IISE Transactions

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

Multiple event identification and characterization by retrospective analysis of structured data streams

Andi Wang, Tzyy-Shuh Chang & Jianjun Shi

To cite this article: Andi Wang, Tzyy-Shuh Chang & Jianjun Shi (2022) Multiple event identification and characterization by retrospective analysis of structured data streams, IISE Transactions, 54:9, 908-921, DOI: 10.1080/24725854.2021.1970863

To link to this article: https://doi.org/10.1080/24725854.2021.1970863

IISE TRANSACTIONS
2022, VOL. 54, NO. 9, 908–921
https://doi.org/10.1080/24725854.2021.1970863

Multiple event identification and characterization by retrospective analysis of structured data streams

Andi Wang[a], Tzyy-Shuh Chang[b], and Jianjun Shi[c]

[a]Arizona State University, Mesa, AZ, USA; [b]OG Technologies, Ann Arbor, MI, USA; [c]Georgia Institute of Technology, Atlanta, GA, USA

ABSTRACT

The sensors installed in complex systems generate massive amounts of data, which contain rich information about a system's operational status. This article proposes a retrospective analysis method for a historical data set, which simultaneously identifies when multiple events occur to the system and characterizes how they affect the multiple sensing signals. The problem formulation is motivated by the dictionary learning method and the solution is obtained by iteratively updating the event signatures and sequences using ADMM algorithms. A simulation study and a case study of the steel rolling process validate our approach. The supplementary materials including the appendices and the reproduction report are available online.

ARTICLE HISTORY
Received 11 February 2021
Accepted 9 August 2021

1. Introduction

In various industries, practitioners increasingly install multiple sensors in complex systems to understand their process conditions. These sensors generate sequences of profile sensing signals data in the form of structured data streams (Jin and Shi, 1999; Zhang *et al.*, 2018). They contain rich information about the system's components and record the system's status during its operation. Before using these sensor measurements for real-time monitoring and control of the process, the practitioners need to collect and review the historical data obtained over a period of time. Through the retrospective analysis of the process data, the engineers can gain insight into the process variation in a longer time scale for discovering new root causes.

For a given system, the sensing signals are subject to the impact of multiple system operation conditions. When the system is operating under normal operation conditions, there is a baseline predictable pattern of the sensing signals. However, various faults may occur in the system at particular intervals during the system operation and lead to the changes of associated sensing signals in specific patterns. We refer to those changes related to the same fault as *one event*. When an event occurs in the system, several associated sensing signals will change according to a specific variation pattern. We refer to the variation pattern associated with each event as *event signatures*. In the following examples of systems, we further illustrate the concepts of events and the corresponding event signatures:

1. In the stamping process considered in Jin and Shi (1999), the tonnage signals are composed of multiple segments, corresponding to phases of operations or mechanical interactions within one stamping cycle. In this system, each event relates to a specific fault in the stamping process, such as material thickness error, loose tie rods, and worn bushings (Koh *et al.*, 1999). Those events lead to respective changes in corresponding segments of tonnage signals, and these changes are event signatures.
2. Phasor measurement units, such as Frequency Disturbance Recorders (FDRs), are deployed in power grids to achieve situation awareness (Wang *et al.*, 2014). There are multiple events that may occur to the grid and affect the associated measurements from the phase measurement units. For example, the event of generator tripping (or load shedding) may cause a decrease (or an increase) of the FDR signal. The event of line tripping may lead to the damping waveform signal. These effects on the sensor measurements are event signatures.
3. In a node of an interconnected cyber system, the usage of CPU, memory, disk, network bandwidth, and power usage are recorded, as they reflect the node's working conditions. Examples of events on this node include regular operations (such as processing, uploading, and downloading) and abnormal situations (such as virus infection or port-scan attacks). According to their event signatures, they cause different behavior on all cyber signals (Li *et al.*, 2019).

The systems discussed above have three common characteristics: First, each system generates one or more events that may occur during the system's operations. Second, during the time of operation, each event appears sporadically, and each occurrence of the event tends to last a period of time after it begins. Third, each event is associated with a small and limited number of sensing signals. In practice, there is a large class of systems holding these characteristics in terms of associated events and their event signatures.

CONTACT Andi Wang andi.wang@gatech.edu
Supplemental data for this article can be accessed online at https://doi.org/10.1080/24725854.2021.1970863.

This study aims at developing automatic retrospective analysis methods for the signals generated from systems with those three characteristics. Our objectives are two-fold: First, we aim at characterizing the event signatures that specify how each event affects the signals. This information enables us to extract useful features from the process and perform online monitoring. Second, we aim to identify the periods in which an event occurs and estimate the event's strengths. Both pieces of information shed light on the root cause diagnosis of those events.

In the literature, some studies have tackled similar data analysis problems. However, as discussed in the following literature review section, existing methods have some limitations in simultaneously addressing the above two questions. For example, some algorithms require pre-defined prototype signals containing individual events in addition to the historical data, which involves extra data preprocessing and human labeling efforts (Wang *et al.,* 2014). Other methods consider the signal partition problem, while they cannot associate segments with events or generate useful event signatures (Zhao, 2013; Guo *et al.,* 2016). Various Phase-I retrospective analyses of the sensing data do not apply to structured data streams. They identify the out-of-control samples without considering either multiple event sequences or their signatures.

This article proposes an algorithm that simultaneously *identifies* the events' periods of occurrence and *characterizes* each event with its signature, including the signals associated with each event. This algorithm is called the Multiple Event Identification and Characterization (MEIC) algorithm. In practice, an event captured by the MEIC algorithm is typically related to faults that have occurred in the process. The identified event signatures will be helpful in finding the root causes of the faults. Therefore, the algorithm serves as an automatic tool that gives practitioners hints to discover new root causes of the processes.

This identification and characterization process can be illustrated from a retrospective data analysis for the shape uniformity of rolling bars, as will be discussed in the case study. We obtain the diameter measurements along six axes of the rolling bars, and each axis of a rolling bar is a functional curve that reflects the dimension along its length. From the data analytics of using the MEIC algorithm, we found two events that occurred during two intervals of production and found the type of variation they caused on the diameter measurements. Those identified events provide valuable information for the process operators on the manufacturing conditions.

The MEIC algorithm works by solving an optimization problem that integrates dictionary learning technique with regularization terms specifying the sparsity of the related signals and the temporal smoothness of event strengths. By solving this optimization problem, we can extract useful representations from sensing signals and identify the occurrence of each event. As will be seen from the simulation study and the real case study, the MEIC algorithm performs well if we start it with multiple initial points to avoid suboptimal solutions.

The remainder of his article is organized as follows. In the next section, we review the related literature in greater detail. Section 3 proposes the system model, introduces the formulation of the optimization problem, and gives the solution algorithms. After that, in Section 4 and Section 5, we present our simulation study and real case study based on a steel rolling process. Conclusions are drawn in Section 6.

2. Literature review

Identifying underlying events from sensing data has been reported in the literature for decades. For example, Jin and Shi (1999) proposed wavelet-based criteria to extract event-related information from tonnage signals and identify the events through monitoring the compressed coefficients. However, this method focuses on wavelet analysis of tonnage signals in each stamping cycle. Although they can identify whether an event happens in a stamping cycle, additional efforts are required to associating multiple stamping cycles to a set of events and identify the periods of those events that affect multiple consecutive products. Yan *et al.* (2018) proposed ST-SSD, an additive decomposition approach that identifies the anomalies from streaming images. However, this approach applies to the specific situation where the streaming images are represented by the summation of a gradual varying background that is smooth in the spatial domain and anomalies that are sparse in both space and time. Also, the ST-SSD method only identifies the anomalies, but it does not associate each anomaly with one type of event. Wang *et al.* (2014) developed a situation awareness system that recognizes the events from the data generated from phasor measurement units. They proposed a two-step analysis method: First, use the k-means clustering approach to form a dictionary of the event signatures, and then formulate an ℓ_1-penalization approach to identify the offset of the event within the signal. The main drawback of this approach is that it requires the existence of a data set of observations corresponding to each single event. Another related method that can be used to identify events from structured data streams is to use phase partition algorithms. These partition algorithms transform the data within each time window into low-dimensional features. For example, Zhao (2013) proposed to calculate the residual of the partial least square regression as features for phase partition, and Guo *et al.* (2016) extracted the covariance matrices of sensor measurements within the window. They then used ad-hoc heuristics or greedy algorithms to find the partition points using these low-dimensional features. Note that these algorithms do not associate each interval with a small set of events and do not consider the possibility of overlapping events during the system operations or the variability of the events' strength.

We also note that the event detection and characterization problem has a close relationship with Phase-I control chart and root cause diagnosis. One can view the events during the operation of the system as assignable causes that lead to extra variability of the system and regard the samples affected by these events as being out of control. In this sense, identifying the occurrence of events can be achieved using a Phase-I control chart. Characterizing the event

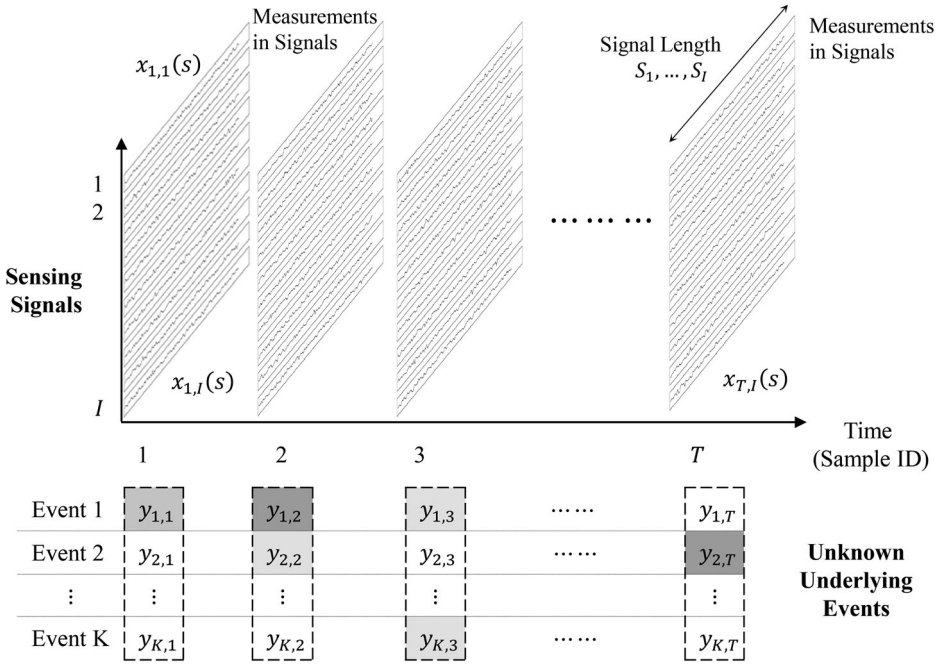


Figure 1. The collected data $x_{t,i}(s)$s and the strengths of the unknown underlying events $y_{k,t}$s.

