

Modeling and monitoring of a multivariate spatio-temporal network system

Di Wang, Fangyu Li & Kaibo Liu

To cite this article: Di Wang, Fangyu Li & Kaibo Liu (2021): Modeling and monitoring of a multivariate spatio-temporal network system, IISE Transactions, DOI: 10.1080/24725854.2021.1973157

To link to this article: <https://doi.org/10.1080/24725854.2021.1973157>



Published online: 30 Nov 2021.



Submit your article to this journal 



Article views: 91

[View related articles](#) View Crossmark data

Modeling and monitoring of a multivariate spatio-temporal network system

Di Wang^a , Fangyu Li^b , and Kaibo Liu^c 

^aDepartment of Industrial Engineering and Management, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China; ^bFaculty of Information Technology, Beijing Key Laboratory of Computational Intelligence and Intelligent System, Engineering Research Center of Digital Community, Ministry of Education, and Beijing Laboratory for Urban Mass Transit, Beijing University of Technology, Beijing, China; ^cDepartment of Industrial and Systems Engineering, University of Wisconsin-Madison, Madison, WI, USA

ABSTRACT

With the development of information technology, various network systems are created to connect physical objects and people by sensor nodes or smart devices, providing unprecedented opportunities to realize automated interconnected systems and revolutionize people's lives. However, network systems are vulnerable to attacks, due to the integration of physical objects and human behaviors as well as the complex spatio-temporal correlated structures of the network systems. Therefore, how to accurately and effectively model and monitor a network system is critical to ensure information security and support system automation. To address this issue, this article develops a multivariate spatio-temporal modeling and monitoring methodology for a network system by using multiple types of sensor signals collected from the network system. We first propose a Multivariate Spatio-Temporal Autoregressive (MSTA) model by integrating a Gaussian Markov Random Field and a vector autoregressive model structure to characterize the spatio-temporal correlation of the network system. In particular, we develop an iterative model learning algorithm that integrates the Bayesian inference, least squares, and a sum square error-based optimization method to learn the network structure and estimate parameters in the MSTA model. Then, we propose two spatio-temporal control schemes to monitor the network system based on the MSTA model. Numerical experiments and a real case study of an IoT network system are presented to validate the performance of the proposed method.

ARTICLE HISTORY

Received 6 January 2021

Accepted 9 August 2021

KEYWORDS

Multivariate spatio-temporal autoregressive model; spatio-temporal control schemes; network structure learning; IoT network system

1. Introduction

With the development of information technology, various network systems have been created to connect physical objects and people using sensor nodes or smart devices, which provides unprecedented opportunities to revolutionize the way people live their lives including business, education, entertainment, and healthcare (Kehoe *et al.*, 2015; Zhou *et al.*, 2016). For example, the Internet of Things (IoT) is a network system that creates a real-time interaction platform between physical “things” and people. The IoT enables a key technology to realize automated systems including smart cities, smart healthcare, and smart homes, which has the great potential of bringing significant benefits in economic competitiveness, quality of life, public health, and essential infrastructure (Liu and Shi, 2015). Other examples, such as machine networks (Kontar *et al.*, 2016) for smart manufacturing, traffic networks (Xian *et al.*, 2020) for smart transportation, and thermal networks for smart agriculture (Wang *et al.*, 2020a) have also been successfully applied in a wide range of domains.

A network system consists of distributed sensor nodes (or smart devices) with a certain network structure that collects signals to reflect the status of the system. In this sense, a network system is a spatio-temporal stochastic system that is

represented by time-varying signals at each sensor node (or smart device) in the spatial domain. In most engineering practices, to fully reflect the status of a network system, multiple types of sensors are simultaneously deployed at each sensor node to collect multiple types of signals, and each type of signal commonly contains partial information on the status of the network system. Therefore, a network system can be considered as a multivariate spatio-temporal system, that is, the network system has multiple types of sensor signals with a certain network structure and each type of signal varies with space and time. As a particular example, Figure 1 shows an IoT testbed for running a distributed seismic ambient noise tomography imaging program, which is a network system built by six wireless Cyber-Physical System (CPS) units. Spatial information is reflected based on the node relations of the CPS units. In the IoT network system, three types of cyber and energy consumption data including network transmission, power, and CPU are collected over time at each CPS unit. These three types of data are critical indicators to present states of the IoT network system.

Network systems face information security challenges, such as attacks, because of the integration of physical objects and human behaviors. Due to multivariate spatio-temporal

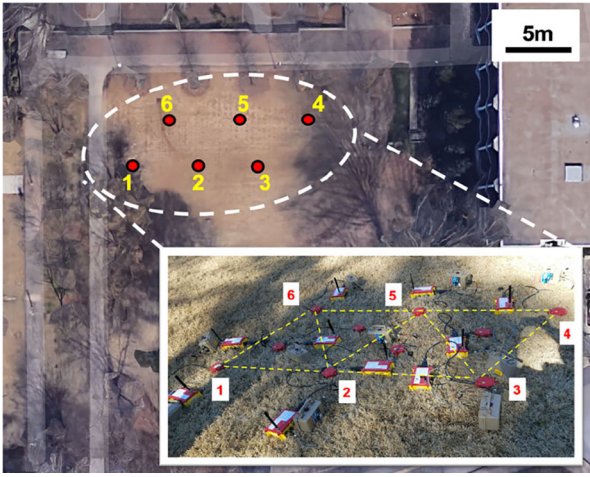


Figure 1. A wireless IoT network constructed by six CPS units. (Note: the dotted yellow lines only indicate a possible spatial relationship.)

correlated structures of a network system, attacks may spread among the network system in space and time, which dramatically increases the vulnerabilities and complexities of the attacks in the network system, leading to a great number of unnecessary economic and social losses as a result. According to a survey for IoT network systems (ISS Source, 2019). In 2018, 80% of industrial organizations experienced attacks on their IoT devices, and suffered substantial losses due to the attacks. The average financial impact for each organization was shown to be more than \$330,000. Therefore, to ensure information security and support system automation of network systems, accurately modeling and monitoring of a network system is critical to fully understand the multivariate spatio-temporal characteristics of the network system and further achieve a timely and effective detection of attacks occurring in the system.

In recent years, many methods have been developed to model and monitor network systems. For the modeling of network systems, statistical approaches have been employed to describe spatio-temporal characteristics. For example, kriging methods (Wang and Zhang, 2019), sparse matrix algorithms (Furrer *et al.*, 2006), reduced rank techniques (Cressie and Johannesson, 2008), and full-scale approximation methods (Zhang *et al.*, 2015) use covariance functions to characterize spatio-temporal correlations of simple systems. The Gaussian Markov Random Field (GMRF) models (Xu and Choi, 2012; Xu and Huang, 2012; Wang *et al.*, 2019) and Spatio-Temporal Conditional Auto-Regressive (STCAR) models (Mariella and Tarantino, 2010; Wang *et al.*, 2020b) characterize spatio-temporal correlations of complex systems based on a single type of sensor signals. For the monitoring of network systems, control charts are the most commonly used methods, such as Shewhart charts (Montgomery, 2009), Exponentially Weighted Moving Average (EWMA) charts (Lucas and Saccucci, 1990), and Cumulative Sum (CUSUM) charts (Crosier, 1998). The supervised learning models including the Neural Network (NN) (Li *et al.*, 2019), Support Vector Machine (SVM) (Liou *et al.*, 2018), Decision Tree (DT) (Moon *et al.*, 2017) and Random Forest (RF) (Ambikavathi and Srivatsa, 2020),

the unsupervised learning models including Self Organizing Map (SOM) (Altman, 1992) and K-Nearest Neighbors (KNN) (Vesanto and Alhoniemi, 2000), as well as the signal processing models including control charts of multivariate Wavelet Transform (WT) features (Wu and Wang, 2011) have also been applied for change point detection. These existing methods mainly monitor a network system based on a single type of sensor signals directly without fully capturing spatio-temporal correlations of the network system. In addition, existing models for the network modeling and monitoring are established under the assumption that the network structure is often known. Modeling and monitoring a network system still face significant challenges as follows: First, the network system is a complex multivariate spatio-temporal system that simultaneously varies with space and time. On the one hand, spatio-temporal characteristics should be considered in the modeling of the network system. On the other hand, the network system contains multiple types of signals at each sensor node. Although spatio-temporal systems with a single type of signals have been studied in recent years, the modeling of multivariate spatio-temporal systems with multiple types of signals remains a challenging task. Second, conventional statistical process control methods (e.g., control charts) are designed without fully capturing spatio-temporal correlations of the network system, and are only applicable for cases when observations of the monitored system are independent at different time points. For network systems that are correlated in the time domain, effective monitoring of the network systems becomes a challenge. Third, in a network system, the network structure reflects if any two sensor nodes are interconnected in the spatial domain. However, in most engineering cases, the network structure is unknown, which increases the difficulty of modeling and monitoring of the network system.

In this article, we develop a multivariate spatio-temporal modeling and monitoring methodology for a network system by using multiple types of sensor signals collected from the network system. Since the network system is a multivariate spatio-temporal system, we first propose a Multivariate Spatio-Temporal Autoregressive (MSTA) model to fully characterize the spatio-temporal correlation of the network system. The proposed MSTA model integrates a GMRF and a vector autoregressive structure that fully considers the spatial correlation under the network structure and the temporal correlation among multiple types of signals. Specifically, we develop an iterative model learning algorithm that integrates the Bayesian inference, least squares, and a sum square error-based optimization method to learn the network structure and estimate parameters in the MSTA model. After characterizing the spatio-temporal correlation of the network system, we then propose the multivariate spatio-temporal control schemes to monitor the network system. Compared with existing control schemes for network systems, the developed control schemes construct control statistics by excluding the temporal dependence from the spatio-temporal characteristics, and thus can monitor the network system more effectively.

The remainder of this article is organized as follows. Section 2 provides the literature review on spatio-temporal modeling and monitoring of network systems. Section 3 introduces the proposed method including the multivariate spatio-temporal modeling of a network system, parameter estimation, network structure learning, and the monitoring of the network system. Sections 4 and 5 present a number of numerical experiments and a case study of an IoT network system. Section 6 provides a conclusion and suggestions for future research.

