)	198(195)	203(219)	200(210)
		0.25	**143**(158)	158(179)	196(205)	176(190)	163(167)
		0.75	**47.6**(41.7)	102(99.9)	140(157)	127(132)	70.6(62.5)
		1.25	**19.5**(14.6)	41.7(38.2)	78.6(79.5)	73.9(62.4)	29.0(20.5)
		1.75	**11.3**(6.84)	25.7(17.3)	48.3(42.6)	45.2(34.5)	16.7(10.6)
		2.75	**4.92**(2.66)	11.6(6.35)	23.0(18.3)	20.3(12.6)	7.77(4.23)
		3.75	**3.01**(1.40)	6.53(3.29)	13.3(10.4)	11.8(6.50)	4.57(2.22)
Model (III)	0.05	0	201(218)	200(205)	199(204)	201(198)	199(203)
		0.25	**165**(179)	188(194)	185(181)	176(165)	177(191)
		0.75	**74.8**(71.4)	126(128)	179(181)	149(139)	107(115)
		1.25	**31.7**(25.1)	68.6(60.2)	172(176)	99.1(83.5)	50.5(42.3)
		1.75	**16.7**(11.1)	36.6(27.4)	160(174)	69.1(60.6)	29.0(19.8)
		2.75	**7.16**(4.40)	16.1(9.82)	131(133)	32.5(21.8)	13.3(7.20)
		3.75	**4.30**(2.33)	9.86(5.22)	109(110)	18.2(10.0)	7.68(3.84)
	0.1	0	197(221)	199(179)	206(195)	196(198)	201(217)
		0.25	192(213)	218(214)	202(190)	**189**(191)	190(201)
		0.75	**134**(153)	165(169)	201(196)	160(162)	133(132)
		1.25	**39.4**(36.9)	99.6(93.0)	197(194)	121(118)	72.2(68.4)
		1.75	**38.9**(40.7)	54.5(46.2)	181(175)	85.9(83.7)	42.4(35.0)
		2.75	**7.62**(5.14)	21.1(13.7)	154(149)	39.1(32.0)	15.3(9.00)
		3.75	**8.21** (5.79)	11.5(5.89)	133(132)	21.1(13.8)	8.63(4.51)

chart (Grasso *et al.*, 2014), and the VPCA chart (Nomikos and MacGregor, 1995). Since some of these charts in the original literature focus only on Phase I monitoring (e.g., the UMPCA chart and the MFPCA chart), in this article, we perform some modifications to extend them to Phase II. In particular, for these four charts, we extract their PCA loadings and estimate their score distributions in Phase I using the m_0 reference data. In Phase II, when each on-line sample comes sequentially, we project the sample onto the PCA loadings and use the corresponding scores to construct the T^2 statistic with the same EWMA technique as the SMFPCA chart. Please refer to online supplement B for detailed derivations about these four alternative methods.

Since it is impossible to enumerate all change patterns for a full-scale study of the performance of all the charts, here we only chose certain representative patterns for illustration. We consider the following two OC scenarios for all the three models:

1. Scenario (I): Mean shift in the 4th, 8th, 12th, 16th and 20th components of $\mathbf{b}_1$, i.e., $b_{1l} = \delta$ with $l = 4, 8, 12, 16, 20$. This indicates the OC change pattern occurs in the $l = 4, 6, 12, 16, 20$ profiles, which is caused by the change of the first PCA score $\boldsymbol{\xi}_{i1}$.
2. Scenario (II): Mean shift in the first component of $\mathbf{b}_k (k = 1, \ldots, 5)$, i.e., $b_{k1} = \delta$, with $k = 1, \ldots, 5$. This indicates the OC change pattern only occurs in the first profile. It is caused by the change of the first five PCA scores $\boldsymbol{\xi}_{ik} (k = 1, \ldots, 5)$.

We set the change point $\tau = 25$ for all the cases, and compare the OC performance of the five competing charts in terms of steady state ARL. This means that any series (replication) where an OC signal occurs before the true change point τ is discarded. Moreover, two values of γ, 0.05 and 0.1, are considered. For every IC model with a particular γ, we find its control limit via simulation to ensure that the IC ARL equals 200. Their OC ARLs are shown in Tables 2 and 3.

The presented results can be summarized as the following points:

1. The proposed SMFPCA chart has higher efficiency than the other four charts, since SMFPCA can correctly model the weak inter-profile correlation structure of multi-channel profiles.
2. The VPCA chart performs the second best, because it breaks the original multi-channel profile structure and

Table 3. OC ARL comparison in detecting mean shifts of Scenario (II) with $m_0 = 200$ for Model (I) and Model (II), and $m_0 = 500$ for Model (III) (numbers in parentheses are SDRL values). The smallest ARLs are shown in bold.