signatures is then a follow-up diagnostic procedure that associates the out-of-control samples with a small set of root causes. Phase-I monitoring for multiple signal data has been identified as an emerging area in statistical process control (Woodall and Montgomery, 2014). Among the few studies in the literature, Wang *et al.* (2018) monitored the principal component scores under a change-point framework. Although they noted the importance of taking the out-of-control information into account when designing control schemes, they did not assume multiple root causes leading to different out-of-control scenarios. Ebrahimi *et al.* (2020) pointed out that there is a lack of literature on scalable and integrated monitoring, and a suitable diagnosis approach, in current Phase-I charting schemes. Although they proposed a seamless monitoring and diagnosis framework, they performed the event identification (monitoring) step and the event signature characterization (diagnosis) step sequentially, which hinders the utilization of out-of-control information in the control chart design. In the next section, we will see that the approach proposed in this article solves the monitoring and diagnostic problem simultaneously and in an interactive from one formulation. Finally, we note that control charts usually assume a simple probabilistic description of the system. The out-of-control situations are generally simple, which enable probabilistic quantifications of a chart's performance. However, they cannot describe systems with complex event signatures and strength profiles.

The MEIC algorithm we proposed identifies the events associated with each sample and estimates all event signatures. In the literature, prototype methods such as the k-means clustering and Gaussian Mixture Models (Friedman *et al.*, 2001) achieve a similar goal. However, the difference is that prototype methods do not consider the temporal sequence of the samples when assigning them to different events. As an alternative, we will use the dictionary learning technique (Lee *et al.*, 2007) to develop the MEIC algorithm. Unlike the k-means or Gaussian Mixture Model, the dictionary learning method identifies events and describes their effect by directly formulating an optimization criterion. It enables us to incorporate the characteristics of the event signatures and event strength sequences through multiple penalization terms on the associated parameters (Yan *et al.*, 2017, 2018; Mou *et al.*, 2021).

3. The MEIC method

In this section, we first present the assumptions on the system and the collected data. In each time point t from 1 to T, we obtain a sample containing I signals, and the ith signal is of length S_i. The sth measurement obtained from signal i at time t is denoted as $x_{t,i}(s)$, $s = 1, \ldots, S_i; i = 1, \ldots, I; t = 1, \ldots, T$. During this period of time, K events may occur with possible overlaps among them, and let $y_{k,t} \geq 0$ represents the strength of the event k at time t. With the assumption that the events appear, stay, and fade away gradually, $y_{k,t}$ is smooth with the change in time t from 1 to T. The collected data $x_{t,i}(s)$s and the unknown strengths of underlying events $y_{k,t}$s are illustrated in Figure 1. In this figure, the background's gray level of each $y_{k,t}$ indicates the magnitude of this value.

If the observations in signals $x_{t,1}(s), \ldots, x_{t,I}(s)$ are independent and identically distributed when no events occur at time t, the covariance matrix is an identity matrix $\mathbf{\Sigma} = \mathbf{I}$. However, these signals may have both within-signal correlation and between-signal correlation. In this case, we can estimate $\hat{\mathbf{\Sigma}} \in \mathbb{R}^{S \times S}$, the covariance matrix of $\{x_{t,i}(s) : i = 1, \ldots, I; s = 1, \ldots, S_i\}$, where $S = S_1 + \cdots + S_I$ is the total number of measurements from I sensor signals. Note that the estimation process should be based on a small segment of clean (or in control) process data without events, and the estimation procedure can be based on any existing method,

such as robust covariance estimation (Campbell, 1980) or multilinear PCA (Paynabar *et al.*, 2013).

We say that event k occurs at time t when $y_{k,t} > 0$. In Figure 1, the background of $y_{k,t}$ is not white. When event k occurs, each signal i contains a smooth variation pattern $\xi_{k,i}(s)$, $i = 1, ..., I$, and the collection of $\{\xi_{k,i}(s) : i = 1, ..., I\}$ is the signature of event k. Recall that each event is associated with few sensing signals, we have $\xi_{k,i}(s) \equiv 0$ for most i varibles. Under the assumption that all events have additive effects on the signals, we thus represent the signal obtained at time t as $x_{t,i}(s) = \sum_{k=1}^{K} \xi_{k,i}(s) y_{k,t} + \epsilon_{t,i}(s)$.

The measurements from sensor i can be aggregated in a matrix $\mathbf{X}_i \in \mathbb{R}^{S_i \times T}$, with $(\mathbf{X}_i)_{s,t} = x_{t,i}(s)$. All sensor measurements then constitute a data matrix

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_I \end{bmatrix} \in \mathbb{R}^{S \times T}.$$

We have two goals from the inference from $\mathbf{X}$:

1. Identify the periods that each event k occurs, and estimate the strength profiles. We achieve this by estimating $y_{k,t}$ for all events $k = 1, ..., K$ at all time points $t = 1, ..., T$.
2. Characterize each event by its event signature $\xi_{k,i}(s)$ for $k = 1, ..., K$ on all signals $i = 1, ..., I$. We achieve this by estimating $\xi_{k,i}(s), k = 1, ..., K;\ i = 1, ...I,$ and $s = 1, ..., S_i$.

3.1. Problem formulation

To facilitate the estimation procedure, we represent each event signature $\xi_{k,i}(s)$ with the wavelet basis $\{h_j(s) : j = 1, ..., J_i\}$:

$$\xi_{k,i}(s) = \sum_{j=1}^{J_i} b_{k,i,j} h_j(s).$$

In this representation, the effect of event k on signal i is a vector of length J_i, $\mathbf{b}_{k,i} = (b_{k,i,1}, ..., b_{k,i,J_i})^\top$. The event signatures are $\boldsymbol{\xi}_{k,i} = [\xi_{k,i}(1), ..., \xi_{k,i}(S_i)]^\top$ and we have $\boldsymbol{\xi}_{k,i} = \mathbf{H}_i \mathbf{b}_{k,i}$ with

$$\mathbf{H}_i = \left[h_j(s)\right]_{\substack{s = 1, ..., S_i \\ j = 1, ..., J_i}}.$$

The overall effect on signal i at time t is then:

$$\hat{x}_{t,i}(s) = \sum_{k=1}^{K} \xi_{k,i}(s) y_k(t), s = 1, ..., S_i.$$

To represent it in the matrix form, we have $\hat{\mathbf{X}} = \mathbf{HBY}$, where

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & & \\ & \ddots & \\ & & \mathbf{H}_I \end{bmatrix} \in \mathbb{R}^{S \times J}$$

is the collection of all basis,

$$\mathbf{B} = \begin{bmatrix} \mathbf{b}_{1,1} & \cdots & \mathbf{b}_{K,1} \\ \vdots & \ddots & \vdots \\ \mathbf{b}_{1,I} & \cdots & \mathbf{b}_{K,I} \end{bmatrix} \in \mathbb{R}^{J \times K}$$

contains all coefficients that determine the event signatures, and $\mathbf{Y} = (y_{k,t})_{K \times T}$ contains the strengths of all events at all time points, $J = \sum_{i=1}^{I} J_i$. The matrix

$$\hat{\mathbf{X}} = \begin{bmatrix} \hat{\mathbf{X}}_1 \\ \vdots \\ \hat{\mathbf{X}}_I \end{bmatrix} \in \mathbb{R}^{S \times T}$$

with $\hat{\mathbf{X}}_i = (\hat{x}_{t,i}(s))_{S_i \times T}$.

Now, we aim at formulating an optimization problem to solve the values of $\mathbf{B}$ and $\mathbf{Y}$, which respectively characterize the event signatures and the strength of the events during the T time points. First, the matrix $\hat{\mathbf{X}}$ provides an approximation to the data matrix $\mathbf{X}$, and we define the loss as the squared Mahalanobis distance $\mathrm{vec}(\mathbf{X} - \hat{\mathbf{X}})^\top \hat{\boldsymbol{\Sigma}}^{-1} \mathrm{vec}(\mathbf{X} - \hat{\mathbf{X}})$. Note that when $\hat{\boldsymbol{\Sigma}}$ is not invertible, we may replace $\hat{\boldsymbol{\Sigma}}^{-1}$ with $\left[\hat{\boldsymbol{\Sigma}} + \epsilon \mathbf{I}\right]^{-1}$ where $\epsilon > 0$ is a small positive number, and $\mathbf{I}$ is the identity matrix. From this equation, we can see that the loss is a quadratic function with regard to $\hat{\mathbf{X}} = \mathbf{HBY}$. For ease of presentation, we assume $\hat{\boldsymbol{\Sigma}} = \mathbf{I}$ in the remainder of this article, and thus the loss reduces to

$$\mathrm{vec}(\mathbf{X} - \hat{\mathbf{X}})^\top \mathrm{vec}(\mathbf{X} - \hat{\mathbf{X}}) = \|\mathbf{X} - \hat{\mathbf{X}}\|_F^2.$$

From the solution algorithm shown later, we will see that the estimated covariance $\hat{\boldsymbol{\Sigma}} \neq \mathbf{I}$ can be handled in a similar maner.

In addition to the loss, we add the following penalization terms to represent the event signatures and the sequence of event strengths.

3.1.1. The event signatures

The wavelet basis usually gives an overcomplete representation of the signals. Motivated by the wavelet shrinkage method (Donoho and Johnstone, 1995), we first apply an ℓ_1 regularization $\lambda_1 \|\mathbf{B}\|_{1,1}$ to improve the estimation of the event signature $\xi_{k,i}(s)$s through overcoming the curse of dimensionality. Recall that the $\boldsymbol{\xi}_{k,i} = \mathbf{0}$ for most event k and signal i, because each event is associated with few signals. Therefore, we have $\mathbf{b}_{k,i} = \mathbf{0}$ for most event k and signal i. We thus add another group Lasso penalty $\lambda_2 \sum_{i=1}^{I} \sum_{k=1}^{K} \|\mathbf{b}_{k,i}\|_2$.