2. Literature review

The goal of this study is to model and monitor a network system by fully considering multivariate spatio-temporal characteristics. In this section, we review relevant studies about the modeling and monitoring of a network system, respectively.

2.1. Modeling of a network system

A network system is a multivariate spatio-temporal process. In recent years, numerous statistical approaches have been developed to model spatio-temporal processes. Conventional studies have used covariance functions to describe spatio-temporal correlations, including kriging methods (Wang and Zhang, 2019), sparse matrix algorithms (Furrer *et al.*, 2006), reduced rank techniques (Cressie and Johannesson, 2008), and full-scale approximation methods (Zhang *et al.*, 2015). However, these methods are only applicable to simple systems, as they describe spatio-temporal correlations just by a covariance function. The modeling accuracy of these methods for complex systems is unsatisfactory, because complex spatio-temporal correlations in the systems cannot be adequately captured by the covariance function.

To address this issue, studies that combine spatial and temporal modeling methods have been conducted to characterize spatio-temporal correlations of complex systems. For spatial modeling, the GMRF has received considerable attention in recent years. For example, the GMRF, which considers grid-based neighborhood structures to model spatial correlation of a complex system, has been well validated in nanowire growth (Xu and Huang, 2012). Xu and Choi (2012) proposed a Gaussian process built on a GMRF to model mobile distribution given resource-constrained mobile sensor networks. Wang *et al.* (2019) integrated a kriging model into a GMRF model to estimate the thermal field of a grain storage system. These GMRF models have been widely applied to characterize spatial correlations, but fail to model temporal correlations. By extending the GMRF, Mariella and Tarantino (2010) and Liu *et al.* (2018) proposed an STCAR model, which is an autoregressive model for a temporal sequence of GMRFs, to characterize spatio-temporal correlations. However, the GMRF and STCAR mainly model spatio-temporal systems based on a single type of signals. To the best of our knowledge, few studies have modeled multivariate spatio-temporal systems based on multiple types of signals. In addition, when

modeling a network system, the GMRF and STCAR are established under the assumption that the network structure is known. For engineering cases where the network structure is not known, the GMRF and STCAR are not applicable.

For structure learning, several methods, including the Bayesian regression graphical model (Dobra *et al.*, 2004), Gaussian graphical model (Lin *et al.*, 2017), graphical lasso algorithm (Friedman *et al.*, 2008), and Min-Max conditional covariance algorithm (Gao and Ye, 2019), have been developed to select neighborhoods and learn graphical structures. However, few of these studies have integrated spatio-temporal characteristics to model multivariate spatio-temporal processes.

2.2. Monitoring of a network system

Different types of Statistical Process Control (SPC) charts have been extensively adopted for monitoring processes to quickly detect attacks in recent years (Cannon *et al.*, 2011), such as Shewhart charts (Montgomery, 2009), EWMA charts (Lucas and Saccucci, 1990), and CUSUM charts (Crosier, 1998). CUSUM charts serve as a popular tool in many process monitoring methods (Woodall, 2006). Specifically, Multivariate CUSUM (MCUSUM) charts have been presented to monitor spatial and temporal processes (Spiegelhalter *et al.*, 2012). By considering spatial correlation among locations, MCUSUM charts use process observations to formulate statistics including Hotelling's T^2 statistics and Log-likelihood Ratio (LR) statistics. For example, Boullosa-Falces *et al.* (2017) obtained Hotelling's T^2 statistics using sensor data at each time point and then formed a CUSUM chart based on a time sequence of the Hotelling's T^2 statistics. Jiang *et al.* (2011) and Lee *et al.* (2012) adopted an LR statistic for each cluster in the spatial domain and scanned all clusters to detect underlying attacks. However, these MCUSUM charts are established under the assumption that the monitored process is independent in the temporal domain. Ignoring temporal dependence of the monitored processes will lead to untimely and inaccurate monitoring results of the MCUSUM charts. By considering temporal dependence, Gombay and Serban (2009) considered the problem of testing for parameter changes in time series models based on a CUSUM test. Shao and Zhang (2010) proposed a normalization-based Kolmogorov-Smirnov CUSUM test for change point detection in time series. Bodnar and Schmid (2017) developed CUSUM control schemes for monitoring the covariance matrix of multivariate time series. Although these CUSUM-based tests can monitor processes by fully considering temporal dependence, these tests are only applicable to time series models and cannot be used to monitor spatio-temporal processes. To the best of our knowledge, few researchers have developed control schemes that fully address the spatio-temporal characteristics.

To fill the research gap, we propose a multivariate spatio-temporal modeling and monitoring methodology tailored to a network system. The proposed method addresses the challenges of modeling and monitoring of the network system

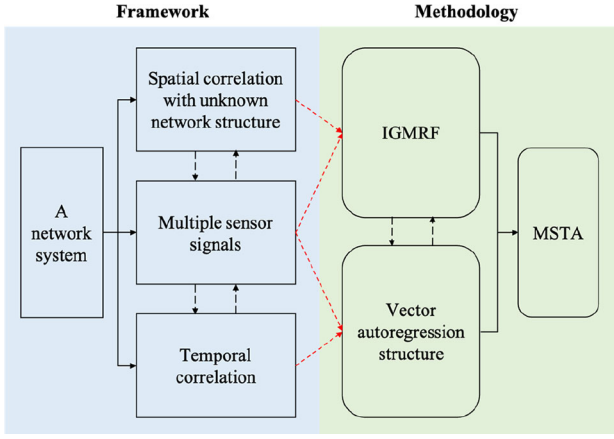


Figure 2. The formulation of the proposed MSTA model.

in the following aspects: First, we propose an MSTA model for a multivariate spatio-temporal process that can fully characterize the spatio-temporal correlation of the network system. Specifically, the proposed model integrates a GMRF and a vector autoregressive structure to describe the spatial and temporal characteristics of the network system based on multiple types of signals. Second, we develop multivariate spatio-temporal control schemes to monitor the network system based on the proposed MSTA model. Compared with existing control schemes, the developed control schemes fully consider the spatio-temporal characteristics of the network system and construct control statistics by excluding temporal dependence from the spatio-temporal characteristics to monitor the network system more efficiently. Third, we propose a Bayesian inference method to learn the neighborhood network structure of the network system. The learnt network structure can reflect the inter-connection of sensor nodes and provide useful information for engineering practice.

3. Methodology

3.1. Multivariate spatio-temporal modeling of a network system

The key to modeling a network system is to fully capture its spatio-temporal characteristics. On the one hand, given that multiple types of sensor signals are collected, and each type of signal reflects partial information of the network system, fully utilizing the multiple types of sensor signals to better characterize the spatio-temporal correlation of the network system is essential to the modeling of the network system. On the other hand, the network structure provides structural information of the network system and captures spatial characteristics, i.e., it reflects if any two sensor nodes are interconnected in the spatial domain. Thus, when modeling the network system, we need to accurately learn the network structure.

In a network system, multiple types of sensor signals are collected that vary with space and time. We propose an MSTA model to fully characterize the multivariate spatio-temporal correlation of the network system. Figure 2 presents the formulation of the proposed MSTA model. As

shown in the figure, the proposed MSTA model integrates an Improved GMRF (IGMRF) and a vector autoregressive structure to fully characterize the spatio-temporal correlation under the network structure among multiple types of signals. In particular, since the neighborhood network structure is generally unknown in practice, it can lead to the inefficiency of the GMRF for spatial modeling of the network system. We propose an IGMRF that learns the neighborhood network structure using multiple types of signals to address this issue. Without loss of generality, we assume that there are n sensor nodes in the spatial domain $\mathcal{S}$ of the network system, and there are L types of signals at each sensor node. We represent the network system as an L -variate spatio-temporal process $\{y_t^l(\mathbf{s}), t \in \mathbb{R}^+, \mathbf{s} \in \mathcal{S} \subseteq \mathbb{R}^n, l = 1, \dots, L\}$, where $\mathbf{s}$ is the location of a sensor node in the spatial domain $\mathcal{S}$ and l is the index of the signal type.

At each time t , we denote the spatial process for the l th signal type as $\mathbf{Y}_t^l = [y_t^l(\mathbf{s}_1), \dots, y_t^l(\mathbf{s}_n)]^T \in \mathbb{R}^{n \times 1}$, with $l = 1, \dots, L$. The IGMRF for the spatial modeling of $\mathbf{Y}_t^l$ can learn the network neighborhood structure as well as characterize the spatial correlation of $\mathbf{Y}_t^l$. Specifically, for each sensor node $\mathbf{s}_i$, we define its neighbors $\mathbf{s}_j$ and specify the conditional distribution of $y_t^l(\mathbf{s}_i)$ as

$$y_t^l(\mathbf{s}_i) | \{y_t^l(\mathbf{s}_j)\}_{\mathbf{s}_j \sim \mathbf{s}_i, j \neq i} \sim N\left(\sum_{\mathbf{s}_j \sim \mathbf{s}_i} \beta_{ij}^0 y_t^l(\mathbf{s}_j), \sigma_i^2(\mathbf{s}_i)\right), \quad (1)$$