	γ	δ	*SMFPCA*	*MFPCA*	*UMPCA*	*MPCA*	*VPCA*
Model (I)	0.05	0	195(218)	204(201)	200(224)	196(206)	202(206)
		0.25	**157**(171)	192(200)	178(196)	207(238)	170(178)
		0.75	**44.1**(43.9)	97.6(97.3)	125(132)	156(175)	75.0(70.0)
		1.25	**17.7**(12.5)	44.9(35.5)	72.7(73.7)	96.5(105)	35.4(25.8)
		1.75	**9.51**(6.00)	24.2(15.8)	44.3(40.7)	56.0(57.0)	18.4(11.4)
		2.75	**4.40**(2.46)	10.8(6.03)	21.1(17.2)	23.8(16.9)	8.46(4.64)
		3.75	**2.75**(1.31)	6.17(3.00)	12.4(8.44)	12.4(7.65)	4.96(2.49)
	0.1	0	201(218)	204(201)	200(224)	195(206)	201(205)
		0.25	**172**(184)	197(200)	191(189)	186(203)	191(195)
		0.75	**68.6**(67.4)	122(118)	163(172)	144(151)	110(108)
		1.25	**22.8**(18.3)	62.7(58.8)	109(118)	93.2(101)	45.5(38.8)
		1.75	**11.5**(7.42)	33.4(25.6)	65.7(71.4)	54.3(62.4)	24.3(16.9)
		2.75	**5.03**(2.86)	12.8(6.87)	28.6(26.9)	22.6(18.2)	10.1(5.20)
		3.75	**2.91**(1.45)	7.25(3.40)	15.3(16.3)	12.1(7.26)	5.64(2.71)
Model (II)	0.05	0	199(215)	197(209)	198(195)	203(219)	201(206)
		0.25	**158**(170)	181(196)	177(191)	175(189)	168(192)
		0.75	**44.6**(38.8)	95.1(94.7)	126(129)	127(134)	73.3(70.9)
		1.25	**15.7**(10.4)	43.9(35.3)	78.2(90.6)	78.1(77.4)	34.3(27.8)
		1.75	**9.96**(6.08)	24.1(15.4)	46.0(42.1)	45.9(37.1)	17.9(11.9)
		2.75	**4.69**(0.57)	11.1(5.94)	22.9(19.5)	20.1(12.8)	8.33(4.44)
		3.75	**2.74**(1.32)	6.28(3.04)	13.2(11.3)	11.6(6.51)	4.92(2.45)
	0.1	0	205(223)	205(217)	198(195)	203(219)	200(221)
		0.25	**163**(176)	205(205)	204(229)	203(227)	181(202)
		0.75	**61.7**(61.7)	129(129)	150(160)	155(165)	106(112)
		1.25	**22.7**(17.5)	61.8(61.8)	98.9(108)	97.6(100)	47.0(43.3)
		1.75	**11.8**(8.15)	31.2(31.2)	59.5(61.2)	59.1(56.3)	23.2(16.0)
		2.75	**4.80**(2.73)	12.7(12.7)	25.3(19.3)	22.7(17.7)	9.70(5.47)
		3.75	**2.81**(1.39)	7.11(7.11)	14.2(10.2)	12.2(6.87)	5.33(2.60)
Model (III)	0.05	0	201(218)	200(205)	199(204)	201(198)	199(203)
		0.25	**177**(202)	194(194)	203(204)	201(187)	175(189)
		0.75	**66.8**(66.0)	121(116)	169(174)	181(175)	104(103)
		1.25	**27.2**(22.9)	63.0(54.4)	123(136)	146(144)	55.0(47.1)
		1.75	**14.4**(10.5)	34.6(24.4)	80.8(84.6)	102.(93.2)	29.8(22.3)
		2.75	**6.44**(3.95)	14.8(8.40)	40.0(34.5)	53.1(38.4)	13.6(7.90)
		3.75	**3.75**(2.00)	8.78(4.47)	23.9(21.2)	31.9(20.9)	7.94(4.24)
	0.1	0	197(221)	199(179)	206(195)	196(198)	201(217)
		0.25	**169**(191)	193(178)	211(201)	217(207)	204(197)
		0.75	**87.5**(99.0)	148(132)	182(173)	196(189)	143(132)
		1.25	**37.1**(37.0)	88.7(78.7)	138(134)	172(165)	87.0(83.2)
		1.75	**17.0**(13.8)	49.5(40.4)	107(108)	144.4(144	46.2(42.2)
		2.75	**6.85**(4.72)	18.6(11.3)	53.7(60.4)	76.0(62.7)	17.0(10.2)
		3.75	**3.65**(2.16)	10.0(4.99)	28.5(25.7)	42.9(31.4)	9.22(4.55)

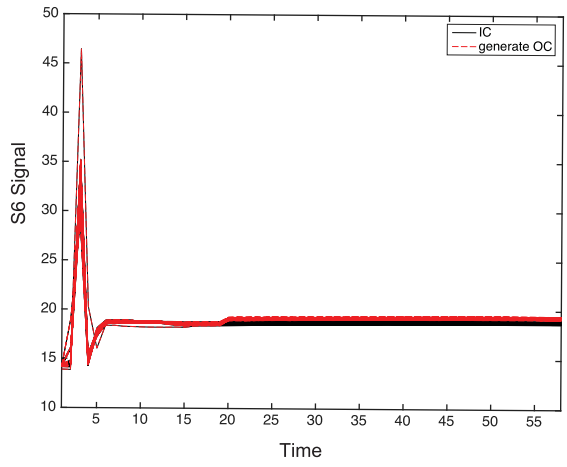
consequently fails to model the inter-profile correlation structure.