3.1.2. The sequence of event strengths

We assume that each event stays for a period of time in a high-volume manufacturing process. Therefore, multiple products in a sequence can be affected by an event. Furthermore, inertia typically exists in physical systems. Even if the actual root causes of the event appear or disappear sharply, their effects on the sensor observations are usually smooth in time. The persistent nature of the event and the inertia of a physical system motivate us to assume a continuous property of the sequence for every event's

strength. Specifically, we apply the smoothness penalty $\lambda_3\|\mathcal{D}\mathbf{Y}^\top\|_F^2$, where

$$\mathcal{D} = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix}$$

is the second-order smoother that applies to the temporal mode of $\mathbf{Y}$. Furthermore, every event only occurs sporadically in time, and therefore we add the ℓ_1-penalty $\lambda_4\|\mathbf{Y}\|_{1,1}$.

Integrating the loss function and all penalties mentioned above, we derive the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{B},\mathbf{Y}} \|\mathbf{X} - \mathbf{H}\mathbf{B}\mathbf{Y}\|_F^2 + \lambda_1\|\mathbf{B}\|_{1,1} \\ &+ \lambda_2 \sum_{i=1}^{I}\sum_{k=1}^{K} \|\mathbf{b}_{k,i}\|_2 + \lambda_3\|\mathcal{D}\mathbf{Y}^\top\|_F^2 + \lambda_4\|\mathbf{Y}\|_{1,1} \\ &\text{subject to } \|\mathbf{b}_{k,\cdot}\|_2 = 1,\ y_{k,t} \geq 0,\ k = 1,\ldots,K; t = 1,\ldots,T, \end{aligned} \tag{1}$$

where $\mathbf{Y} = [\mathbf{y}_{\cdot,1},\ldots,\mathbf{y}_{\cdot,T}]$. Note that we added another constraint $\|\mathbf{b}_{k,\cdot}\|_2 = 1$ in this formulation, where

$$\mathbf{b}_{k,\cdot} = \begin{bmatrix} \mathbf{b}_{k,1} \\ \vdots \\ \mathbf{b}_{k,I} \end{bmatrix}$$

is the coefficient vector of event k corresponding to all signals $1,\ldots,I$. Due to the orthogonality of $\mathbf{H}$, this condition indicates that $\|\boldsymbol{\xi}_{k,\cdot}\|_F^2 = \sum_{i=1}^{I}\|\boldsymbol{\xi}_{k,i}\|_F^2 = 1$. Essentially, it keeps the scales of all event signatures the same and specifies the unit for measuring the strengths of each event.

Problem (1) is motivated by the dictionary learning problem (Lee *et al.*, 2007). In this problem, matrices $\mathbf{B}$ and $\mathbf{Y}$ simultaneously give a K-dimensional representation of the historical data $\mathbf{X}$, where the matrix $\mathbf{B}$ represents the wavelet coefficients that define the events, and $\mathbf{Y}$ represents the sequences of the events' strengths. In the standard dictionary learning formulation (Lee *et al.*, 2007), it is assumed that events occur sporadically ($\mathbf{Y}$ in Problem (1) is sparse) and that the data is affected by few types of events (K is small). These assumptions are critical for the feasibility of the algorithm. In addition to these assumptions, the smoothness assumption on the continuity of the event strengths and the regularizations on the matrix $\mathbf{B}$ improve the interpretability and predictive accuracy of the estimation. These assumptions can be tailored based on the necessity of particular systems.

We may compare the optimization problem (1) with the formulation of the smooth sparse decomposition methods (Yan *et al.*, 2017, 2018; Mou *et al.*, 2021). These methods focus on decomposing the image or tensor data into the sum of the smooth background and the anomaly with sparsity properties. However, in our problem formulation, the data matrix $\mathbf{X}_i$ of sensor i has a zero trend without event, and therefore, is a sole anomaly component. This data matrix is decomposed into $\sum_{k=1}^{K}\xi_{k,i}(s)y_k(t)$, denoting the additive effect of several types of events happening at sparse time intervals. In this sense, the MEIC method provides a finer anatomy of the anomalies comparing with the smooth sparse decomposition techniques.

Based on the solution of $\mathbf{B}$ and $\mathbf{Y}$, we can answer the two questions discussed at the beginning of this section. For every event $k = 1,\ldots,K$, we can obtain $\mathbf{b}_{k,\cdot}$, which shows the effect of this event on the signal i as $\xi_{k,i}(s) = \sum_{j=1}^{J_i} b_{k,i,j}h_j(s)$. Also, from $\mathbf{y}_{k,\cdot} \in \mathbb{R}^T$, the kth row of $\mathbf{Y}$, we can identify the time that the event k occurs with considerable strength and the time at which event k does not occur. From this perspective, this framework performs events characterization (diagnostics) and event identification (off-line detection) simultaneously. The readers should note that the penalizations may perform two functions here. First, the minimization of the loss involves many coefficients in the matrices $\mathbf{B}$ and $\mathbf{Y}$. Regularizations make it possible to obtain a solution that satisfies the sparsity and smoothness conditions and improves the estimation accuracy. The term $\lambda_4\|\mathbf{Y}\|_{1,1}$ enables us to identify the time points that each event occurs by observing if $y_{k,t} = 0$. In this sense, choosing λ_2 and λ_4 enables the practitioners to adjust the number of sensors associated with each event and the number of time points at which the event occurs to facilitate the root cause diagnosis of the system.

3.2. Solution algorithms

Problem (1) can be solved through a Blockwise Coordinate Descent (BCD) algorithm, where we iteratively update the matrix $\mathbf{B}$ and the matrix $\mathbf{Y}$, as shown in Algorithm 1. The steps of updating $\mathbf{B}$ and updating $\mathbf{Y}$ are performed with two alternative-direction method of multipliers (ADMM) algorithms, detailed in the following two subsections. In Section 3.2.3, we discuss how to select the initial value of $\mathbf{Y}^0$.

Algorithm 1. The MEIC algorithm

Initiate $\mathbf{Y} = \mathbf{Y}^0$.
Loop until converge:
  Update $\mathbf{B}$ given $\mathbf{Y}$, as detailed in Section 3.2.1.
  Update $\mathbf{Y}$ given $\mathbf{B}$, as detailed in Section 3.2.2.

3.2.1. Updating B

To update $\mathbf{B}$, we need to solve Problem (2):

$$\begin{aligned} &\min_{\mathbf{B}} \|\mathbf{X} - \mathbf{H}\mathbf{B}\mathbf{Y}\|_F^2 + \lambda_1\|\mathbf{B}\|_{1,1} + \lambda_2\sum_{i=1}^{I}\sum_{k=1}^{K}\|\mathbf{b}_{k,i}\|_2 \\ &\text{subject to } \|\mathbf{b}_{k,\cdot}\|_2 = 1. \end{aligned} \tag{2}$$

This problem can be reformulated as

$$\min_{\mathbf{Z}} \sum_{m=1}^{M} f_m(\mathbf{Z}) \tag{3}$$

where $M = 4$, $f_1(\mathbf{B}) = \|\mathbf{X} - \mathbf{H}\mathbf{B}\mathbf{Y}\|_F^2$, $f_2(\mathbf{B}) = \lambda_1\|\mathbf{B}\|_{1,1}$, $f_3(\mathbf{B}) = \lambda_2\sum_{i=1}^{I}\sum_{k=1}^{K}\|\mathbf{b}_{k,i}\|_2$, and $f_4(\mathbf{B}) = \sum_{k=1}^{K} I_{\|\mathbf{b}_{k,\cdot}\|_2=1}$.

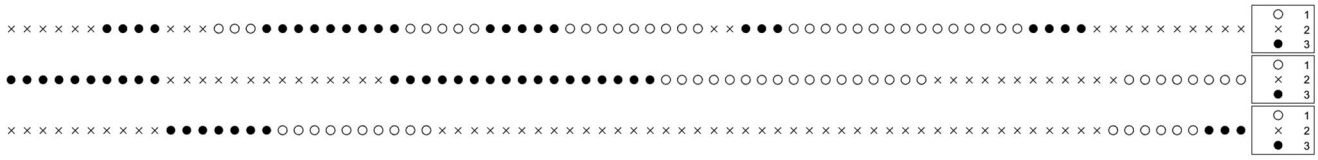


Figure 2. The sample paths of $\mathbf{y}^0$ in initialization, corresponding to $\alpha = 1/5, 1/10,$ and $1/15$. The circles, crosses, and dots indicate three events.

Problem (3) can be solved with the ADMM consensus algorithm (Parikh and Boyd, 2014), summarized in Algorithm 2 below.

Algorithm 2: The ADMM consensus Algorithm

Initiate replicates $\mathbf{Z}^{(m)}, \mathbf{U}^{(m)} = \mathbf{O}$ for $m = 1, ..., M$, of the same shape as $\mathbf{Z}$. Set step size η.
Iterate until convergence:
  Update $\mathbf{Z}^{(m)} = \text{prox}_{\eta f_m}[\overline{\mathbf{Z}} - \mathbf{U}^{(m)}]$, for $m = 1, ..., M$ in parallel.
  $\overline{\mathbf{Z}} = \sum_{m=1}^{M} \mathbf{Z}_m / M$
  $\mathbf{U}^{(m)} = \mathbf{U}^{(m)} + \mathbf{Z}^{(m)} - \overline{\mathbf{Z}}$.

To implement Algorithm 2 in solving Problem (2), we need to evaluate the proximal operators of $\eta f_1, ..., \eta f_4$. The results are in Proposition 1, and the derivation is in Appendix A.