where $\mathbf{s}_j \sim \mathbf{s}_i$ denotes $\mathbf{s}_j$ is the neighbor of $\mathbf{s}_i$; β_{ij}^0 is a spatial dependence coefficient that captures the spatial correlation between $y_t^l(\mathbf{s}_i)$ and $y_t^l(\mathbf{s}_j)$ with $\beta_{ii}^0 = 0$, and a larger β_{ij}^0 means a stronger spatial correlation between $y_t^l(\mathbf{s}_i)$ and $y_t^l(\mathbf{s}_j)$; and $\sigma_i^2(\mathbf{s}_i)$ is the conditional variance of the l th sensor type at node $\mathbf{s}_i$. When the network system is in the normal state, we assume stationary spatial correlation patterns for each sensor node in the spatial domain, that is, $\sigma_i^2(\mathbf{s}_i) = \sigma_l^2$ for all $\mathbf{s}_i$ with $i = 1, \dots, n$, where σ_l^2 denotes the overall variance of the l th signal type. Here, we determine if $\mathbf{s}_j \sim \mathbf{s}_i$ based on the network neighborhood structure. To better represent the network neighborhood structure, we denote γ as the network structure matrix, i.e., γ is an $n \times n$ binary latent state matrix, where $\gamma_{ij} = 1$ ($i \neq j$) represents that $\mathbf{s}_i$ and $\mathbf{s}_j$ are neighbors, and $\gamma_{ij} = 0$ otherwise. We will introduce the learning of the network neighborhood structure in Section 3.2 in detail. We denote the spatial dependence matrix as $\beta_0 = [\beta_{ij}^0(\gamma_{ij})]_{i,j=1,\dots,n} \in \mathbb{R}^{n \times n}$, where $\beta_{ij}^0(\gamma_{ij}) = \beta_{ij}^0$ if $\gamma_{ij} = 1$ and $\beta_{ij}^0(\gamma_{ij}) = 0$ if $\gamma_{ij} = 0$, and denote $\mathbf{B}_0 = \mathbf{I}_n - \beta_0$, where $\mathbf{I}_n$ is an $n \times n$ identity matrix. When $\mathbf{B}_0$ is invertible and $\mathbf{B}_0^{-1}$ is symmetric and positive definite, we obtain from Equation. (1) that $\mathbf{Y}_t^l$ follows

$$\mathbf{Y}_t^l \sim N(\mathbf{0}, \sigma_l^2 \mathbf{B}_0^{-1}). \quad (2)$$

We denote the spatial process for all the L signal types at time t as $\mathbf{Y}_t = [\mathbf{Y}_t^1, \dots, \mathbf{Y}_t^L]^T \in \mathbb{R}^{nL \times 1}$. In this article, we assume $\mathbf{Y}_t^1, \dots, \mathbf{Y}_t^L$ are independent across different signal types. This is a reasonable assumption here since $\mathbf{Y}_t^1, \dots, \mathbf{Y}_t^L$

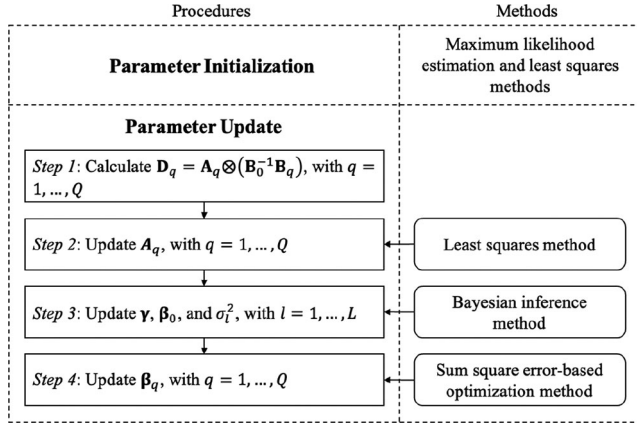


Figure 3. Parameter estimation procedure.

capture the network structure in the spatial domain, respectively. Actually, the correlation of different signal types is mainly reflected in the temporal domain, and we will further capture the correlation of different signal types when modeling temporal characteristics of the network system. Thus, $\mathbf{Y}_t$ follows

$$N\left(\mathbf{0}, \begin{bmatrix} \sigma_1^2 \mathbf{B}_0^{-1} & & \\ & \ddots & \\ & & \sigma_L^2 \mathbf{B}_0^{-1} \end{bmatrix}\right),$$

that is,

$$\mathbf{Y}_t \sim N(\mathbf{0}, \mathbf{C}_\sigma \otimes \mathbf{B}_0^{-1}), \quad (3)$$

where $\mathbf{C}_\sigma = \text{diag}\{\sigma_1^2, \dots, \sigma_n^2\}$ and $\otimes$ denotes the Kronecker product. $\mathbf{Y}_t$ can also be written as a matrix form $\mathbf{Y}_t = \mathbf{I}_L \otimes \beta_0 \mathbf{Y}_t + \boldsymbol{\eta}_t$, where $\mathbf{I}_L$ is an $L \times L$ identity matrix and $\boldsymbol{\eta}_t$ is a vector of the pseudo errors. From $\boldsymbol{\eta}_t = (\mathbf{I}_{Ln} - \mathbf{I}_L \otimes \beta_0) \mathbf{Y}_t = (\mathbf{I}_L \otimes \mathbf{B}_0) \mathbf{Y}_t$, we obtain the distribution of $\boldsymbol{\eta}_t$ as follows:

$$\boldsymbol{\eta}_t \sim N(\mathbf{0}, \mathbf{C}_\sigma \otimes \mathbf{B}_0^T). \quad (4)$$

Then, we construct a time sequence of multivariate spatial processes by integrating the temporal evolution to the above multivariate spatial process model. In Equation (3), we have $\mathbf{Y}_t \sim N(\mathbf{0}, \mathbf{C}_\sigma \otimes \mathbf{B}_0^{-1})$. Similarly, we obtain $\mathbf{Y}_{t-q} \sim N(\mathbf{0}, \mathbf{C}_\sigma \otimes \mathbf{B}_q^{-1})$ for $q = 1, 2, \dots$, where $\mathbf{B}_q = \mathbf{I}_n - \beta_q$. Considering the temporal sequence of multivariate spatial processes from time $t - Q$ to time t , where Q is a positive integer value that denotes the order of the temporal sequence, we establish the framework of MSTA model by integrating the multivariate spatial processes into a vector autoregressive structure to construct the relation between $\mathbf{Y}_t$ and $\mathbf{Y}_{t-1}, \dots, \mathbf{Y}_{t-Q}$ as follows:

$$\mathbf{Y}_t = \sum_{q=1}^Q \mathbf{D}_{t,t-q} \mathbf{Y}_{t-q} + \boldsymbol{\delta}_t, \quad (5)$$

where $q = 1, \dots, Q$, $\mathbf{D}_{t,t-q} = \boldsymbol{\Sigma}_{t,t-q} \boldsymbol{\Sigma}_{t-q,t-q}^{-1}$, and $\boldsymbol{\Sigma}_{t-q,t-q}$ denotes the covariance matrix of $\mathbf{Y}_{t-q}$ with $\boldsymbol{\Sigma}_{t-q,t-q} = \mathbf{C}_\sigma \otimes \mathbf{B}_q^{-1}$. The detailed formulation of the MSTA model in Equation (5) and $\mathbf{D}_{t,t-q}$ are illustrated in Appendix A of supplemental online materials. Note that while we assume

$\mathbf{Y}_t^1, \dots, \mathbf{Y}_t^L$ are independent across different signal types, the correlation of different signal types is captured in the temporal domain when we model the temporal characteristics of the network system in Equation (5). Network systems in engineering practice can be considered as stationary processes when the systems are in the normal state, and the spatial processes of the network systems at consecutive time points $\mathbf{Y}_t, \mathbf{Y}_{t-1}, \dots$, and $\mathbf{Y}_{t-Q}$ are temporal correlated. Thus, we further capture the temporal correlation of the network system by a vector autoregressive model structure with temporal-associated matrix $\mathbf{A}_q$ at time $t - q$, with $q = 1, \dots, Q$, which captures the temporal correlation of multiple types of signals between time $t - q$ and t . In particular, we consider $\boldsymbol{\Sigma}_{t,t-q} = (\mathbf{A}_q \otimes \mathbf{I}_n) \boldsymbol{\Sigma}_{t-q,t-q}$ by inserting the temporal-associated matrix $\mathbf{A}_q$ into the spatial covariance matrix $\boldsymbol{\Sigma}_{t-q,t-q}$ to characterize multivariate spatiotemporal characteristics of the network system. In this way, we have

$$\mathbf{D}_{t,t-q} = \mathbf{A}_q \otimes (\mathbf{B}_0^{-1} \mathbf{B}_q), \quad (6)$$

where the calculation of $\mathbf{D}_{t,t-q}$ is elaborated in Appendix B of supplemental online materials in detail. It should be noted that $\mathbf{A}_q$, with $q = 1, \dots, Q$, is assumed to be invertible and symmetric. We define $\boldsymbol{\delta}_t$ as the vector of pseudo errors $\boldsymbol{\eta}_t$ structured with a spatial component $(\mathbf{I}_L \otimes \mathbf{B}_0)^{-1}$, i.e.,

$$\boldsymbol{\delta}_t = (\mathbf{I}_L \otimes \mathbf{B}_0)^{-1} \boldsymbol{\eta}_t. \quad (7)$$

By inserting Equations (6) and (7) into Equation (5), we obtain the general expression of the MSTA model as follows:

$$(\mathbf{I}_L \otimes \mathbf{B}_0) \mathbf{Y}_t = \sum_{q=1}^Q (\mathbf{A}_q \otimes \mathbf{B}_q) \mathbf{Y}_{t-q} + \boldsymbol{\eta}_t, \quad (8)$$

$$\boldsymbol{\eta}_t \sim N(\mathbf{0}, \mathbf{C}_\sigma \otimes \mathbf{B}_0^T),$$

In the proposed MSTA model, we use a temporal-associated matrix $\mathbf{A}_q$, spatial-associated matrices $\mathbf{B}_0$ and $\mathbf{B}_q$, and the product of a temporal-associated matrix and spatial-associated matrices $\mathbf{A}_q \otimes \mathbf{B}_q$, with $q = 1, \dots, Q$, to characterize the temporal correlation of the multiple types of sensor signals in the same node at different time points, the spatial correlation in nearby nodes at the same time point, and the spatio-temporal interaction of the multiple types of sensor signals in nearby nodes at different time points, respectively.

3.2. Network structure learning and parameter estimation

When the network system is in the normal state, its corresponding multivariate spatio-temporal process follows the above proposed model distribution (Equation (8)). We use the observed sensor data in the normal state with M time points of the network system to learn the network neighborhood structure and estimate parameters of the proposed model. The proposed model contains a set of parameters to be estimated, including the network structure matrix γ , the overall variance σ_l^2 of the l th signal type, with $l = 1, \dots, L$,

the spatial dependence matrices β_0 and β_q , with $q = 1, \dots, Q$, and the temporal-associated matrix A_q , with $q = 1, \dots, Q$. We propose an iterative model learning algorithm to estimate these unknown parameters, which includes two procedures: parameter initialization and parameter update.