3. The other three PCA charts (the MPCA, MFPCA, and UMPCA charts) have poorer detection power than the VPCA chart. This makes sense since either the inner-profile or the inter-profile correlation structures assumed by these three PCA charts are not satisfied in the data with different features. As a result, they cannot extract correct PCA loadings or scores, and therefore loose detection power.
4. The UMPCA chart has the worst performance, due to its inherent limitation that the number of PCs that can be extracted is no more than $\min\{p, n, m_0\}$ (Paynabar *et al.*, 2013; Yan *et al.*, 2015). As a result, the explained data variance by UMPCA is very limited, resulting in its poor performance for modeling and detection. For example, in the first two models with $p = 20$, $n = 50$, and $m_0 = 200$, the number of PCs is limited to 20.
5. For the same change magnitude, the performance of all the charts in Model (I) and Model (II) is generally better than that in Model (III). This is because the detection power of these five charts for the proposed hypothesis test depends not only on the Mahalanobis distance between $\mathbf{u}_k^0$ and $\mathbf{u}_k^1$, i.e., $D = \sum_{k=1}^{d}(\mathbf{u}_k^1 - \mathbf{u}_k^0)\boldsymbol{\Sigma}_k^{-1}(\mathbf{u}_k^1 - \mathbf{u}_k^0)$, but also on p in the sense that for the same D, a higher p indicates larger IC noise, smaller equivalent shift magnitude, and consequently smaller detection power. Therefore in our case, Model (III) with a higher $p = 50$ has a smaller detection power than Model (I) and Model (II). In the online supplement, we provide more simulation results to illustrate the influence of the number of channels p on the detection power.
6. Consistent with the performance of general EWMA-type charts, the smaller shift magnitude in Model (III) makes the charts with a smaller $\gamma = 0.05$ have a better detection power than those with $\gamma = 0.1$. On the contrary, for Model (I) and Model (II), the charts with a larger $\gamma = 0.1$ are preferred.

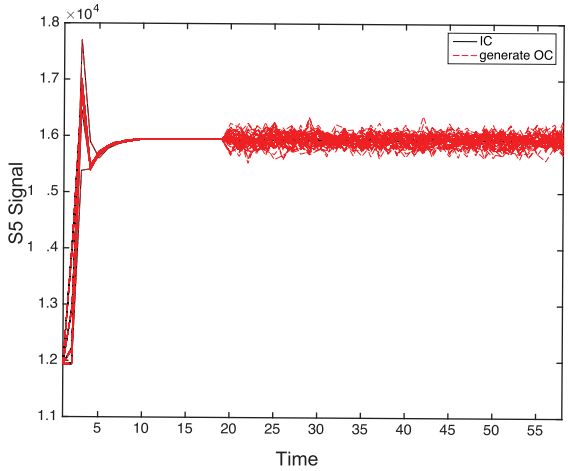
Additional simulation results with other IC and OC settings also demonstrate the above conclusions. Due to space limitations, these results are not shown here; however, they are available on request.

5. Case studies

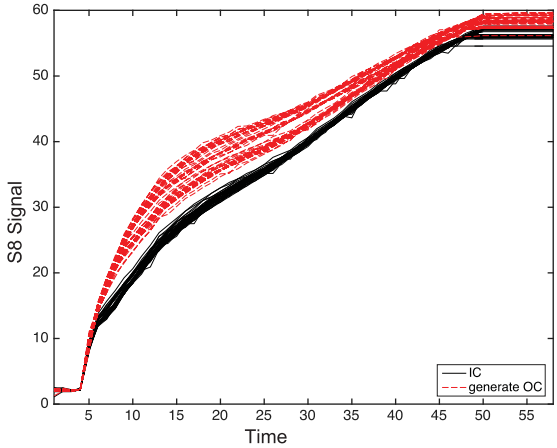
In this section, we use real data from a manufacturing system to illustrate the application of the SMFPCA chart. In this manufacturing system, many sensors are installed in the



(a) Mixed shift: mean shift of S6 with $\delta = 5$



(b) Mixed shift: fluctuation of S5 with $\delta = 5$



(c) Slope shift of S8 with $\delta = 0.2$

Figure 4. Illustration of the surrogate OC patterns.

reaction chamber to measure key process variables of the fabrication process, such as temperature, movement, electricity, pressure, etc. Each sensor generates high-dimensional functional data within each part fabrication cycle. Furthermore, those sensing signals produce complex patterns, due to different product-to-product variations and a complex self-controlled mechanism. Additional complexities in those functional data include unsynchronized signal characteristics in terms of timing, long-term natural drifts, and inherent changes in local segments (Zhang *et al.*, 2018). Depending on the nature of the product design and machine conditions, each variation pattern is associated with only a subset of sensors (e.g., weak correlations of multi-channel signals). Similarly, in Phase II monitoring, if a system change occurs, it is unlikely that it will simultaneously influence all those variation patterns. Please refer to Zhang *et al.* (2018) for more details about sensing data characteristics of this manufacturing system.