Proposition 1. *Let* $f_1(\mathbf{B}) = \|\mathbf{X} - \mathbf{HBY}\|_F^2, f_2(\mathbf{B}) = \lambda_1 \|\mathbf{B}\|_{1,1}$, $f_3(\mathbf{B}) = \lambda_2 \sum_{i=1}^{I} \sum_{k=1}^{K} \|\mathbf{b}_{k,i}\|_2$ *and* $f_4(\mathbf{B}) = \sum_{k=1}^{K} I_{\|\mathbf{b}_{k,\cdot}\|_2 = 1}$. *The proximal operators of* $\eta f_1, ..., \eta f_4$ *are given as follows*: *Let* $\mathbf{A}$ *and* $\mathbf{Z}$ *have the same size as* $\mathbf{B}$, *and partition them into*

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_{1,1} & \cdots & \mathbf{a}_{K,1} \\ \vdots & \ddots & \vdots \\ \mathbf{a}_{1,I} & \cdots & \mathbf{a}_{K,I} \end{bmatrix} \text{ and } \mathbf{Z} = \begin{bmatrix} \mathbf{z}_{1,1} & \cdots & \mathbf{z}_{K,1} \\ \vdots & \ddots & \vdots \\ \mathbf{z}_{1,I} & \cdots & \mathbf{z}_{K,I} \end{bmatrix}$$

according to

$$\mathbf{B} = \begin{bmatrix} \mathbf{b}_{1,1} & \cdots & \mathbf{b}_{K,1} \\ \vdots & \ddots & \vdots \\ \mathbf{b}_{1,I} & \cdots & \mathbf{b}_{K,I} \end{bmatrix}.$$

Similarly, we let $\mathbf{B}_{\cdot,i} = [\mathbf{b}_{1,i} ... \mathbf{b}_{K,i}]$, $\mathbf{Z}_{\cdot,i} = [\mathbf{z}_{1,i} ... \mathbf{z}_{K,i}]$, *and* $\mathbf{A}_{\cdot,i} = [\mathbf{a}_{1,i} ... \mathbf{a}_{K,i}]$.

1. *If* $\mathbf{Z} = \text{prox}_{\eta f_1}[\mathbf{A}]$, *we have*

$$\text{vec}(\mathbf{Z}_{\cdot,i}) = \left(\mathbf{E}_{KJ_i \times KJ_i} + \eta(\mathbf{Y}\mathbf{Y}^\top) \otimes \mathbf{E}_{J_i \times J_i}\right)^{-1} \left[\text{vec}(\mathbf{A}_{\cdot,i}) + \eta \text{vec}\left(\mathbf{H}_i^\top \mathbf{X}_i \mathbf{Y}^\top\right)\right],$$

   where $\otimes$ *is the Kronecker product, and* $\mathbf{E}_{r \times r}$ *is the identity matrix of order* r.
2. *If* $\mathbf{Z} = \text{prox}_{\eta f_2}[\mathbf{A}]$, $Z_{l,i} = S_{\lambda_1 \eta}(A_{l,i})$, *where* $Z_{l,i}, A_{l,i}$ *are the* (l,i) *element of* $\mathbf{Z}$ *and* $\mathbf{A}$ *respectively, and*

$$S_{\lambda\eta}(x) = \begin{cases} x + \lambda_1 \eta, & x \le -\lambda_1 \eta \\ 0, & -\lambda_1 \eta \le x \le \lambda_1 \eta \\ x - \lambda_1 \eta, & x > \lambda_1 \eta \end{cases}.$$

3. *If* $\mathbf{Z} = \text{prox}_{\eta f_3}[\mathbf{A}]$, $\mathbf{z}_{k,i} = \left(1 - \frac{\eta \lambda_2}{\|\mathbf{a}_{k,i}\|}\right) \mathbf{a}_{k,i}$.
4. *If* $\mathbf{Z} = \text{prox}_{\eta f_4}[\mathbf{A}]$, $\mathbf{z}_{k,\cdot} = \frac{\mathbf{a}_{k,\cdot}}{\|\mathbf{a}_{k,\cdot}\|}$. *Here* $\mathbf{a}_{k,\cdot}$ *and* $\mathbf{z}_{k,\cdot}$ *are the* k*th column of* $\mathbf{A}$ *and* $\mathbf{Z}$, *respectively.*

3.2.2. Updating Y

In $\mathbf{Y}$-update, we need to solve Problem (4):

$$\min_{\mathbf{B},\mathbf{Y}} \|\mathbf{X} - \mathbf{HBY}\|_F^2 + \lambda_3 \sum_{t=1}^{T} \|\mathcal{D}\mathbf{Y}^\top\|_F^2 + \lambda_4 \|\mathbf{Y}\|_{1,1} \quad \text{subject to } y_{k,t} \ge 0, k = 1, ..., K; t = 1, ..., T. \tag{4}$$

Problem (4) is also in the form of Problem (3), with $M = 3$ and $f_1(\mathbf{Y}) = \|\mathbf{X} - \mathbf{HBY}\|_F^2$, $f_2(\mathbf{Y}) = \lambda_3 \sum_{t=1}^{T} \|\mathcal{D}\mathbf{Y}^\top\|_F^2$ and $f_3(\mathbf{Y}) = \lambda_4 \|\mathbf{Y}\|_{1,1} + \sum_{t=1}^{T} \sum_{k=1}^{K} I_{y_{k,t} \ge 0}$. We use Algorithm 2 again to solve the problem, with the proximal operators of the functions that are given in Proposition 2. The derivation is given in Appendix B.

Proposition 2. *Let* $f_1(\mathbf{Y}) = \|\mathbf{X} - \mathbf{HBY}\|_F^2$, $f_2(\mathbf{Y}) = \lambda_3 \sum_{t=1}^{T} \|\mathcal{D}\mathbf{Y}^\top\|_F^2$ *and* $f_3(\mathbf{Y}) = \lambda_4 \|\mathbf{Y}\|_{1,1} + \sum_{t=1}^{T} \sum_{k=1}^{K} I_{y_{k,t} \ge 0}$. *The proximal operators of* $\eta f_1, ..., \eta f_3$ *are given as follows*:

1. *If* $\mathbf{Z} = \text{prox}_{\eta f_1}[\mathbf{A}]$, $\mathbf{z}_{\cdot,t} = \left(\mathbf{E}_{K \times K} + \eta \mathbf{B}^\top \mathbf{B}\right)^{-1} \left[\mathbf{a}_{\cdot,t} + \eta \mathbf{B}^\top \mathbf{H}^\top \mathbf{x}_{\cdot,t}\right]$, *for* $t = 1, ..., T$.
2. *If* $\mathbf{Z} = \text{prox}_{\eta f_2}[\mathbf{A}]$, $\mathbf{z}_{k,\cdot} = \mathcal{F}^{-1}\left[\mathbf{c} \odot \mathcal{F}[\mathbf{a}_{k,\cdot}]\right]$ *where* $\mathcal{F}$ *and* $\mathcal{F}^{-1}$ *denotes the Discrete Fourier Transform and Inverse Discrete Fourier Transform, respectively.* "$\odot$" *represents the elementwise product and* $\mathbf{c} \in \mathbb{R}^T$ *with*

$$c_t = \frac{1}{1 + 4\lambda_3 \eta \left(1 - \cos \frac{(t-1)}{T} \pi\right)^2}, \quad t = 1, ..., T.$$

3. *If* $\mathbf{Z} = \text{prox}_{\eta f_3}[\mathbf{A}]$, $Z_{k,t} = \max(A_{k,t} - \lambda_4 \eta, 0)$ *for all* $k = 1, ..., K$ *and* $t = 1, ..., T$.

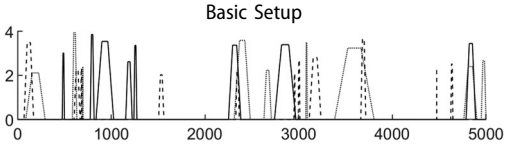
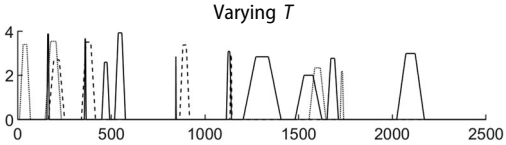
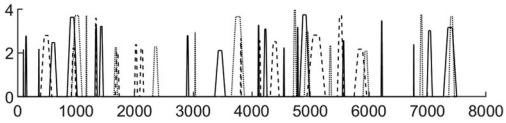
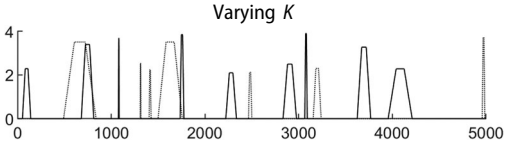
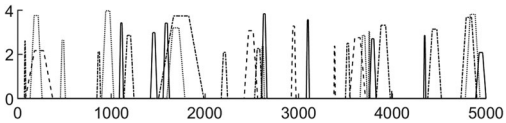
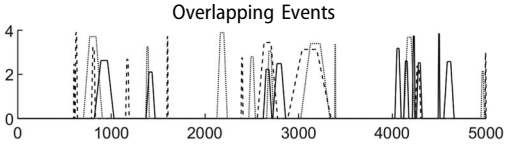
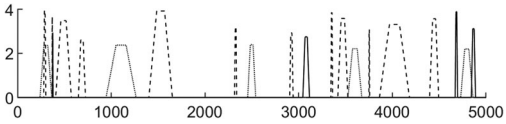
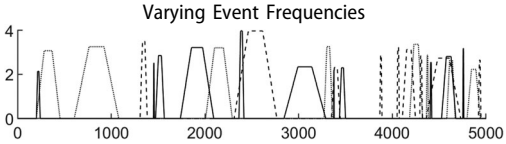
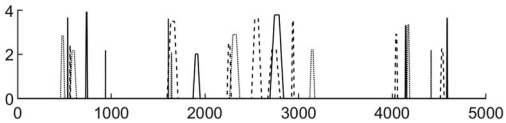
Here $\mathbf{z}_{k,\cdot}, \mathbf{z}_{\cdot,t}$, and $Z_{k,t}$ denotes the row k, column t, and element (k,t) of matrix $\mathbf{Z}$. The notations $\mathbf{a}_{k,\cdot}, \mathbf{a}_{\cdot,t}$, and $A_{k,t}$ are similarly defined. $\mathbf{x}_{\cdot,t}$ represents the tth column of $\mathbf{X}$.

3.2.3. Initialization

Problem (1) is not convex. Moreover, the problem of $\mathbf{B}$-update is not convex, due to the constraint $\|\mathbf{b}_{k,\cdot}\|_F = 1$, and thus we can only obtain a local optimum. The $\mathbf{Y}$-update is a convex problem. In general, the MEIC algorithm converges to a local optimum, and therefore a good initialization of $\mathbf{Y}$ is important.