As shown in Figure 3, for parameter initialization, we apply maximum likelihood estimation and least squares methods to initialize the model parameters. The procedure initializes the model's temporal and spatial-associated parameters by excluding the spatial dependence and temporal dependence, respectively. The detailed procedure of parameter initialization is provided in Appendix C of the supplemental online materials. For parameter update, we propose and integrate the least squares method, the Bayesian inference method, and the sum square error-based optimization method to respectively estimate the model parameters. Specifically, the parameter update procedure includes four steps: The first step is the calculation of $D_q = A_q \otimes (B_0^{-1} B_q)$, with $q = 1, \dots, Q$, where D_q is considered as an intermediate auxiliary matrix for parameter estimation. The second step is the update of A_q , with $q = 1, \dots, Q$, by the least squares method. In the third step, we propose a Bayesian inference method to learn the network neighborhood structure matrix γ , as well as to estimate the spatial dependence matrix β_0 and the overall variance σ_l^2 of the l th signal type. In the last step, we propose a sum square error-based optimization method to estimate the spatial dependence matrix β_q , with $q = 1, \dots, Q$. The procedure of parameter update is implemented iteratively until convergence. We introduce the parameter update procedure in detail as follows.

- (a) *Step 1: Calculate $D_q = A_q \otimes (B_0^{-1} B_q)$, with $q = 1, \dots, Q$.*
- (b) *Step 2: Update A_q , with $q = 1, \dots, Q$, using the least squares method.*

We rewrite Equation (8) as $G_t = \sum_{q=1}^Q (A_q \otimes I_n) G_{t-q} + \eta_t$, where $G_t = (I_L \otimes B_0) Y_t$ and $G_{t-q} = (I_L \otimes B_q) Y_{t-q}$, with $q = 1, \dots, Q$. We can see from this equation that our proposed model can be considered as a vector autoregressive model framework. Therefore, $A = [A_1, \dots, A_Q]^T$ can be estimated using the least squares method as follows

$$A = (G_*^T G_*)^{-1} G_*^T G_{**}, \quad (9)$$

where

$$G_* = \begin{bmatrix} G_1^1 & \dots & G_1^L & \dots & G_Q^1 & \dots & G_Q^L \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ G_{M-Q}^1 & \dots & G_{M-Q}^L & \dots & G_{M-1}^1 & \dots & G_{M-1}^L \end{bmatrix}$$

is an $(M-Q)n \times QL$ matrix, and

$$G_{**} = \begin{bmatrix} G_{Q+1}^1 & \dots & G_{Q+1}^L \\ \vdots & \ddots & \vdots \\ G_M^1 & \dots & G_M^L \end{bmatrix}$$

is an $(M-Q)n \times L$ matrix. Finally, the estimated value of A_q is obtained from A .

- (c) *Step 3: Network structure learning of γ and parameter estimation of β_0 and σ_l^2 , with $l = 1, \dots, L$, by proposing a Bayesian inference method.*

We calculate $U_t = Y_t - \sum_{q=1}^Q D_q Y_{t-q}$ with $t = 1, \dots, M$, and then propose a Bayesian inference method for network structure learning of γ and parameter estimation of β_0 and σ_l^2 , with $l = 1, \dots, L$, based on U_t . We first introduce prior distributions on γ_{ij} , β_{ij}^0 (with $i, j = 1, \dots, n$), and σ_l^2 (with $l = 1, \dots, L$), respectively. In particular, it is reasonable to assume the prior distribution on γ_{ij} , with $i, j = 1, \dots, n$, is Bernoulli, i.e., $P(\gamma_{ij} = 1) = 1 - P(\gamma_{ij} = 0) = q$. Conditioning on γ_{ij} , β_{ij}^0 is assumed to follow a normal mixture distribution (George and McCulloch, 1997), i.e., $\beta_{ij}^0 | \gamma_{ij} \sim (1 - \gamma_{ij})N(0, \tau_0^2) + \gamma_{ij}N(0, \tau_1^2)$, where q , τ_0^2 , and τ_1^2 are prefixed hyperparameters. We assume an Inverse Gamma (IG) prior on σ_l^2 , with $l = 1, \dots, L$, as $\sigma_l^2 \sim IG\left(\frac{\nu_l}{2}, \frac{\lambda \nu_l}{2}\right)$. Without loss of generality, we set $\nu_l = 0$ in this article, and the IG prior reduces to a flat prior (Li and Zhang, 2010).

Then, the posterior distributions of γ , β_0 , and σ_l^2 , with $l = 1, \dots, L$, are obtained as follows:

• Posterior distribution of β_0

We denote $V_t = \sum_{l=1}^L U_t^l$, and $V = (V_1, \dots, V_M)^T$. The posterior distribution of β_0 is obtained based on V . For the sensor node s_i , with $i = 1, \dots, n$, the subset $O_i = \{1, \dots, n\} \setminus \{i\}$ corresponds to the indices of sensor nodes without s_i . For β_{iO_i} that is the i th column of β_0 excluding the i th element in that column, its posterior distribution $P(\beta_{iO_i} | V_i) \propto P(V_i | \beta_{iO_i}) P(\beta_{iO_i})$ is a multivariate Gaussian distribution

$$\beta_{iO_i} | V_i \sim N\left(\left(V_{O_i}^T V_{O_i} + \sigma_{\text{sum}}^2 \mathbf{d}_i^{-1}\right)^{-1} V_{O_i}^T V_i, \left(\frac{1}{\sigma_{\text{sum}}^2} V_{O_i}^T V_{O_i} + \mathbf{d}_i^{-1}\right)^{-1}\right), \quad (10)$$

where V_i is the i th column of V , V_{O_i} is the corresponding remaining columns of V and $\sigma_{\text{sum}}^2 = \sum_{l=1}^L \sigma_l^2$. We have $\beta_{iO_i} \sim N(0, \mathbf{d}_i)$, where $\mathbf{d}_i$ is an $(n-1) \times (n-1)$ matrix and the j th diagonal entry of $\mathbf{d}_i$ takes the value as

$$(\mathbf{d}_i)_{jj} = \begin{cases} \tau_0^2, & \text{if } (\gamma_{iO_i})_j = 0 \\ \tau_1^2, & \text{if } (\gamma_{iO_i})_j = 1 \end{cases},$$

in which γ_{iO_i} is the i th column of γ excluding the i th element in that column. The detailed information about the posterior distribution $P(\beta_{iO_i} | V_i)$ is introduced in Appendix D of supplemental online materials. We utilize the mean of the posterior distribution $P(\beta_{iO_i} | V_i)$ to update the point estimator of β_{iO_i} , i.e., $\hat{\beta}_{iO_i} = \left(V_{O_i}^T V_{O_i} + \sigma_{\text{sum}}^2 \mathbf{d}_i^{-1}\right)^{-1} V_{O_i}^T V_i$. After $\hat{\beta}_{iO_i}$, with $i = 1, \dots, n$, is obtained, the updated estimate of β_0 can be obtained.

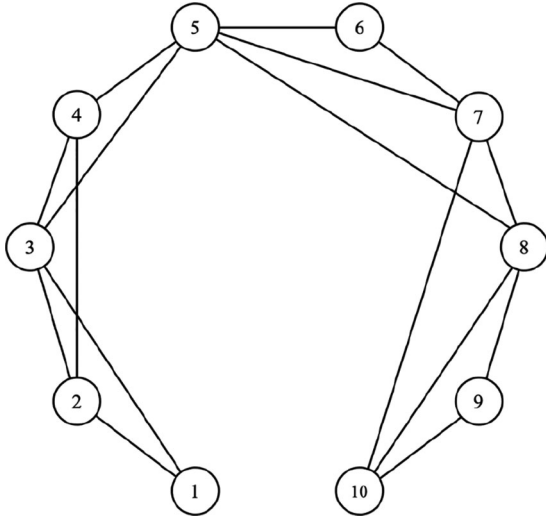


Figure 4. The network structure of systems with sudden changes in mean.

- **Posterior distribution of σ_l^2 , with $l = 1, \dots, L$**

We denote the set of $\mathbf{U}_t^l$, with $t = 1, \dots, M$, as $\mathbf{U}^l = \{\mathbf{U}_1^l, \dots, \mathbf{U}_M^l\}$. The posterior distribution of σ_l^2 , with $l = 1, \dots, L$, i.e., $P(\sigma_l^2 | \mathbf{U}^l) \propto P(\mathbf{U}^l | \sigma_l^2)P(\sigma_l^2)$, is an IG distribution

$$\sigma_l^2 | \mathbf{U}^l \sim IG\left(\frac{1}{2}(\nu_l + Mn), \frac{1}{2}\left(\lambda\nu_l + \sum_{t=1}^M \mathbf{U}_t^{lT} \mathbf{B}_0 \mathbf{U}_t^l\right)\right), \quad (11)$$

where recall we set $\nu_l = 0$ in this article. The detailed derivation of the posterior distribution $P(\sigma_l^2 | \mathbf{U}^l)$ is introduced in Appendix E of the supplemental online materials. We utilize the mean of the posterior distribution $P(\sigma_l^2 | \mathbf{U}^l)$ to update the point estimator of σ_l^2 , i.e.,

$$\hat{\sigma}_l^2 = \frac{1}{2} \left(\lambda\nu_l + \sum_{t=1}^M \mathbf{U}_t^{lT} \mathbf{B}_0 \mathbf{U}_t^l \right) / \left(\frac{1}{2}(\nu_l + Mn) - 1 \right).$$

- **Posterior distribution and updating of γ**

The columns of γ can be updated independently. The posterior distribution of γ_{iO_i} , i.e., $P(\gamma_{iO_i} | \mathbf{V}_i, \beta_{iO_i})$, is

$$P(\gamma_{iO_i} | \mathbf{V}_i, \beta_{iO_i}) \propto P(\beta_{iO_i} | \mathbf{V}_i, \gamma_{iO_i})P(\gamma_{iO_i}). \quad (12)$$

Here $P(\beta_{iO_i} | \mathbf{V}_i, \gamma_{iO_i})$ can be calculated by Equation (10) and $P(\gamma_{iO_i}) = q^{n_{O_i}}(1-q)^{n-n_{O_i}}$, where n_{O_i} denotes the number of the elements that are equal to zero in γ_{iO_i} . We update the point estimator of γ_{iO_i} as $\hat{\gamma}_{iO_i} = \arg\max\{P(\gamma_{iO_i} | \mathbf{V}_i, \beta_{iO_i}), i = 1, \dots, n\}$.