In this case study, we select 15 sensors that monitor 15 process variables during the fabrication of every product sample. These sensor variables are denoted as S1 to S15. For every variable, we measure its signal every 0.1 seconds. We have 46 IC samples and six OC samples (with a slope shift in S1, as mentioned in Section 1). Figure 1 illustrates the profiles of the six selected sensor variables as introduced in Section 1. It should be noted that for different samples, their profile length n is different, due to the inherent fluctuations in the manufacturing system. Therefore, we first remove this non-synchronization effect for different samples using the Dynamic Time Warping (DTW) algorithm and set their length to be equal to $n = 58$. The DTW algorithm is shown in detail in online supplement C. Then we use the data to demonstrate the SMFPCA chart as below. We generate some OC patterns with different shift magnitudes from the 46 IC samples as surrogate OC samples for testing. These surrogate OC patterns are designed by the engineers based on the true anomaly patterns in the manufacturing process. In particular, We consider two types of OC shift patterns:

1. Mixed shift in sensors $j \in \{1, 2, 3, 4, 5, 6, 12, 13\}$: fluctuation (random mean shift) in sensors $j \in \{1, 3, 5, 12, 13\}$ in the time interval $t \in [20, 58]$ of magnitude $d_{ij}(t)$ for the ith OC sample, i.e., $\mu_{ij}^1(t) = \mu_j^0(t) + d_{ij}(t)$, where $d_{ij}(t) \sim \mathrm{N}(0, (\delta \int_{20}^{58} s_j(t)/38\mathrm{d}t)^2)$ and $s_j(t)$ is the standard deviation of $Y_j(t)$ and can be estimated from the IC samples; constant mean shift in sensors $j \in \{2, 4, 6\}$ in the time interval $t \in [20, 58]$ of magnitude $\delta \cdot (\int_{20}^{58} s_j(t)/38\mathrm{d}t)$, i.e., $\mu_j^1(t) = \mu_j^0(t) + \delta \cdot (\int_{20}^{58} s_j(t)/38\mathrm{d}t)$.
2. Slope shift in sensors $j \in \{8, 9, 11\}$ of magnitude $d_j(t)$ in the entire time interval $t \in (0, 58]$, i.e., $\mu_j^1(t) = \mu_j^0(t) + \delta \cdot d_j(t)$ where $d_j(t)$ is a pre-specified slope change pattern for sensor j at time point t. Here we set $d_j(t) = \sum_{i=1}^{6} Y_{ij}^{\mathrm{OC}}(t)/6 - \sum_{i=1}^{46} Y_{ij}^{\mathrm{IC}}(t)/46$ as the average sample difference between the true OC profile signals with slope shifts and the IC profile signals.

Illustrations of these two OC patterns are shown in Figure 4.

We set $\gamma = 0.05$, $\tau = 50$, and $m_0 = 100$. We select $\rho = 5$ based on the BIC and tune the control limit to ensure that the IC ARL equals 200. In every simulation replication, we run the chart as follows. We randomly draw $m_0 = 100$ IC samples with replacement as reference samples from the 46 IC samples. Since we assume that the process is IC until sample τ, we draw the first τ on-line testing samples from the 46 IC samples with replacement. Then we draw the subsequent on-line samples from the surrogated OC samples with a certain δ with replacement. The chart runs until an OC signal is triggered, and the corresponding run length is recorded. For illustration purpose, Figure 5 shows the estimated eigen-functions of SMFPCA and MFPCA in one simulation replication. It is clear to see that these extracted eigen-functions of SMPCA represent different features of the data (after removing the mean curve as shown in Figure 6). In contrast, the eigen-functions of MFPCA mix features from different profiles together. For example, the "spike