In the MEIC algorithm, there are two intuitive considerations about the initial value $\mathbf{Y}^0$. First, we hope that in the first step of $\mathbf{B}$-update, the columns of the solution $\mathbf{B}$ are significantly different from each other, to capture the

Table 1. Simulation setups.

Index	*Event Sequence*	*T*	*K*	*Overlap*	*Frequency*
	Basic Setup				
1		5000	3	M	M
	Varying *T*				
2		2500	3	M	M
3		7500	3	M	M
	Varying *K*				
4		5000	2	M	M
5		5000	4	M	M
	Varying the Number of Overlapping Events				
6		5000	3	H	M
7		5000	3	L	M
	Varying Event Frequencies				
8		5000	3	M	H
9		5000	3	M	L

information regarding multiple events. Therefore, we want the collection of time points with large $Y^0_{t,k}$ to be different for different k values. To achieve this, we assign only one event at each time t in the initial event sequences $\mathbf{Y}^0$, so that for different k, the collections of time points with large $Y^0_{t,k}$ are disjoint. Furthermore, it is good to assign small consecutive time points in $\mathbf{Y}^0$ to the same event k, because the true values of $\mathbf{Y}_{\cdot,t}$ and $\mathbf{Y}_{\cdot,t'}$ are similar when times t and t' are close given the continuity of the events in the temporal domain. Second, an event may occur at any time. Therefore, the sequence corresponding to every event should cover the entire sequence.

Based on the above considerations, we propose the following scheme of initializing $\mathbf{Y}^0$. We generate $\mathbf{Y}^0$ randomly based on a Markovian chain with K states. Specifically, let $\mathbf{y}^0$ be a Markov chain on state $\{1, ..., K\}$ with the following transition probability:

$$p_{k,k} = 1 - \alpha, \text{ and } p_{k,k'} = \tfrac{\alpha}{K-1}.$$

Here, α is a tuning parameter that adjusts the frequency of jumps between states. After simulating $\mathbf{y}^0$ as a path of T time points, we set $Y^0_{k,t} = 1$ if $y^0_t = k$, and $Y^0_{k,t} = 0$,

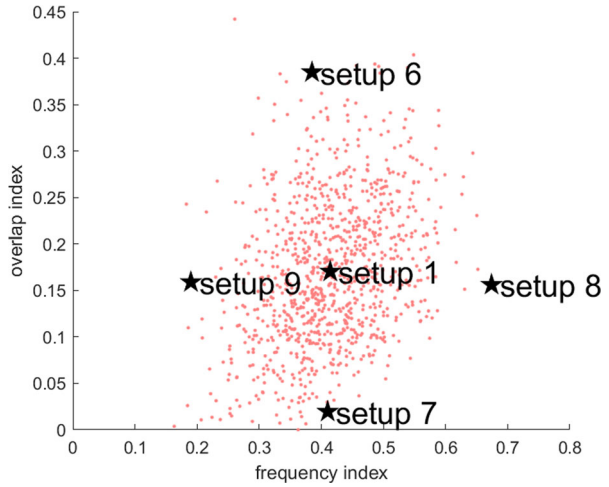


Figure 3. The scatter plot of r_f and r_o and selected event sequences $\mathbf{Y}$'s.

otherwise. Three sample paths of $\mathbf{y}^0$ corresponding to three values of $\alpha = 1/5, \alpha = 1/10$, and $\alpha = 1/15$ are illustrated in Figure 2 with $K = 3$ and $T = 80$. In practice, α can be selected as the inverse of the expected length of the event periods.

Note that the initialization procedure of $\mathbf{Y}^0$ is random. It enables us to run Algorithm 1 multiple times with different realizations of $\mathbf{Y}^0$, and select the local optimum with the minimal objective value.

3.2.4. Selecting the tuning parameters and the identifiability issue

In the MEIC method, there are four tuning parameters in total. As these parameters correspond to different characteristics of $\mathbf{B}$ and $\mathbf{Y}$, we suggest the following one-step BCD procedure of selecting their values. The procedure is semi-automatic, so that the practitioners can incorporate their knowledge of the event sequences and the approximate number of signals involved in every type of event, which include the following steps:

1. Identify the sensitivity of every tuning parameter λ_i by adjusting it and observing the change of the estimated event signature and event sequence.
2. Start with adjusting the least sensitive parameter, then adjust the next sensitive one sequentially, until the most sensitive parameter to achieve the desired property.

In Step 2 above, the tuning parameters λ_1 and λ_2 on $\mathbf{B}$ should be selected smaller to achieve the desired estimation accuracy, as $\mathbf{B}$ characterizes the events' signatures. However, the tuning parameters λ_4 that control the sparsity of $\mathbf{Y}$ can be selected larger, as $\mathbf{Y}$ is typically used to identify the period of events, which is similar to variable selection. Also, note that the solution of $\mathbf{Y}$ guarantees sparsity, whereas $\mathbf{B}$ does not: if we fix $\mathbf{B}$, the estimation of $\mathbf{Y}$ is sparse due to the Lasso penalization. However, solving $\mathbf{B}$ fixing $\mathbf{Y}$ is a non-convex optimization problem, and the algorithm cannot guarantee a sparse solution. Because $\mathbf{Y}$ identifies the periods of the events, only the sparsity of $\mathbf{Y}$ is desired, and thus it is not a limitation of the method.

To use the MEIC method, we found that if we choose the value of K bigger than the number of actual events, more than one estimated event with similar signatures may be generated, corresponding to the same actual event. These estimated events correspond to different time intervals, which constitutes durations that this actual event occurs. In this case, it is necessary to aggregate the identified events to give the final results if K is too large.

Generally, our experience advocates selecting a smaller number of events. When no abnormal event occurs to the system, we should have $K = 0$, indicating that no events affect the data. When employing the MEIC method based on an existing dataset, we may set a small number of K and thereby focus on the few types of events that have major impacts on the sensing signals. After these major types of events are identified, we should take actions to eliminate these quality issues, then collect new production data and use the MEIC method again to identify the secondary types of events that affect the sensing signals.

The success of the dictionary learning method does not require that the signatures be orthogonal like PCA. However, if two signatures are identical with each other, the learning algorithm cannot identify the corresponding sequences of event strengths. In this situation, the two events with identical signatures will be regarded as one event. Similarly, if two sequences of event strengths are very similar with each other, these events always happen together, and thus, we cannot distinguish their corresponding event signatures. However, the detailed statistical assumptions that guarantee the identifiability will be left for future study.

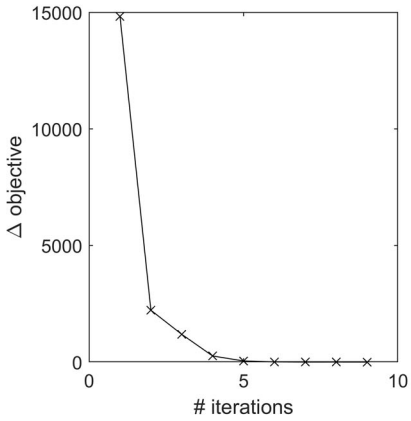
4. Simulation studies

In this section, we present the following simulation study to investigate the MEIC algorithm. As discussed in the literature review, there is no method that simultaneously characterizes the event signatures and identifies the event sequences based on historical data. Therefore, the simulation study proposed here mainly aims to evaluate the performance of our method and test if the MEIC algorithm is stable under multiple settings. To illustrate the benefit of the MEIC method over the prototype methods, we also compare it with the k-means method for event identification.

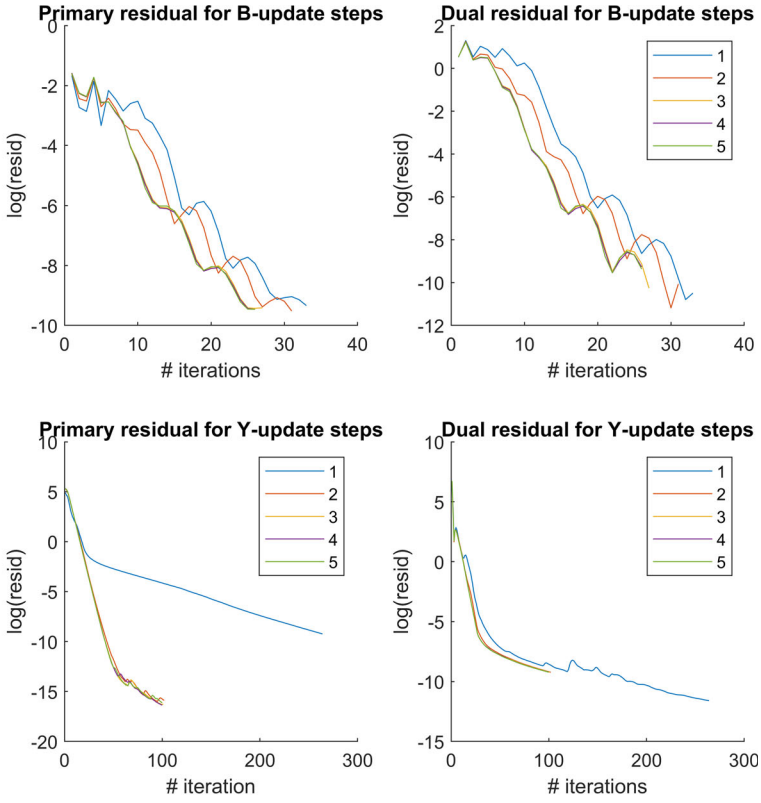
4.1. Simulation setup

We consider a simulation testbed of $I = 10$ signals, each with length $J_i = 128$, $i = 1, \ldots, 10$. In general, we generate the data $\mathbf{X}$ by simulating the coefficients of the event signatures $\mathbf{B}$ and the event sequences $\mathbf{Y}$, and then calculating $\mathbf{X} = \mathbf{HBY} + \mathbf{E}$, where $E_{ij} \sim N(0, 0.1^2)$. We select the Haar basis as the wavelet basis $\mathbf{H}$.

In practice, there is a wide variety of possible scenarios in terms of the number of events, the length of the historical data, as well as the periods in which events occur. In our simulation study, we therefore consider evaluating the



(a) the convergence of the BCD algorithm



(b) convergence of each B-update and Y-update

Figure 4. (a) The change of the objective value converges to zero within several iterations in the BCD algorithm and (b) the primary and dual residual converges to zero rapidly for both B-update and Y-update steps.