3.3. Multivariate spatio-temporal control schemes

We denote the observed L -variate spatio-temporal process of the network system at time t as $\mathbf{X}_t = [\mathbf{X}_t^1, \dots, \mathbf{X}_t^L]^T \in \mathbb{R}^{nL \times 1}$, where $\mathbf{X}_t^l = [x_t^l(s_1), \dots, x_t^l(s_n)]^T \in \mathbb{R}^{n \times 1}$. To effectively monitor a multivariate spatio-temporal network

system, we monitor the residual part ξ_t at time t obtained from the proposed MSTA model as follows

$$\xi_t = (\mathbf{I}_L \otimes \mathbf{B}_0) \mathbf{X}_t - \sum_{q=1}^Q \mathbf{A}_q \otimes \mathbf{B}_q \mathbf{X}_{t-q}. \quad (13)$$

When $\mathbf{X}_t$ is in the normal state, $\xi_t \sim N(\mathbf{0}, \mathbf{C}_\sigma \otimes \mathbf{B}_0^T)$ and is independent over the temporal domain.

To better monitor the network system, we consider the characteristic quantities for monitoring both the changes of the means and the covariances of the network system as $T_{t,1}$ and $T_{t,2}$, respectively. The characteristic quantity for monitoring the change of the mean $T_{t,1}$ is

$$T_{t,1} = (\mathbf{C}_\sigma \otimes \mathbf{B}_0^T)^{-\frac{1}{2}} \xi_t.$$

The characteristic quantity for monitoring the change of the covariance $T_{t,2}$ is

$$T_{t,2} = \text{vech}\left((\mathbf{C}_\sigma \otimes \mathbf{B}_0^T)^{-\frac{1}{2}} \xi_t \xi_t^T (\mathbf{C}_\sigma \otimes \mathbf{B}_0^T)^{-\frac{1}{2}}\right),$$

where $\text{vech}(\cdot)$ transfers a symmetric $nL \times nL$ matrix to an $nL \times (nL + 1)/2$ vector including the elements below and on the diagonal. When $\mathbf{X}_t$ is in the normal state, the mean vector and the covariance matrix of $T_{t,1}$ are $E(T_{t,1}) = \mathbf{0}$ and $\text{Cov}(T_{t,1}) = \mathbf{I}_{nL}$, and those of $T_{t,2}$ are $E(T_{t,2}) = \text{vech}(\mathbf{I}_{nL})$ and $\text{Cov}(T_{t,2}) = \mathbf{\Omega}$. Finally, we consider the simultaneous monitoring for both the mean change and the covariance change. The characteristic quantity T_t is a combination of $T_{t,1}$ and $T_{t,2}$, that is,

$$T_t = \begin{bmatrix} T_{t,1} \\ T_{t,2} \end{bmatrix}.$$

When $\mathbf{X}_t$ is in the normal state, T_t is normally distributed with the mean

$$E(T_t) = \begin{bmatrix} \mathbf{0} \\ \text{vech}(\mathbf{I}_{nL}) \end{bmatrix}$$

and the covariance matrix

$$\text{Cov}(T_t) = \begin{bmatrix} \mathbf{I}_{nL} & \mathbf{0}_{nL \times nL(nL+1)/2} \\ \mathbf{0}_{nL(nL+1)/2 \times nL} & \mathbf{\Omega} \end{bmatrix}.$$

We establish two CUSUM-based control charts (Garthoff and Otto, 2017) based on the characteristic quantity T_t to monitor the network system as follows:

3.3.1. The first type of T_t -based CUSUM (TCUSUM-1) control chart

At each time t , the TCUSUM-1 control chart is based on the square of the Mahalanobis distance of T_t from its in-control mean and covariance matrix:

$$V_t = (T_t - E(T_t))^T \text{Cov}(T_t)^{-1} (T_t - E(T_t)). \quad (14)$$

Then, the control statistic $TC1(t)$ is equal to

$$TC1(t) = \max\left\{0, TC1(t-1) + V_t - k_1 \left(nL + \frac{nL(nL+1)}{2}\right)\right\} \quad (15)$$

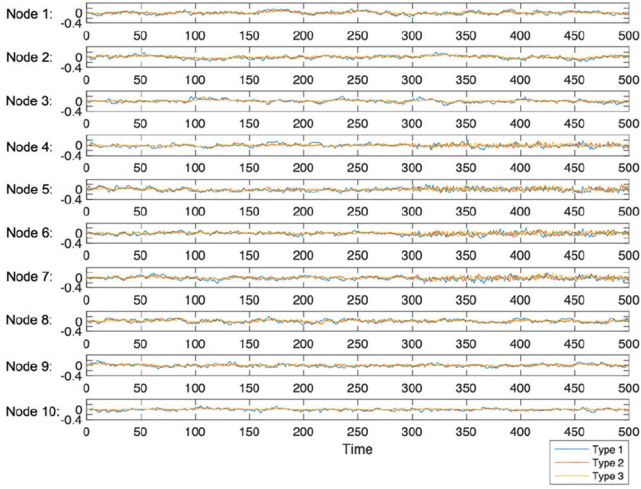


Figure 5. An example of a simulated network system.

with $TC1(0) = 0$, where k_1 is the reference parameter with $k_1 \geq 0$. An attack is detected whenever $TC1(t)$ exceeds a prespecified control limit h_1 , i.e., $TC1(t) > h_1$. h_1 is specified based on the requirement for satisfying a certain in-control average run length (ARL_0) through Monte Carlo simulations, where ARL_0 represents the expected number of runs until a false alarm occurs when an in-control network system is actually monitored.

3.3.2. The second type of T_t -based CUSUM (TCUSUM-2) control chart

At each time t , a TCUSUM-2 control chart is based on the cumulative sum in terms of T_t from time $t - n_t + 1$ to t as follows:

$$S_{t-n_t, t} = \sum_{t=t-n_t+1}^t T_t - E(T_t). \quad (16)$$

Then, the respective norm of $S_{t-n_t, t}$ is

$$\|S_{t-n_t, t}\|_{Cov(T_t)} = \left(S_{t-n_t, t}^T Cov(T_t)^{-1} S_{t-n_t, t} \right)^{\frac{1}{2}}. \quad (17)$$

The control statistic $TC2(t)$ is equal to

$$TC2(t) = \max \left\{ 0, \|S_{t-n_t, t}\|_{Cov(T_t)} - k_2 n_t \right\} \quad (18)$$

with $TC2(0) = 0$, where

$$n_t = \begin{cases} n_{t-1} + 1, & \text{if } TC2(t-1) > 0 \\ 0, & \text{if } TC2(t-1) = 0 \end{cases}$$

and k_2 is the reference parameter with $k_2 \geq 0$. An attack is detected whenever $TC2(t)$ exceeds a prespecified control limit h_2 . Similarly, h_2 is specified based on the requirement for satisfying a certain ARL_0 through Monte Carlo simulations.

4. Numerical experiments

We evaluate the performance of our proposed method through numerical experiments under different scenarios and compare it with various existing benchmark methods.

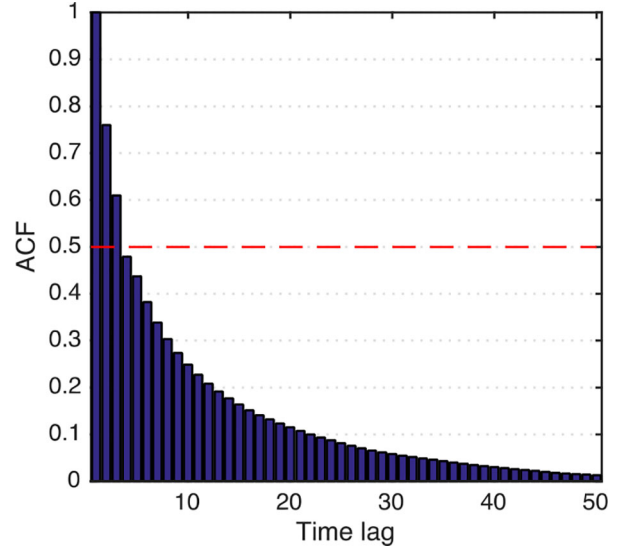


Figure 6. The ACFs under different time lags.

In Sections 4.1 and 4.2, we first introduce the benchmark methods and evaluation metrics of model performances. Then, we compare the results of the proposed method with the benchmark methods under four scenarios:

- Scenario 1: Network systems with sudden changes in mean.
- Scenario 2: Network systems with sudden changes in variation.
- Scenario 3: A cyclical network system.
- Scenario 4: A chaotic-nonlinear network system.

We present Scenario 1 in Section 4.3, and the results of Scenarios 2, 3 and 4 are provided in the supplemental online materials.

4.1. Benchmark methods

We consider various benchmark methods for network modeling and network monitoring, respectively.

For network modeling, we compare the proposed MSTA model with several benchmark models for spatio-temporal modeling as follows:

- Without consideration of multivariate spatio-temporal modeling (NONE).
- The Gaussian Process model that uses a simple covariance function for multivariate spatio-temporal modeling (Wang and Zhang, 2019).
- The VAR model that only considers multivariate temporal correlation (Di Giacinto, 2010).
- The STCAR model that considers univariate spatio-temporal correlation for each signal and omits the correlation among signals (Mariella and Tarantino, 2010).