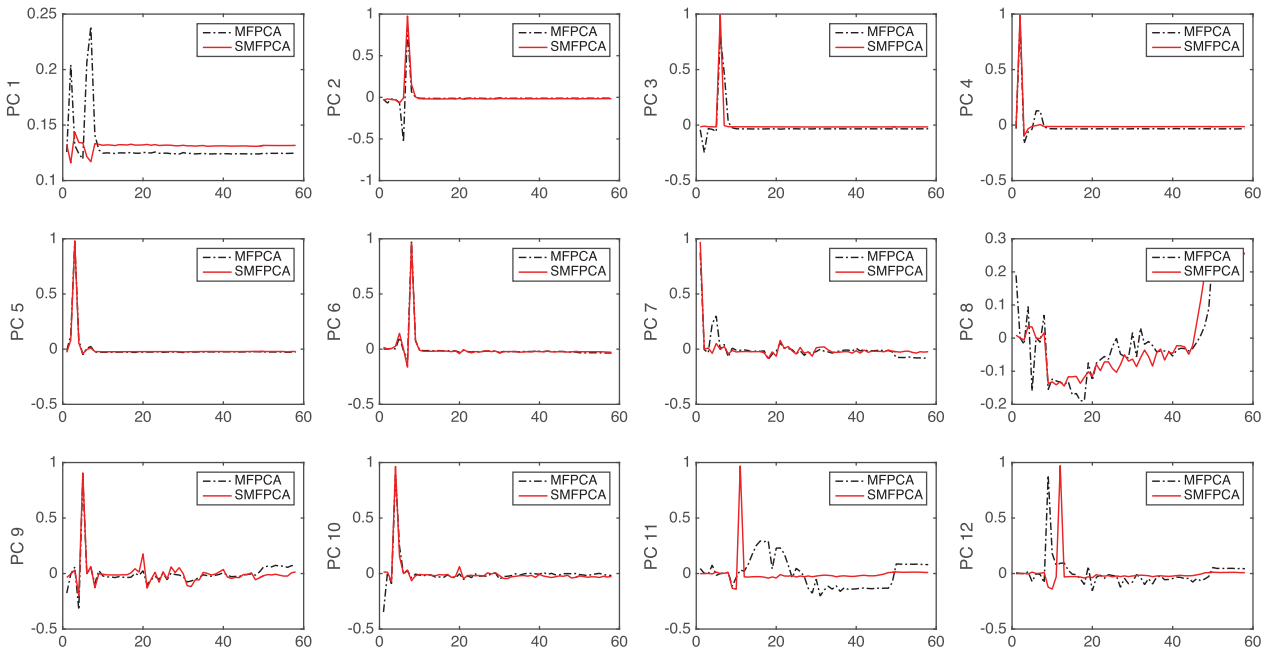


Figure 5. Estimated PCA loadings for the case study.

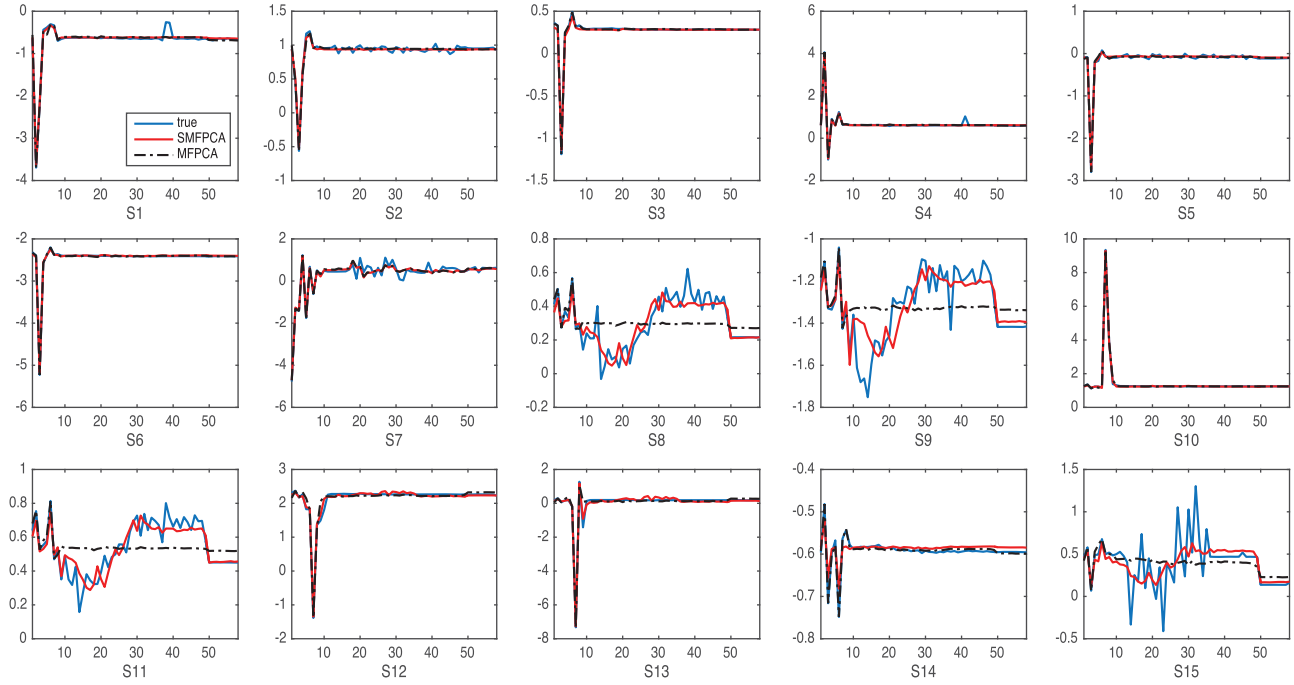


Figure 6. Profile description for the case study.

features" are very common and show up in different locations in the process. These features are well captured by PC2–PC7, PC9–PC12 in SMFPCA. However, MFPCA mixes spikes at different locations together (such as PC2–PC4). Figure 6 draws the fitted profiles based on the first 15 eigen-functions of SMFPCA and MFPCA. SMFPCA can describe the various features of different profiles accurately, whereas the MFPCA fails in some cases (such as S8, S9, S11). This is because the features of these profiles, such as the PC8 in Figure 5, are contaminated by the features of other profiles. Consequently, the final fitting is poor.