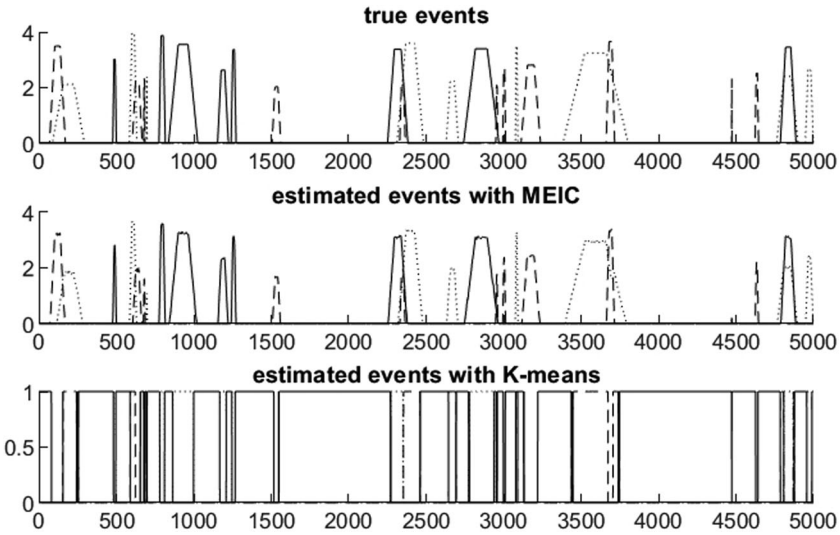


Figure 5. The true event sequence and the estimated event sequence according to Setup 1.

algorithm under different scenarios by varying the lengths of the historical data T, the total number of events K, the lengths of time duration covered with at least one event, and the lengths of time duration with overlapping events. Specifically, we select nine represented scenarios upon which the MEIC algorithms are tested. Their number of events, length of the historical data, associated characteristics, and the event sequences $\mathbf{Y}$ are shown in Table 1, where the letters H, M, and L represent High, Medium, and Low. In the subsection below, we detail the algorithm of generating $\mathbf{Y}$.

4.1.1. Generating the event sequences Y

The matrices $\mathbf{Y}$ corresponding to the nine setups are generated from the same general randomized procedure. Our strategy is to first use this procedure to generate $R = 1000$ realizations of $\mathbf{Y}$, and then we select typical realizations $\mathbf{Y}$

Table 2. The error rate of event sequence identification.

Setup		*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
MEIC	Type I	0.0076	0.0071	0.0076	0.0089	0.0069	0.0092	0.0062	0.0136	0.0033
	Type II	0.0023	0.0039	0.0023	0.0018	0.0011	0.0019	0.0024	0.0020	0.0040
	Total	0.0099	0.0109	0.0098	0.0107	0.0080	0.0111	0.0086	0.0156	0.0073
k-means	Type I	0.0654	0.0755	0.0628	0.0633	0.0926	0.0895	0.0616	0.0983	0.0279
	Type II	0.0000	0.0301	0.0000	0.0000	0.0111	0.0000	0.0235	0.0000	0.0000
	Total	0.0654	0.1056	0.0628	0.0633	0.1037	0.0895	0.0851	0.0983	0.0279

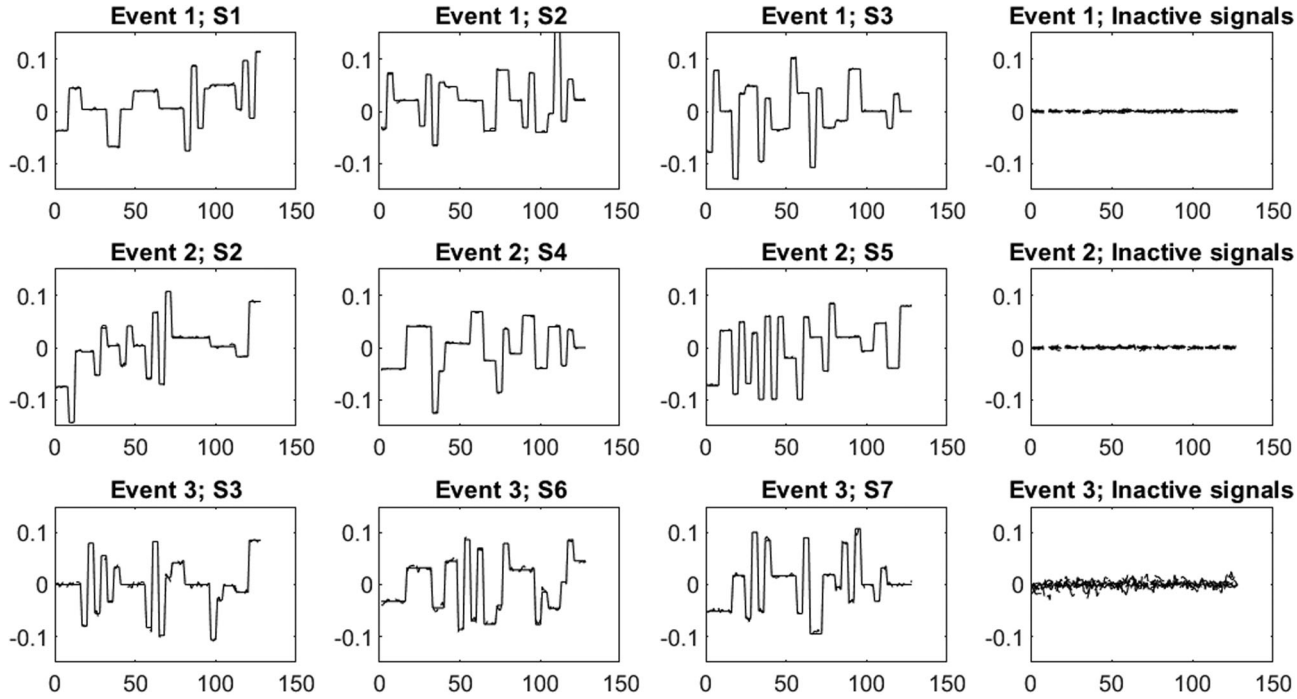


Figure 6. The estimated event signatures on 10 sensors of the three events, according to Setup 1. The horizontal axis in each figure represents the measurement points in each signal.

Table 3. The values of V_ξ for nine setups.

	Setup								
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
V_ξ	0.0010	0.0036	0.0040	0.0010	0.0089	0.0013	0.0012	0.0030	0.0010

for each of the nine setups. In what follows, we first describe the general randomized procedure and then demonstrate how the matrices $\mathbf{Y}$ corresponding to the nine setups are selected.

The general randomized procedure of generating a matrix $\mathbf{Y}$ is as follows. We independently generate the strength sequence $\mathbf{Y}_{k,\cdot}$ for every event $k = 1, ..., K$. For each sequence $\mathbf{Y}_{k,\cdot}$, we generate it from an alternating renewal process (Ross, 1996), of which the occurrences of event k and the interims appear iteratively. Their lengths are exponentially distribution with Exp(100^{-1}) and Exp(500^{-1}), respectively. In each occurrence of an event, the event strength increases from zero to a random level following $U(2, 4)$ and then decreases gradually to zero. In each interim, $Y_{k,t} = 0$.

For each randomly generated matrix of $\mathbf{Y}$, we define two indices. The *frequency* index r_f reflects the frequency of time points at which more than zero events occur, and the *overlap* index r_o reflects the proportion of time points with more than one overlapping event to all time points with at least one event occurs. Here $\#\{A\}$ means the number of elements in set A :

$$r_f = \#\left\{t : \sum_{k=1}^{K} 1_{\{Y_{k,t}>0\}} > 0, t = 1, ..., T\right\}/T$$

$$r_o = \#\left\{t : \sum_{k=1}^{K} 1_{\{Y_{k,t}>0\}} > 1, t = 1, ..., T\right\}$$
$$/\#\left\{t : \sum_{k=1}^{K} 1_{\{Y_{k,t}>0\}} > 0, t = 1, ..., T\right\}$$

In Setups 1, 6, 7, 8, and 9, the numbers of events are all $K = 3$, and the length of the historical data are all T =5000. We generate $R = 1000$ candidate matrices of $\mathbf{Y}$ with the prescribed values of K and T through the procedure described above. Then, we draw the scatter plot of (r_f, r_o) for these replicates, as shown in Figure 3. We select five points from the scatter plot, corresponding to medium r_o and medium r_f, with high/low r_o and medium r_f, and with medium r_o and high/low r_f, as illustrated by the star points in Figure 3. These points correspond to the matrices $\mathbf{Y}$s for Setups 1, 6, 7, 8 and 9.

In Setups 2, 3, 4, and 5, we obtain the $\mathbf{Y}$ in a similar procedure. For each setup, we generate $R = 1000$ candidate

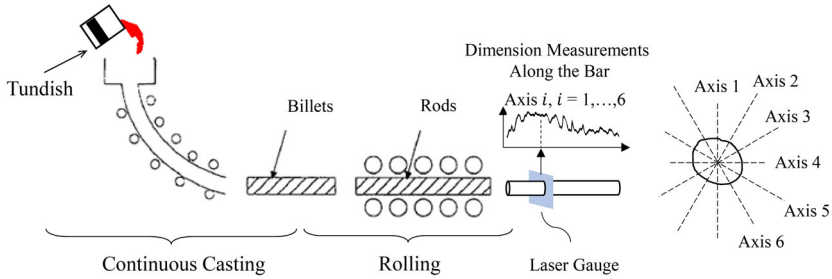


Figure 7. An illustration of a rolling process, where the blue square represents the measurement plane of the laser gauge. The shape at the right-hand side illustrates the cross-sectional shape of a rolling bar and its diameter measurements along six axes.

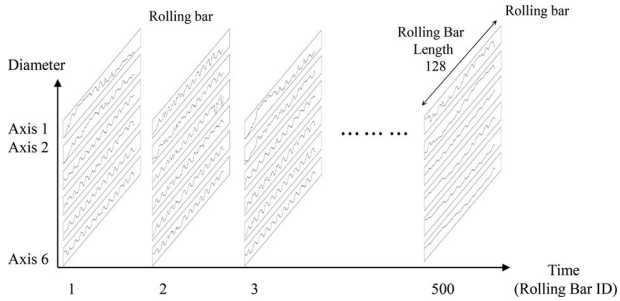


Figure 8. The illustration of the raw data for the case study.

matrices of $\mathbf{Y}$ corresponding to the corresponding values of K and T. Among these candidate matrices, we pick the $\mathbf{Y}$ whose r_f and r_o indices are close to their respective average among all 1000 replicates.