For network monitoring, we compare the TCUSUM-1 and TCUSUM-2 in the proposed method with four types of change point detection benchmark methods as follows:

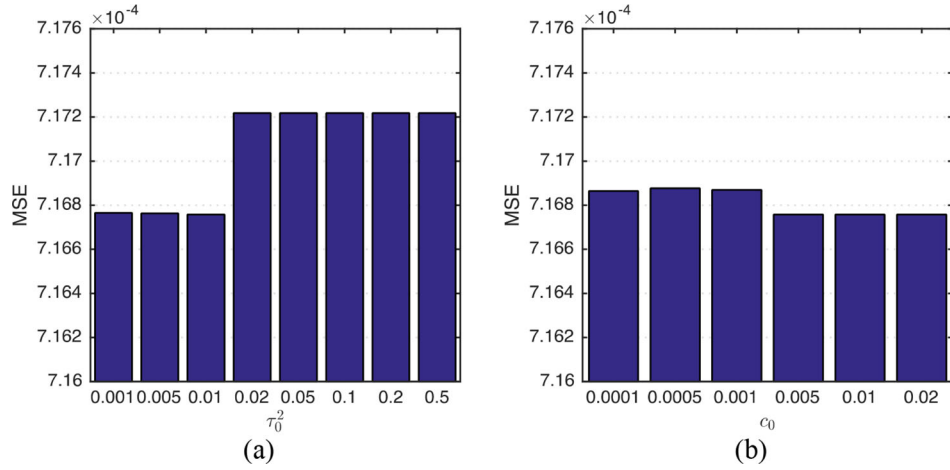


Figure 7. The MSE given various values of the hyperprior parameters: (a) τ_0^2 , (b) c_0 .

- (i) The control charts including Conventional CUSUM (CCUSUM) (Woodall and Ncube, 1985), EWMA (Lucas and Saccucci, 1990), multivariate Hotelling T^2 (Grimshaw *et al.*, 2013), TCUSUM-1-Mean, and TCUSUM-2-Mean. In particular, the frameworks of the TCUSUM-1-Mean and the TCUSUM-2-Mean charts are similar to TCUSUM-1 and TCUSUM-2 charts. The only difference is that the TCUSUM-1-Mean and the TCUSUM-2-Mean charts only consider characteristic quantity of the mean change.
- (ii) The supervised learning models including the NN (Li *et al.*, 2019), SVM (Liou *et al.*, 2018), DT (Moon *et al.*, 2017), and RF (Ambikavathi and Srivatsa, 2020).
- (iii) The unsupervised learning models including SOM (Altman, 1992) and KNN (Vesanto and Alhoniemi, 2000).
- (iv) The signal processing models including control charts of multivariate WT features (Wu and Wang, 2011).

For the benchmark methods with respect to network modeling, we first use the proposed MSTA model and benchmark methods to model the network and then use the TCUSUM-1 and TCUSUM-2 in the proposed method to monitor the network. For the benchmark methods with respect to network monitoring, we first use the proposed MSTA to model the network and then use TCUSUM-1 and TCUSUM-2 in the proposed method and the benchmark methods to monitor the network. Besides, all evaluation results of the proposed method and benchmark methods are computed based on 1000 simulation runs.

4.2. Evaluation metrics

We employ several metrics to evaluate the model performance. One metric is the out-of-control average run length ARL_1 , which denotes the number of time points until an alarm occurs when the monitored network system is under attack, to measure the timeliness of detecting an attack. A

small ARL_1 value indicates the attack is detected quickly in real time. The other metrics are widely used in the machine learning literature and measure the efficiency and stability of monitoring, including accuracy, precision, recall, and F1-score. Large values of Accuracy, Precision, Recall, and F1-Score indicate the attack is detected with high efficiency and stability. The detailed introduction of the metrics is provided in the supplemental online materials.

4.3. Network systems with sudden changes in mean

We simulate a network system that has $n = 10$ sensor nodes, and there are $L = 3$ types of signals collected at each sensor node. The network structure that reflects the interconnections of the sensor nodes is shown in Figure 4, and the network structure matrix γ is set as

$$\gamma = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}.$$

The simulated data $\mathbf{Y} = \{\mathbf{Y}_t\}_{t=1,2,\dots}$ of the network system are generated as follows: At the initial time point $t = 1$, $\mathbf{Y}_t = \mathbf{0}$. Then, at each time point ($t > 1$), $\mathbf{Y}_t$ is obtained from $\mathbf{Y}_{t-Q}, \dots, \mathbf{Y}_{t-1}$ following the equation:

$$\mathbf{Y}_t = \sum_{q=1}^Q (\mathbf{I}_L \otimes (\mathbf{I} - \boldsymbol{\beta}_0))^{-1} (\mathbf{A}_q \otimes (\mathbf{I} - \boldsymbol{\beta}_q)) \mathbf{Y}_{t-q} + \boldsymbol{\delta}_t,$$

where $\boldsymbol{\delta}_t$ is randomly generated from the multivariate normal distribution $N(\mathbf{0}, \mathbf{C}_\sigma \otimes (\mathbf{I} - \boldsymbol{\beta}_0)^{-1})$. Here, we set $Q = 1$,

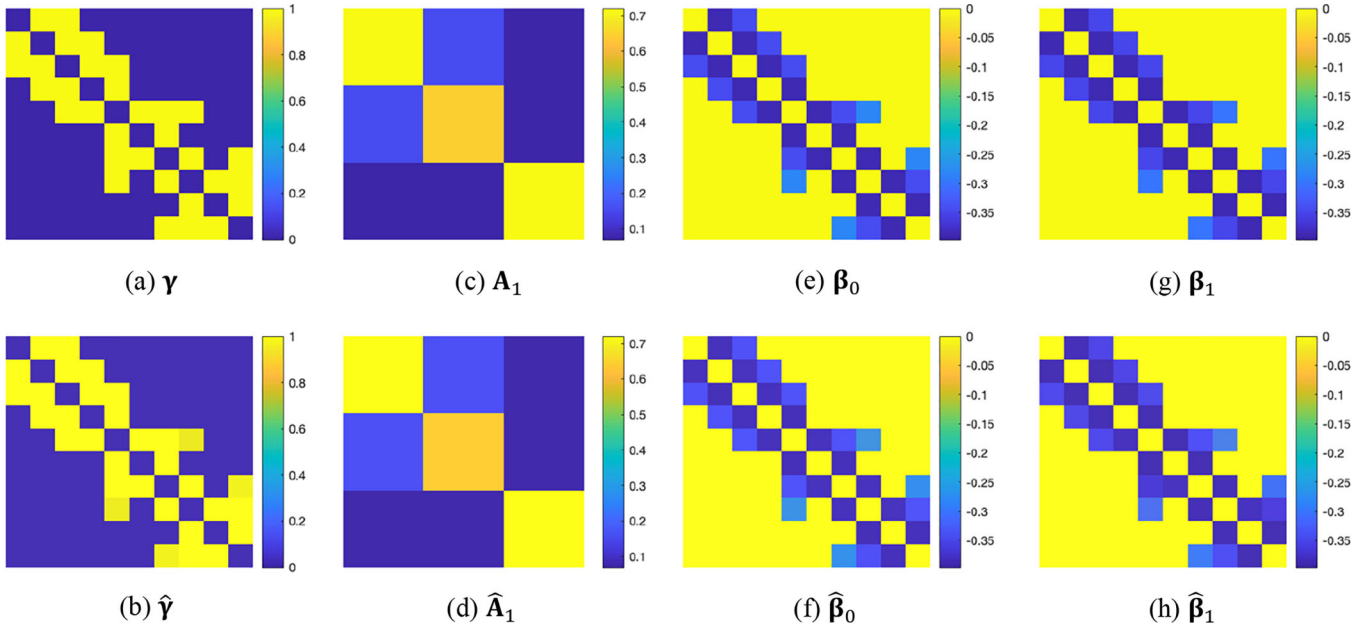


Figure 8. True values and estimated values of γ , A_1 , β_0 , and β_1 .

$$A_q = \begin{bmatrix} 0.72 & 0.16 & 0.08 \\ 0.16 & 0.64 & 0.08 \\ 0.08 & 0.08 & 0.72 \end{bmatrix},$$

$$C_\sigma = \begin{bmatrix} 0.0009 & 0 & 0 \\ 0 & 0.0004 & 0 \\ 0 & 0 & 0.0001 \end{bmatrix}, \beta_0 = [\beta_{ij}^0]_{i,j=1,\dots,n},$$

with

$$\beta_{ij}^0 = \begin{cases} 0.4e^{-\frac{\|s_j - s_i\|^2}{5^2}}, & \text{if } \gamma_{ij} = 1, \\ 0, & \text{if } \gamma_{ij} = 0 \end{cases}, \text{ and } \beta_q = [\beta_{ij}^q]_{i,j=1,\dots,n},$$

with

$$\beta_{ij}^q = \begin{cases} 0.4e^{-\frac{\|s_j - s_i\|^2}{5.5^2}}, & \text{if } \gamma_{ij} = 1, \\ 0, & \text{if } \gamma_{ij} = 0 \end{cases}.$$

Then, the observed data $\mathbf{X} = \{\mathbf{X}_t\}_{t=1,2,\dots}$ of the network system, which have two states including the normal state and the under-attack state, are generated based on the simulated data $\mathbf{Y} = \{\mathbf{Y}_t\}_{t=1,2,\dots}$. For the normal state:

$$x_t^l(\mathbf{s}) = y_t^l(\mathbf{s}) \text{ for all } \mathbf{s} \in \mathcal{S}, \quad (19)$$

where $x_t^l(\mathbf{s})$ and $y_t^l(\mathbf{s})$ denote the observed data and the simulated data of the l th signal at location $\mathbf{s}$ in $\mathbf{X}_t$ and $\mathbf{Y}_t$, respectively. For the under-attack state, we consider mean shifts as follows:

$$x_t^l(\mathbf{s}) = \begin{cases} y_t^l(\mathbf{s}), & \text{for } \mathbf{s} \notin \mathcal{S}_o, \\ \alpha_l + y_t^l(\mathbf{s}), & \text{for } \mathbf{s} \in \mathcal{S}_o, \end{cases} \quad (20)$$

where $\mathcal{S}_o$ denotes the outbreak cluster, i.e., the set of all nodes that have the change, and α_l denotes the mean shift of the l th signal. In Equation (20), $x_t^l(\mathbf{s}) = y_t^l(\mathbf{s})$ represents the l th signal at node $\mathbf{s}$ and time t of the network system is in the normal state, and $x_t^l(\mathbf{s}) = \alpha_l + y_t^l(\mathbf{s})$ represents the l th signal at node $\mathbf{s}$ and time t of the network system is under

attack. Figure 5 shows an example of a simulated network system that is in the normal state at the beginning and becomes under attack at time 300 on nodes 4, 5, 6 and 7 with $\alpha_l = 0.05$ ($l = 1, \dots, L$).