Figure 7 shows one replication of the SMFPCA chart, where the dash red line represents the control limit and the blue curve connecting with stars represents the monitoring statistics. When the process is IC as Figure 7(a) shows, the SMFPCA chart is always below the control limit, confirming the stability of the chart. When the process is OC with a process change happening after sample $\tau = 50$, as Figure 7(b) shows, the SMFPCA chart has a quick response to the shift with a timely increase in the monitoring statistic. Finally, the statistic exceeds the control limit at sample $i = 60$, signaling an OC alarm with run length $i - \tau = 10$. To better test the performance of the SMFPCA chart, similar to Section 4, we compare its performance with other five charts in terms of OC ARL in Table 4, where the control limits of all the charts are tuned to ensure that their IC ARL equals 200. The MFPCA, UMPCA, MPCA, and VPCA charts are the same as those in Section 4. Furthermore, we consider the multi-FPCA chart for comparison. The multi-FPCA chart applies FPCA for every individual channel (sensor) separately and constructs p local charts using their FPCA scores. Then these local monitoring statistics are summed together as the final monitoring statistic. It shows that except for SMFPCA,

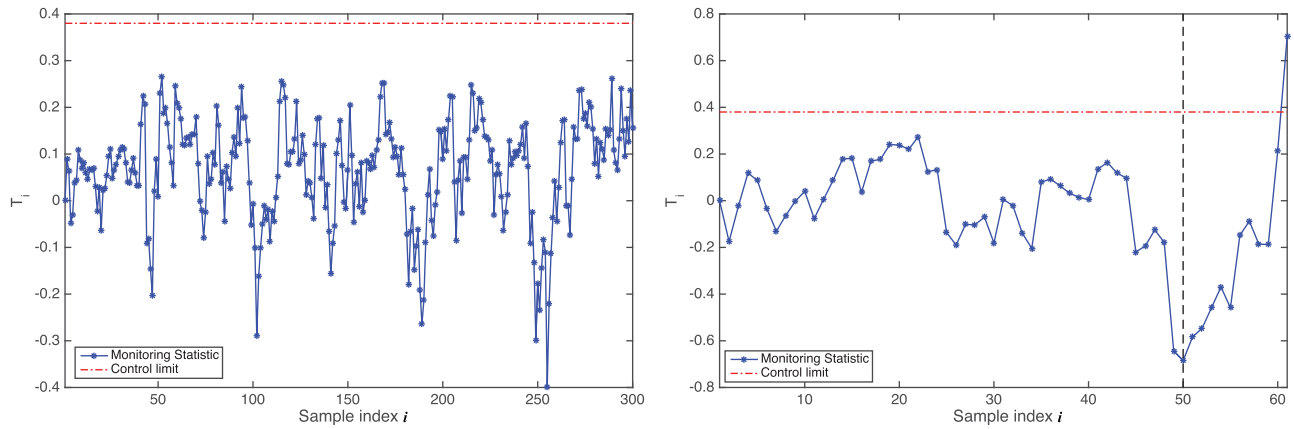


Figure 7. The monitoring statistics of the SMFPCA chart with $\rho = 5$, $\gamma = 0.05$, and $m_0 = 100$ in one simulation replication: (a) the process is IC; and (b) the process is IC for the first $\tau = 50$ samples, and then goes OC after sample 51.

Table 4. The ARL of different charts for monitoring a manufacturing process (numbers in parentheses are SDRL values).

($\gamma = 0.05$)	δ	*SMFPCA*	*MFPCA*	*UMPCA*	*MPCA*	*VPCA*	*multi-FPCA*
IC	0	200(152)	198(144)	198(176)	200(196)	201(181)	201(181)
Mixed shift	0.5	**53.9**(56.5)	171(136)	188(171)	218(198)	184(177)	206(179)
	1	**10.7**(6.51)	110(106)	204(181)	169(174)	181(166)	187(168)
	1.5	**4.63**(1.87)	52.7(51.7)	182(168)	118(141)	153(152)	159(155)
	2	**3.22**(0.98)	30.7(24.3)	180(173)	66.4(78.2)	110(121)	116(122)
	2.5	**2.45**(0.61)	18.7(13.4)	183(169)	37.0(34.3)	72.8(84.1)	63.4(49.5)
	3	**2.18**(0.40)	13.9(7.34)	177(171)	20.6(12.7)	50.5(54.8)	39.6(38.1)
Slope shift	0.01	**69.7**(97.1)	143(124)	209(180)	222(201)	188(176)	66.3(60.1)
	0.02	**30.6**(65.8)	53.7(56.3)	207(179)	203(193)	174(163)	29.51(24.63)
	0.03	**10.0**(14.6)	22.9(16.8)	207(180)	207(196)	161(158)	11.87(10.62)
	0.04	**5.72**(5.09)	15.3(9.77)	188(168)	199(196)	108(117)	5.53(4.96)
	0.05	**3.76**(2.41)	9.44(4.35)	193(176)	197(188)	84.1(99.8)	2.99(2.87)