4.1.2. Generate the event signature coefficient B

In the simulation platform, we assume each event affects three sensing signals. Therefore, there are three signal is such that $\mathbf{b}_{k,i} \neq \mathbf{0}$ for every $k = 1, ..., K$. In the nine setups above, there are at most $K = 4$ events involved in each simulation setup. We let event $k = 1$ affect signal i =1, 2, 3, event 2 affect signals 2, 4, 5, event 3 affect signals 3,6,7, and event 4 affect signals 5, 7, 8. For each event k and one affected signal i, we generate $\mathbf{b}_{k,i}$ by randomly choosing five non-zero elements and sample their values from $U(1,3)$. Finally, each vector $\mathbf{b}_{k,\cdot}$ is scaled to the unit length.

4.2. Estimation results and comparisons

For each simulation setup, we obtain the estimated event signature $\left\{\hat{\boldsymbol{\xi}}_{k,i} : k = 1, ..., K; i = 1, ..., I\right\}$ and event sequences $\left\{\hat{y}_{k,t} : k = 1, ..., K; t = 1, ..., T\right\}$. In the estimation, we run the BCD algorithms based on 10 initial $\mathbf{Y}_0$ values and the same tuning parameters $\lambda_1 = 7, \lambda_2 = 0.3, \lambda_3 = 10$ and $\lambda_4 = 0.3$. The BCD algorithm terminates when the objective value decreases less than 0.01% in one iteration, and the threshold to the primal and dual residual in the D-step and Y-step, respectively, are both set to be 10^{-4}. The computational time for the nine setups to achieve this prescribed solution accuracy are 1788 s, 803 s, 2866 s, 768 s, 2796 s, 2392 s, 2640 s, 1399 s, and 1891 s. Thus, computational time is longer if either T or K is larger. As a comparison, we also applied the k-means method to each case, where we always select the number of clusters as $K + 1$, representing the clusters of K types of events and the cluster with no event that occurs. As noted in the literature review, the model assumption of the k-means method cannot identify two events that are overlapping, and it cannot specify the strength of the event. This specific benefit of the MEIC will be illustrated in the detailed results below.

We now investigate the convergence behavior of the MEIC method for the basic setup. In Setup 1, Figure 4(a) illustrates the decrease of objective values of the optimization problem after every **B**-update and **Y**-update in the BCD iterations. This figure shows that the objective value decreases monotonically, and the algorithm converges with few BCD iterations. Each line in Figure 4(b) illustrates the change of primal and dual residual errors in the logarithm scale in each individual **B**-update and **Y**-update process (colored version available online). We can see that the ADMM consensus algorithm converges rapidly in each **B**-update and **Y**-update step. The convergence behavior of the Setups 2 to 9 is similar.

4.2.1. Identification of event sequences

We match the estimated event $k = 1, ..., K$ with a true event k' by finding $\text{argmin}_{k'} \left\{\sum_{i=1}^{I} \|\mathbf{Y}_{k',\cdot} - \hat{\mathbf{Y}}_{k,\cdot}\|_2^2\right\}$. Figure 5 shows the estimation of the event sequences for Setup 1 using both the MEIC method and the k-means method. We can see for the MEIC method that the sequences of the estimated event strengths are very similar to their true values, although the magnitudes are slightly lower, due to the smoothing effect. However, for the k-means approach, we can see that the strength of the events is not estimated, and each time point is associated with at most one type of event. For this reason, the result of event labeling is subject to errors whenever multiple events occur simultaneously, as shown in the box of Figure 5.

To give a numerical performance measure of the error identification, we find all the time points that are identified as associated with event k for the MEIC method, i.e., $\hat{E}_k = \left\{t : \hat{Y}_{k,t} > 0\right\}$. Then, we compare the set $\hat{E}_k$ with E_k, the set of time points where event k indeed occurs, $E_k = \{t : Y_{k,t} > 0\}$. The time points misidentified as not event k are subject to the type I error, and a time point misclassified as event k is subject to the type II error. They can be represented as $E_k - \hat{E}_k$ and $\hat{E}_k - E_k$, respectively, and thus the type I and type II error rates are

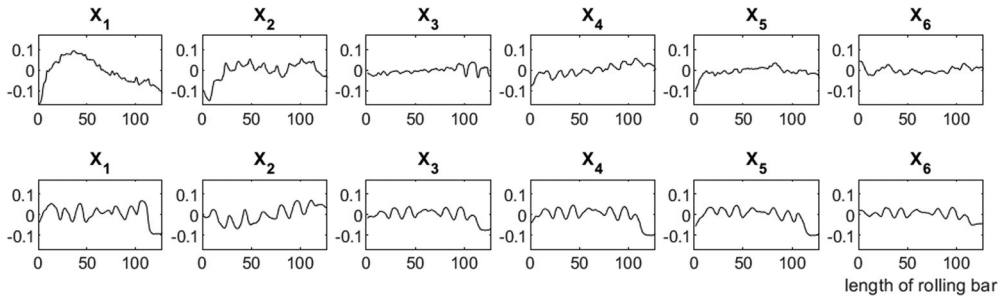


Figure 9. The event signatures on six signals for event 1 (first row) and event 2 (second row).

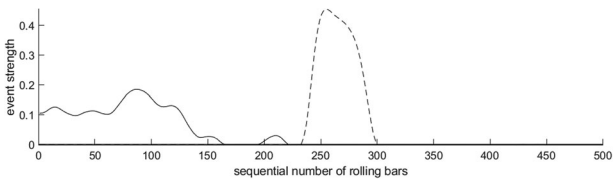


Figure 10. The sequence of event 1 (solid line) and event 2 (dashed line).

$$V_{Y,1} = \sum_{k=1}^{K} \#\{E_k - \hat{E}_k\}/TK \text{ and } V_{Y,2} = \sum_{k=1}^{K} \#\{\hat{E}_k - E_k\}/TK.$$

We list both values and their sum in the rows corresponding to the MEIC method in Table 2.

We also compare the results of type I, type II, and total error rates with those of the k-means method. The results are listed in Table 2. From the comparison, we find that the total error rate of the k-means method is significantly larger than that of the MEIC. Although we find that the type II error rate for the k-means approach is frequently zero, it should not be regarded as an advantage of the k-means method. As the strength of the event is not considered by the k-means method, it only identifies events with large strengths. On the contrary, many time points with small event strengths, including those time points without any events, are identified into the cluster of no event occurrences. This leads to zero type II errors for multiple setups for the k-means method.

Note that the tuning parameters determine the trade-off between type I and type II error rates. For fairness in comparisons, we select the same parameter λ_4 in all setups, and thus we mainly compare the total error. We can see that the error rate does not exceed 2% for all setups from the result. Setup 8, corresponding to the highest frequency of occurrence, has the largest error rate of 1.56%. In all setups, we found that misidentification occurs when an event appears or disappears. Therefore, a possible reason for the large misidentification rate of Setup 8 is that high frequency of event periods associates with multiple events occurrence and disappearance.

4.2.2. Characterization of event signatures

Figure 6 shows the reconstructed event signatures corresponding to events $k = 1, 2, 3$ on curve $i = 1, \ldots, 10$ for Setup 1, obtained from the MEIC method. Plots in each row correspond to an event. The first three subfigures in each row illustrate the estimation of the event signature on three signals that are affected by this event. These signals are marked by S1–S7 in the respective plot titles. We can see that the dashed lines (representing the estimated event signature on these signals) are very close to the solid lines (representing the true event signature on these signals). The fourth subfigure in each row illustrates the estimation of the event signature on signals not affected by an event. The dashed lines, representing the estimated event signature on irrelevant signals, are all close to zero. This result shows that the event signatures are estimated accurately and that signals that are truly associated with an event can be identified correctly.

After estimating the event signatures, we evaluate the estimation accuracy of those event signatures by the mean squared error

$$V_\xi = \frac{1}{KI} \sum_{k,i} \|\xi_{k,i} - \hat{\xi}_{k,i}\|_2^2,$$

as shown in Table 3. For Setup 1, the value of V_ξ is 0.0010. We calculated the value V_ξ for all nine setups, and find that Setup 5 (the case corresponding to $K = 4$) has the largest $V_\xi = 0.0089$, whereas in all other cases, the values of V_ξ do not exceed 0.0040. This shows that the MEIC algorithm characterizes the event signatures accurately.

4.2.3. Importance of multiple starting points

In our simulation study, we discover that the result of the MEIC algorithm is sensitive to the initial value of $\mathbf{Y}^0$. Occasionally, the algorithm may converge to a local minimum, in which two identified events correspond to one real event and have similar event signatures. Similar behavior also occurs to prototype methods like Gaussian Mixture Model and the k-means clustering. Therefore, we perform the optimization algorithm starting from multiple initial values of $\mathbf{Y}^0$ and pick the solution with minimal objective values. In this way, we find that the MEIC algorithm is able to get excellent identification results in every experiment we performed.

5. Case study

In steel rolling processes, the shape uniformity of the rolling bars is an important quality characteristic (Roberts, 1983). In a rolling production line, a laser gauge is installed for in-situ measurement of the cross-sectional shapes of a rolling

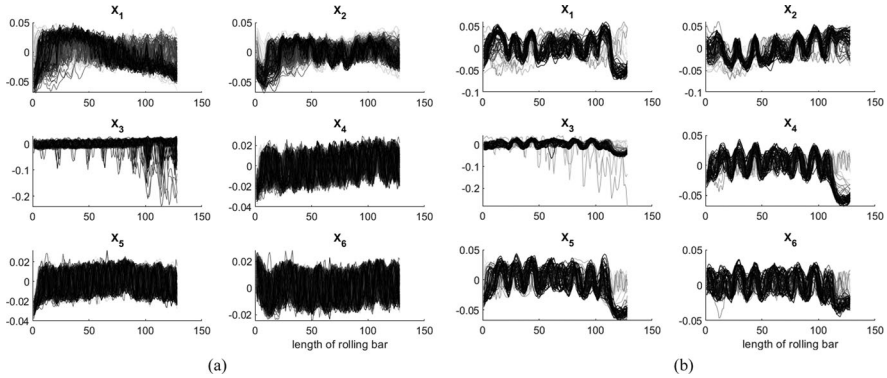


Figure 11. The sample signals that are associated with event 1 (a) and event 2 (b).

bar, which generates six profiles $x_{t,1}(s), \ldots, x_{t,6}(s)$ to represent the diameter measurements along six axes of every rolling bar t. Each $x_{t,i}(s)$ $(i = 1, \ldots, 6)$ is a functional curve that reflects the dimension from the beginning to the end of each rolling bar. The rolling process and the laser gauge measurements are illustrated in Figure 7.