4.3.1. Multivariate spatiotemporal modeling by the proposed MSTa method

We generate a number of observed data $\mathbf{X} = \{\mathbf{X}_t\}_{t=1,2,\dots,M}$, with $M = 10,000$, of the network system in the normal state, and take the generated normal data to estimate parameters of the proposed MSTa method. Before estimating model parameters, we need to set the order Q and prefix the hyperprior-parameters of the proposed MSTa method, including q in the prior distribution of γ , τ_0^2 and τ_1^2 in the prior distribution of β_0 , and the predefined threshold c_0 in the initialization of γ . To begin with, we first obtain candidate values of the order Q by using Auto-Correlation Functions (ACFs) of the observed data under different time lags. As shown in Figure 6, we choose the candidate values of Q as one and two, whose ACFs are larger than 0.5 (William, 2019). For each candidate value of Q , we choose the hyperprior-parameters using the following procedure: Here, we take $Q = 1$ as an example. For γ , γ_{ij} has two candidate values one and zero, with $P(\gamma_{ij} = 1) = 1 - P(\gamma_{ij} = 0) = q$. In this article, we set $q = 0.5$. For β_0 , τ_0^2/τ_1^2 should be smaller than one. Without loss of generality, we set $\tau_1^2 = 1$. We should further adjust τ_0^2 and c_0 to find their suitable values. We first fix c_0 as a randomly selected number close to zero, i.e., $c_0 = 0.005$. Then, given various values of τ_0^2 , i.e., $\tau_0^2 = 0.001, 0.005, 0.01, 0.05, 0.1, 0.2$ and 0.5 , we estimate model parameters and predict $\mathbf{X}_t$ following Equation (8), and calculate the Mean Square Error (MSE) of the predicted $\hat{\mathbf{X}}_t$ values and the true $\mathbf{X}_t$ values. We choose τ_0^2 as the value with the smallest MSE. Figure 7(a) shows the calculated MSE given the various values of τ_0^2 , in which we

can see that the MSE has the smallest value when $\tau_0^2 = 0.01$. After τ_0^2 is chosen, we analyze the sensitivity of c_0 . Given various values of c_0 , i.e., $c_0 = 0.0001, 0.0005, 0.001, 0.005, 0.01$, and 0.02 , we estimate model parameters and predict $\mathbf{X}_t$ following Equation (8). Figure 7(b) shows the calculated MSE given the various values of c_0 , in which we can see that c_0 has little effect on the MSE. A possible reason is that c_0 is only used for the initialization of γ and the estimated result of γ (i.e., $\hat{\gamma}$) is obtained by iterative updates based on the observed data, and thus c_0 will not significantly influence $\hat{\gamma}$. Finally, we choose $c_0 = 0.005$. Similarly, for the order $Q = 2$, we prefix the hyperprior parameters using the aforementioned procedure and estimate model parameters. We calculate the MSE of the proposed model and obtain $\text{MSE} = 7.168 \times 10^{-4}$ for $Q = 1$ and $\text{MSE} = 7.185 \times 10^{-4}$ for $Q = 2$. Finally, we select the order $Q = 1$ for our proposed model, which has the minimal MSE value.

After prefixing the hyperprior parameters and setting the order, we estimate the network structure matrix γ , the temporal-associated matrix $\mathbf{A}_q$, with $q = 1, \dots, Q$, the spatial dependence matrices β_0 and β_q , with $q = 1, \dots, Q$, and the overall variance of the l th signal type σ_l^2 , with $l = 1, \dots, L$, using the generated normal data and repeat the corresponding procedures 100 times to demonstrate the robustness of the proposed parameter estimation algorithm. Figure 8 presents the true values of γ , $\mathbf{A}_1$, β_0 , and β_1 and the corresponding estimated values under 100 replications. Table 1 shows the true values and the estimated values of σ_l^2 with $l = 1, \dots, L$, where the estimated values include the mean values and the standard deviations under 100 replications. The estimated value of γ is demonstrated in the heat map in Figure 8(b), and is very close to the true value shown in Figure 8(a), which indicates the estimated γ learns

Table 1. True values and estimated values of σ_l^2 , with $l = 1, \dots, L$.

$\times 10^{-3}$	σ_1^2	σ_2^2	σ_3^2
True values	0.9	0.4	0.1
Estimated values	0.8998 (0.0016)	0.3999 (0.0005)	0.1001 (0.0002)

Note: The estimated values are the mean values under 100 replications, and the values in parentheses are the corresponding standard deviations.

the network structure very well. For $\mathbf{A}_1$, β_0 , β_1 , and σ_l^2 , with $l = 1, \dots, L$, it can be observed that the estimations for all four types of parameters are close to their true underlying values, which indicates the proposed MSTA method accurately characterizes the spatial dependence, the temporal association, and the overall variance of all signal types, respectively. All of the parameters have small standard deviations, which indicates the proposed parameter estimation algorithm is satisfactory. In addition, we show the estimated σ_l^2 with $l = 1, \dots, L$ at each iteration under one replication as an example in Figure 9 to demonstrate the convergence of the proposed parameter estimation algorithm. We can see that σ_l^2 converges quickly after iteration 4.

4.3.2. Evaluation of network modeling and monitoring

We first specify the control limits h_1 and h_2 of the proposed TCUSUM-1 and TCUSUM-2 control charts by setting $\text{ARL}_0 = 1000$ for network monitoring. The control limits h_1 and h_2 are determined by using the in-control data of the network system generated from Equation (19). To better evaluate our proposed model performance for monitoring of a network system, we simulate a number of under-attack scenarios of the network system as follows: we consider three outbreak cluster states, i.e., small, medium, large outbreak cluster, corresponding to $\mathcal{S}_O = \{5\}$, $\mathcal{S}_O = \{5, 6\}$, and $\mathcal{S}_O = \{4, 5, 6, 7\}$. In addition, we set three mean shift magnitudes from small to large shifts, corresponding to $\alpha_l = 0.03, 0.05$, and 0.1 (with $l = 1, \dots, L$). We generate observed data of the network system under each under-attack scenario using Equation (20).

Then, we use the proposed method and the benchmark methods to model and monitor the network system and repeat the corresponding procedures 1000 times. Tables 2 and 3 present average values of the metrics of the proposed method and the benchmark methods based on 1000 replications for network modeling and monitoring, respectively. For network modeling in Table 2, the proposed MSTA + TCUSUM-1 and MSTA + TCUSUM-2 perform better at all under-attack scenarios by effectively characterizing the multivariate spatio-

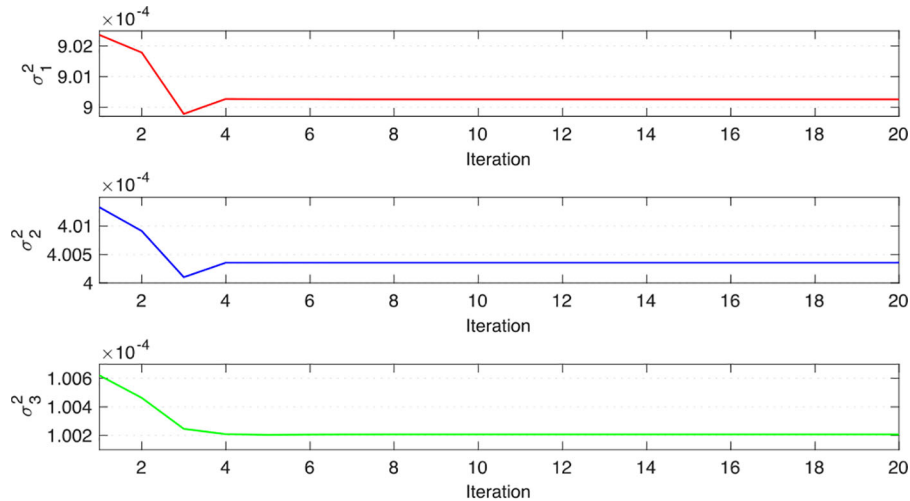


Figure 9. The convergence of σ_l^2 , with $l = 1, \dots, L$.

Table 2. Modeling performances of the proposed and benchmark methods for systems with sudden changes in mean.