the other five charts have quite unsatisfactory detection powers. In particular, for the mixed shift, the MFPCA chart performs second best. This is due to it having the same model structure as SMFPCA. Although MFPCA does not consider the weak correlations (feature diversity) of different sensors, its extracted PCA loadings can still capture certain sensor features to some degree. Consequently, the chart has an acceptable, but not desirable, detection power. As for the other four charts, since their models fail to describe the data structure, they have poor detection powers. For the slope shift, the multi-FPCA chart has a similar performance to that of the SMFPCA chart. This is because for the three channels with slope shifts, they have such similar features with each other that learning the eigen-functions by FPCA for every channel separately results in values that are very close to those learned by SMFPCA with the three channels jointly. In other words, these three channels can be regarded as one channel. In this case, the detection power of joint monitoring will be similar to that of separate monitoring. Furthermore, it should be noted that the UMPCA chart performs the worst, due to its inherent drawback that only limited features can be extracted. In conclusion, these benchmark methods fail to extract different features from different profiles to some degree, and consequently fail to detect changes in the features.

6. Concluding remarks

Although profile monitoring has been extensively studied in the literature, the challenges associated with designing monitoring schemes for weakly correlated multi-channel profiles with sparse changes have not yet been be addressed. This article proposes a SMFPCA-based monitoring scheme to fill this research gap. In particular, we propose SMFPCA by adding the LASSO penalty on MFPCA scores. In this way, SMFPCA allows every profile to be a sparse combination of the extracted loadings, and therefore, is capable of modeling weakly correlated multi-channel profiles with different features. Furthermore, this sparsity makes the extracted PCs easy to interpret. Moreover, since the sparsity penalty sets the PCA scores with small magnitudes to be zero, it filters out IC noise. Consequently, the nonzero scores naturally indicate the potential sparse OC directions to some degree. With this in mind, we construct a monitoring scheme by projecting the profile data to these nonzero PCA scores. In this way, the scheme automatically takes into account the unknown OC directions and is particularly designed for sparse OC changes. Numerical studies, as well as a case study, demonstrate the effectiveness and applicability of the proposed methodology.

Along with this research direction, there are several potential valuable extensions. First, we could consider including a penalty for the smoothness of the extracted PCA loadings to make them more interpretable. Second, more efficient algorithms could be explored to consider jointly estimating the SMFPCA loadings and scores. Finally, the proposed modeling techniques could be readily extended to Phase I monitoring as well.

Acknowledgement

The authors are grateful to the valuable comments provided by the editors and referees.

Funding

This project is partially supported by the NSF grant CCF-1740776.

Notes on contributors

Chen Zhang received her B.Eng. degree in electronic science and technology (optics) from Tianjin University in 2012, and her Ph.D. degree in industrial systems engineering & management from National University of Singapore in 2017. Currently, she is an assistant professor at the Department of Industrial Engineering, Tsinghua University. Her research interests include developing new approaches for modeling and monitoring of engineering systems with complex data. She is a member of IISE and INFORMS.

Hao Yan received a B.S. degree in physics from Peking University, Beijing, China, in 2011. He also received an M.S. degree in statistics, an M.S. degree in computational science and engineering, and a Ph.D. degree in industrial engineering from Georgia Institute of Technology, Atlanta, in 2015, 2016, 2017, respectively. Currently, he is an assistant professor in the School of Computing, Informatics, and Decision Systems Engineering at Arizona State University. His research interests focus on scalable machine learning algorithms for large-scale high-dimensional data with complex heterogeneous data structure to extract information or useful features for the purpose of system performance assessment, anomaly detection, intelligent sampling and decision making. He is a member of INFORMS and IISE.

Seungho Lee received a B.Eng. degree in industrial engineering from Korea University, an M.S. degrees in industrial & systems engineering from Texas A&M University, and a Ph.D. in systems & industrial engineering from the University of Arizona in 2009. Currently, he is the Principle Engineer in Samsung Electronics. His research areas include multivariable anomaly detection, multistage process analysis, and predictive maintenance.

Jianjun Shi received B.S. and M.S. degrees in electrical engineering from the Beijing Institute of Technology, Beijing, China, in 1984 and 1987, respectively, and a Ph.D. degree in mechanical engineering from the University of Michigan, Ann Arbor, in 1992. Currently, he is the Carolyn J. Stewart Chair Professor in the H. Milton Stewart School of Industrial and Systems Engineering and George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta. His research interests include the fusion of advanced statistical and domain knowledge to develop methodologies for modeling, monitoring, diagnosis, and control for complex manufacturing systems. He is a Fellow of the IISE, a Fellow of ASME, a Fellow of INFORMS, an academician of the International Academy for Quality, an elected member of the ISI, and a life member of ASA.