In this case study, the dimensional profile measurements are obtained from a continuous production of $n = 500$ rolling bars. By analyzing those data, we would like to find out if there are any special events that occurred during the production and further identify the specific events and their signatures. To serve this purpose, we apply the MEIC algorithm to this data set.

The data sequence corresponding to each rolling bar has the length $d = 128$. After certain preprocessing steps, we obtain the tensor data $\mathcal{X} \in \mathbb{R}^{128\times6\times500}$, illustrated in Figure 8, where at each time $t = 1, \ldots, 500$ we obtain six curves of length 128, denoting the diameter measurements of the rolling bar along Axes 1-6. The data is then reshaped into the matrix $\mathbf{X} \in \mathbb{R}^{(128\times6)\times500}$.

We set $K = 2$, apply the Haar wavelet as the basis $\mathbf{H}$, and specify the tuning parameters as $\lambda_1 = 0, \lambda_2 = 0.04, \lambda_3 = 500,$ and $\lambda_4 = 0.15$. Here λ_1 is set to zero because there is no need to specify that an event affects only a subset of sensing signals: the abnormal condition in the rolling process will affect the entire cross-section of a rolling bar, and thus the diameter measurements of all Axes 1-6 will be impacted simultaneously. After using the MEIC algorithm to analyze the data set, we obtain the estimated matrics $\mathbf{B}$ and $\mathbf{Y}$. From the matrix $\mathbf{B}$, we recover the event signatures on six signals, as illustrated in Figure 9, and identify the event sequences as in Figure 10. From Figure 9, we can conclude that two abnormal events have occurred in the rolling process. The first event is associated with an increased diameter along Axis 1 near the first quarter of the billet and an increasing diameter along Axis 4. The second event relates to a sharp drop in the diameter value along Axes 1, 3, 4, and 5. In Figure 10, the solid line represents the first event, and the dashed line represents the second event. It tells us that event 1 occurs during the fabrication of the first 150 billets, and event 2 occurs between the 240th to 290th rolling bars.

To validate the existence of those two events and their characteristics, we draw the preprocessed signals obtained when each event occurs in Figure 11(a) and (b). Those two figures correspond to event 1 and event 2, respectively. In each plot, a curve represents the measurements along one sample rolling bar. The gray level indicates the estimated strength of events 1 or 2 for this bar. Comparing Figure 11 with Figure 9, we can observe that those two events are indeed associated with the event signatures we estimated, as the patterns of the dark curves are very similar to the event signatures.

6. Conclusion

It is common to have multiple sensors installed in a system to faithfully record its operation status and generate structured data streaming. Retrospective analysis of those sensing data enables a better understanding of the system status during its operation and allows insights to be gained on new events that affect the system performance. A common question in this retrospective data analysis is to ask if there are some special events (e.g., machine failure, material change, process perturbation, etc.) that occurred during the production time period. If so, what is the event that has occurred? When did it occur? What is the duration of the event? Which sensing signals did the event affect? And what are the event signatures shown on the related sensing signals? Answering those questions will enable the development of more effective monitoring and diagnosis tools for process control and quality improvements. These questions motivate us to define *events* and associated *event signatures* in a system and to develop an automatic tool for identifying and characterizing those events from the system operational data.

The MEIC algorithm simultaneously identifies the occurrence of each event during the system operation and characterizes how each event impacts sensing signals by estimating the event signatures. Our approach has two major advantages: First, it does not require labeled observations corresponding to every single event, which typically involves practitioners going through the data stream and label the abnormal segment in sensing data. Second, it allows the anomaly identification and characterization in a single step to streamline the analysis procedure and focus on vital quality issues. Therefore, this approach leads to a deep

understanding of the system and its operations based on the sensing signals.

In our simulation study, we verify the validity of the MEIC method in identifying event sequences and characterizing event signatures. The algorithm effectively avoids suboptimal points with our scheme of generating random initial values from the computational aspect. Both the BCD algorithm and the inner loops of the ADMM algorithms converge rapidly. The case study successfully identified two events from a set of dimensional sensing data of rolling bars and simultaneously characterized how each event affects the dimension of the rolling bar. As for future research, one direction is to perform process monitoring based on the result of the MEIC method, and another direction is to develop an efficient algorithm to update the model based on new data generated from the system.

Funding

This research is funded by the National Science Foundation Award ID 2019378.

Notes on contributors

Andi Wang received a BS in statistics from Peking University in 2012, a PhD in industrial engineering from Hong Kong University of Science and Technology in 2016, an MS in computer science and engineering from Georgia Institute of Technology in 2021, and a PhD in industrial Engineering from Georgia Institute of Technology in 2021. He is now an assistant professor in the Ira A. Fulton Schools of Engineering, Arizona State University. Andi's research interests include smart manufacturing, data fusion and data-driven systems modeling, process monitoring, and root cause diagnostics. He is a member of Institute of Industrial and Systems Engineers (IISE) and Institute for Operations Research and the Management Sciences (INFORMS).

Tzyy-Shuh Chang received BS, MS, and PhD degrees in mechanical engineering from National Taiwan University (1987), the Ohio State University (1991), and the University of Michigan (1995), respectively. He co-founded OG Technologies, Inc., a Michigan corporation, and led the company to be the globally leading supplier and brand of advanced surface inspection equipment, an R&D100 awardee, for the steel industry. Since its inception, OG Technologies has established its business by way of advanced research and development and cooperative activities with the academia and metal industry, focusing on bringing the state-of-the-art technologies in sensing and data analytics to the millennium old industry.

Jianjun Shi received BS and MS degrees in automation from the Beijing Institute of Technology in 1984 and 1987, respectively, and a PhD degree in mechanical engineering from the University of Michigan in 1992. Currently, Dr. Shi is the Carolyn J. Stewart Chair and Professor at the Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology. 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. Dr. Shi is a Fellow of the Institute of Industrial and Systems Engineers (IISE), a Fellow of American Society of Mechanical Engineers (ASME), a Fellow of the Institute for Operations Research and the Management Sciences (INFORMS), a Fellow of Society of Manufacturing Engineers (SME), an elected member of the International Statistics Institute, a life member of the American Statistics Association (ASA), an Academician of the International Academy for Quality (IAQ), and a member of National Academy of Engineers (NAE).

References

Campbell, N.A. (1980) Robust procedures in multivariate analysis I: Robust covariance estimation. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **29**, 231–237.

Donoho, D.L. and Johnstone, I.M. (1995) Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, **90**, 1200–1224.

Ebrahimi, S., Ranjan, C. and Paynabar, K. (2020) Monitoring and root-cause diagnostics of high-dimensional data streams. *Journal of Quality Technology* (in press and online available).

Friedman, J., Hastie, T. and Tibshirani, R. (2001) *The Elements of Statistical Learning*, Springer, New York, NY.

Guo, R., Guo, K. and Dong, J. (2016) Phase partition and online monitoring for batch process based on multiway BEAM. *IEEE Transactions on Automation Science and Engineering*, **14**, 1582–1589.

Jin, J. and Shi, J. (1999) Feature-preserving data compression of stamping tonnage information using wavelets. *Technometrics*, **41**, 327–339.

Koh, C.K., Shi, J., Williams, W. and Ni, J. (1999) Multiple fault detection and isolation using the Haar transform, part 2: Application to the stamping process. *AMSE Transactions, Journal of Manufacturing Science and Engineering*, **121**, 295–299.

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

Li, F., Shi, Y., Shinde, A., Ye, J. and Song, W. (2019) Enhanced cyber-physical security in internet of things through energy auditing. *IEEE Internet of Things Journal*, **6**, 5224–5231.

Mou, S., Wang, A., Zhang, C. and Shi, J. (2021) Additive tensor decomposition considering structural data information. *IEEE Transactions on Automation Science and Engineering* (in press and online available).

Parikh, N. and Boyd, S. (2014) Proximal algorithms. *Foundations and Trends® in Optimization*, **1**, 127–239.

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**, 1235–1247.

Roberts, W.L. (1983) *Hot Rolling of Steel*, CRC Press, New York, NY.

Ross, S.M. (1996) *Stochastic Processes*, John Wiley, New York, NY.

Wang, W., He, L., Markham, P., Qi, H., Liu, Y., Cao, Q.C. and Tolbert, L.M. (2014) Multiple event detection and recognition through sparse unmixing for high-resolution situational awareness in power grid. *IEEE Transactions on Smart Grid*, **5**, 1654–1664.

Wang, Y., Mei, Y. and Paynabar, K. (2018) Thresholded multivariate principal component analysis for phase I multichannel profile monitoring. *Technometrics*, **60**, 360–372.

Woodall, W.H. and Montgomery, D.C. (2014) Some current directions in the theory and application of statistical process monitoring. *Journal of Quality Technology*, **46**, 78–94.

Yan, H., Paynabar, K. and Shi, J. (2017) Anomaly detection in images with smooth background via smooth-sparse decomposition. *Technometrics*, **59**, 102–114.

Yan, H., Paynabar, K. and Shi, J. (2018) Real-time monitoring of high-dimensional functional data streams via spatio-temporal smooth sparse decomposition. *Technometrics*, **60**, 181–197.

Zhang, C., Yan, H., Lee, S. and Shi, J. (2018) Weakly correlated profile monitoring based on sparse multi-channel functional principal component analysis. *IISE Transactions*, **50**, 878–891.

Zhao, C. (2013) A quality-relevant sequential phase partition approach for regression modeling and quality prediction analysis in manufacturing processes. *IEEE Transactions on Automation Science and Engineering*, **11**, 983–991.