References

Allen, G.I. (2013) Sparse and functional principal components analysis. *arXiv preprint arXiv:1309.2895*.

Chang, S.I. and Yadama, S. (2010) Statistical process control for monitoring non-linear profiles using wavelet filtering and b-spline approximation. *International Journal of Production Research*, **48**(4), 1049–1068.

Chen, K. and Lei, J. (2015) Localized functional principal component analysis. *Journal of the American Statistical Association*, **110**(511), 1266–1275.

Ding, Y., Zeng, L. and Zhou, S. (2006) Phase I analysis for monitoring nonlinear profiles in manufacturing processes. *Journal of Quality Technology*, **38**(3), 199–216.

Grasso, M., Colosimo, B. and Pacella, M. (2014) Profile monitoring via sensor fusion: The use of PCA methods for multichannel data. *International Journal of Production Research*, **52**(20), 6110–6135.

Jensen, W.A., Birch, J. and Woodall, W. (2008) Monitoring correlation within linear profiles using mixed models. *Journal of Quality Technology*, **40**, 167–183.

Jeong, M.K., Lu, J.-C. and Wang, N. (2006) Wavelet-based SPC procedure for complicated functional data. *International Journal of Production Research*, **44**(4), 729–744.

Kim, J., Huang, Q., Shi, J. and Chang, T.-S. (2006) Online multichannel forging tonnage monitoring and fault pattern discrimination using principal curve. *Journal of Manufacturing Science and Engineering*, **128**(4), 944–950.

Lee, H., Battle, A., Raina, R. and Ng, A.Y. (2007) Efficient sparse coding algorithms, in *Advances in Neural Information Processing Systems*, pp. 801–808. Vancouver, Canada.

Mahmoud, M.A. and Woodall, W.H. (2004) Phase I analysis of linear profiles with calibration applications. *Technometrics*, **46**(4), 380–391.

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

Noorossana, R., Eyvazian, M., Amiri, A. and Mahmoud, M.A. (2010) Statistical monitoring of multivariate multiple linear regression profiles in phase I with calibration application. *Quality and Reliability Engineering International*, **26**(3), 291–303.

Noorossana, R., Saghaei, A. and Amiri, A. (2011) *Statistical Analysis of Profile Monitoring*, John Wiley & Sons, Hoboken, NJ.

Paynabar, K. and Jin, J. (2011) Characterization of non-linear profiles variations using mixed-effect models and wavelets. *IIE Transactions*, **43**(4), 275–290.

Paynabar, K., Jin, J. and Pacella, M. (2013) Monitoring and diagnosis of multichannel nonlinear profile variations using uncorrelated multilinear principal component analysis. *IIE Transactions*, **45**(11), 1235–1247.

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

Qiu, P. (2008) Distribution-free multivariate process control based on log-linear modeling. *IIE Transactions*, **40**(7), 664–677.

Qiu, P., Zou, C. and Wang, Z. (2012) Nonparametric profile monitoring by mixed effects modeling. *Technometrics*, **52**(3), 265–277.

Tseng, P. (2001) Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*, **109**(3), 475–494.

Wang, K. and Jiang, W. (2009) High-dimensional process monitoring and fault isolation via variable selection. *Journal of Quality Technology*, **41**(3), 247–258.

Wang, Y., Paynabar, K. and Mei, Y. (2016) Thresholded multivariate principal component analysis for multi-channel profile monitoring. *arXiv preprint arXiv:1603.05265*.

Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S. and Ma, Y. (2009) Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **31**(2), 210–227.

Yan, H., Paynabar, K. and Shi, J. (2015) Image-based process monitoring using low-rank tensor decomposition. *IEEE Transactions on Automation Science and Engineering*, **12**(1), 216–227.

Yu, G., Zou, C. and Wang, Z. (2012) Outlier detection in functional observations with applications to profile monitoring. *Technometrics*, **54**(3), 308–318.

Zhang, C., Yan, H., Lee, S. and Shi, J. (2018) Multiple profiles sensor-based monitoring and anomaly detection. *Journal of Quality Technology*. Manuscript in production.

Zhou, S., Sun, B. and Shi, J. (2006) An SPC monitoring system for cycle-based waveform signals using Haar transform. *IEEE Transactions on Automation Science and Engineering*, **3**(1), 60–72.

Zou, C., Ning, X. and Tsung, F. (2012) Lasso-based multivariate linear profile monitoring. *Annals of Operations Research*, **192**(1), 3–19.

Zou, C., Tsung, F. and Wang, Z. (2008) Monitoring profiles based on nonparametric regression methods. *Technometrics*, **50**(4), 512–526.

Zou, H., Hastie, T. and Tibshirani, R. (2006) Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, **15**(2), 265–286.