Control Charts	NONE + TC USUM-1	NONE + TC USUM-2	GP + TC USUM-1	GP + TC USUM-2	VAR + TC USUM-1	VAR + TC USUM-2	STCAR + TC USUM-1	STCAR + TC USUM-2	MSTA + TC USUM-1	MSTA + TC USUM-2
<i>Metrics</i>										
$\alpha_I = 0.03, S_0 = \{5\}$										
ARL_1	33.60(31.06)	50.30(47.57)	154.88(75.8)	156.51(40.83)	16.68(15.6)	52.24(13.16)	9.79(8.99)	9.10(6.20)	4.72(3.75)	7.60(7.37)
Accuracy	0.81(0.10)	0.61(0.01)	0.61(0.12)	0.69(0.08)	0.69(0.03)	0.90(0.03)	0.71(0.03)	0.95(0.08)	0.99(0.01)	0.96(0.03)
Precision	0.98(0.07)	0.93(0.12)	0.80(0.35)	1.00(<0.01)	0.99(0.02)	1.00(<0.01)	0.99(0.02)	0.95(0.12)	1.00(0.01)	0.99(0.01)
Recall	0.54(0.26)	0.04(0.02)	0.16(0.30)	0.23(0.20)	0.23(0.09)	0.74(0.07)	0.29(0.08)	0.96(0.03)	0.98(0.02)	0.96(0.03)
F1-Score	0.66(0.24)	0.07(0.04)	0.17(0.28)	0.33(0.26)	0.37(0.11)	0.85(0.04)	0.44(0.10)	0.95(0.07)	0.99(0.01)	0.96(0.03)
$\alpha_I = 0.05, S_0 = \{5\}$										
ARL_1	5.71(4.39)	8.17(8.06)	118.55(73.66)	112.99(33.3)	4.54(3.52)	18.20(7.60)	3.34(2.56)	3.80(2.52)	2.08(1.25)	2.61(2.00)
Accuracy	0.99(0.01)	0.70(0.03)	0.70(0.14)	0.78(0.07)	0.99(0.01)	0.97(0.02)	0.99(0.01)	0.96(0.08)	1.00(<0.01)	0.99(0.01)
Precision	0.99(0.02)	0.99(0.02)	0.87(0.26)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.95(0.12)	1.00(0.01)	1.00(<0.01)
Recall	0.97(0.02)	0.26(0.07)	0.38(0.35)	0.44(0.17)	0.97(0.04)	0.91(0.04)	0.97(0.03)	0.99(0.01)	0.99(0.01)	0.98(0.02)
F1-Score	0.98(0.02)	0.41(0.09)	0.42(0.34)	0.59(0.17)	0.98(0.02)	0.95(0.02)	0.98(0.02)	0.96(0.07)	1.00(<0.01)	0.99(0.01)
$\alpha_I = 0.1, S_0 = \{5\}$										
ARL_1	1.84(1.14)	2.05(1.64)	28.92(17.7)	32.84(12.39)	1.69(1.05)	4.42(2.51)	1.40(0.75)	1.73(0.97)	1.26(0.53)	1.32(0.69)
Accuracy	0.99(0.01)	1.00(0.01)	0.89(0.10)	0.94(0.02)	1.00(<0.01)	0.99(0.01)	1.00(<0.01)	0.97(0.07)	1.00(<0.01)	1.00(<0.01)
Precision	0.99(0.02)	1.00(0.01)	0.92(0.15)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.96(0.11)	1.00(0.01)	1.00(<0.01)
Recall	1.00(0.01)	0.99(0.01)	0.86(0.09)	0.84(0.06)	1.00(0.01)	0.98(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)
F1-Score	0.99(0.01)	0.99(0.01)	0.88(0.08)	0.91(0.04)	1.00(<0.01)	0.99(0.01)	1.00(<0.01)	0.97(0.07)	1.00(<0.01)	1.00(<0.01)
$\alpha_I = 0.03, S_0 = \{5, 6\}$										
ARL_1	5.97(4.75)	8.52(7.89)	123.94(72.24)	113.00(34.67)	7.01(6.55)	25.93(9.11)	3.19(2.46)	3.83(2.44)	2.11(1.20)	2.64(1.93)
Accuracy	0.98(0.01)	0.70(0.03)	0.69(0.13)	0.78(0.07)	0.94(0.05)	0.95(0.02)	0.99(0.01)	0.97(0.07)	1.00(<0.01)	0.99(0.01)
Precision	0.99(0.02)	0.99(0.02)	0.86(0.27)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.95(0.11)	1.00(0.01)	1.00(<0.01)
Recall	0.97(0.03)	0.25(0.07)	0.36(0.34)	0.44(0.17)	0.85(0.12)	0.88(0.05)	0.97(0.03)	0.99(0.01)	0.99(0.01)	0.98(0.03)
F1-Score	0.98(0.02)	0.40(0.09)	0.40(0.33)	0.59(0.18)	0.91(0.08)	0.93(0.03)	0.98(0.02)	0.97(0.06)	1.00(<0.01)	0.99(0.01)
$\alpha_I = 0.05, S_0 = \{5, 6\}$										
ARL_1	2.27(1.43)	2.64(2.13)	41.24(26.09)	49.39(16.84)	2.53(1.84)	8.39(4.34)	1.61(0.99)	1.97(1.15)	1.37(0.64)	1.49(0.90)
Accuracy	0.99(0.01)	0.98(0.02)	0.87(0.10)	0.90(0.03)	1.00(<0.01)	0.99(0.01)	1.00(<0.01)	0.97(0.08)	1.00(<0.01)	1.00(<0.01)
Precision	0.99(0.02)	1.00(0.01)	0.92(0.16)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.95(0.11)	1.00(0.01)	1.00(<0.01)
Recall	0.99(0.01)	0.96(0.05)	0.80(0.13)	0.76(0.08)	0.99(0.01)	0.96(0.02)	1.00(0.01)	1.00(0.01)	1.00(<0.01)	1.00(<0.01)
F1-Score	0.99(0.01)	0.98(0.03)	0.84(0.10)	0.86(0.05)	0.99(0.01)	0.98(0.01)	1.00(<0.01)	0.97(0.07)	1.00(<0.01)	1.00(<0.01)
$\alpha_I = 0.1, S_0 = \{5, 6\}$										
ARL_1	1.28(0.56)	1.30(0.65)	10.23(6.55)	8.95(4.81)	1.29(0.61)	2.72(1.5)	1.10(0.32)	1.18(0.42)	1.06(0.23)	1.07(0.26)
Accuracy	1.00(0.01)	1.00(<0.01)	0.93(0.11)	0.98(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.97(0.08)	1.00(<0.01)	1.00(<0.01)
Precision	0.99(0.02)	1.00(0.01)	0.91(0.15)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.95(0.11)	1.00(0.01)	1.00(<0.01)
Recall	1.00(<0.01)	1.00(<0.01)	0.95(0.03)	0.96(0.02)	1.00(<0.01)	0.99(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)
F1-Score	0.99(<0.01)	1.00(<0.01)	0.92(0.09)	0.98(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.97(0.07)	1.00(<0.01)	1.00(<0.01)
$\alpha_I = 0.03, S_0 = \{4, 5, 6, 7\}$										
ARL_1	2.38(1.54)	2.93(2.59)	46.71(30.51)	54.56(18.56)	3.34(2.60)	12.12(5.76)	1.76(1.15)	2.13(1.27)	1.38(0.64)	1.49(0.82)
Accuracy	0.99(0.01)	0.96(0.03)	0.85(0.10)	0.89(0.04)	0.99(0.01)	0.98(0.01)	1.00(<0.01)	0.97(0.07)	1.00(<0.01)	1.00(<0.01)
Precision	0.99(0.02)	1.00(0.01)	0.91(0.16)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.96(0.10)	1.00(0.01)	1.00(<0.01)
Recall	0.99(0.01)	0.91(0.08)	0.77(0.15)	0.73(0.09)	0.98(0.02)	0.94(0.03)	1.00(0.01)	0.99(0.01)	1.00(<0.01)	1.00(<0.01)
F1-Score	0.99(0.01)	0.95(0.05)	0.81(0.10)	0.84(0.06)	0.99(0.01)	0.97(0.02)	1.00(<0.01)	0.97(0.06)	1.00(<0.01)	1.00(<0.01)
$\alpha_I = 0.05, S_0 = \{4, 5, 6, 7\}$										
ARL_1	1.41(0.71)	1.46(0.89)	15.3(10.30)	16.03(7.82)	1.61(0.92)	4.44(2.43)	1.19(0.48)	1.35(0.63)	1.10(0.32)	1.13(0.41)
Accuracy	1.00(0.01)	1.00(<0.01)	0.92(0.10)	0.97(0.02)	1.00(<0.01)	0.99(<0.01)	1.00(<0.01)	0.97(0.08)	1.00(<0.01)	1.00(<0.01)
Precision	0.99(0.02)	1.00(0.01)	0.92(0.15)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.95(0.11)	1.00(0.01)	1.00(<0.01)
Recall	1.00(<0.01)	1.00(0.01)	0.93(0.05)	0.93(0.04)	1.00(<0.01)	0.98(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)
F1-Score	0.99(0.01)	1.00(<0.01)	0.91(0.09)	0.96(0.02)	1.00(<0.01)	0.99(0.01)	1.00(<0.01)	0.97(0.07)	1.00(<0.01)	1.00(<0.01)
$\alpha_I = 0.1, S_0 = \{4, 5, 6, 7\}$										
ARL_1	1.07(0.27)	1.08(0.29)	4.55(3.03)	3.47(2.12)	1.11(0.34)	1.78(0.90)	1.03(0.18)	1.06(0.25)	1.02(0.14)	1.02(0.16)
Accuracy	1.00(0.01)	1.00(<0.01)	0.94(0.11)	0.99(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.97(0.08)	1.00(<0.01)	1.00(<0.01)
Precision	0.99(0.02)	1.00(0.01)	0.92(0.15)	1.00(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.95(0.11)	1.00(0.01)	1.00(<0.01)
Recall	1.00(<0.01)	1.00(<0.01)	0.98(0.02)	0.99(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)
F1-score	1.00(0.01)	1.00(<0.01)	0.94(0.09)	0.99(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.97(0.07)	1.00(<0.01)	1.00(<0.01)

Note: Values in parentheses denote the standard deviations of the metrics.

temporal correlation of the network system. Especially for detecting small outbreak cluster (i.e., $S_0 = \{5\}$) and small shift (i.e., $\alpha_I = 0.03$), the superior behaviors of the proposed MSTA + TCUSUM-1 and MSTA + TCUSUM-2 are more evident with much smaller ARL_1 values and larger values of Accuracy, Precision, Recall, and F1-score than NONE + TCUSUM-1 and NONE + TCUSUM-2, GP + TCUSUM-1 and GP + TCUSUM-2 with a simple multivariate spatiotemporal covariance function, VAR + TCUSUM-1 and VAR + TCUSUM-2 that only consider multivariate temporal correlation, as well as STCAR + TCUSUM-1 and STCAR + TCUSUM-2 that omit the correlation among signals. For network monitoring in Table 3, compared with the benchmark

methods, TCUSUM-1 and TCUSUM-2 in the proposed method have comparable ARL_1 values and higher values of Accuracy, Precision, Recall, and F1-score, which indicates the timeliness and stability of the proposed method. As the outbreak cluster and mean shift become large, both the proposed method and benchmark methods perform better for network monitoring.

5. Real case study

In this section, we use a real wireless IoT testbed to evaluate our proposed method. The IoT testbed is a network system built by six wireless CPS devices that include Raspberry Pis

Table 3. Monitoring performances of the proposed and benchmark methods for systems with sudden changes in mean.

Control Charts		Metrics													
Metrics		T^2	CCUSUM	EWMA	SVM	DT	RF	NN	KNN	SOM	WT	TCUSUM-1-Mean	TCUSUM-1-Mean	TCUSUM-2-Mean	
ARL ₁ Accuracy Precision Recall F1-Score		5.50(4.55)	4.35(3.38)	11.5(10.71)	200.8(6.29)	2.98(2.39)	3.72(3.23)	$\alpha_i = 0.03, S_0 = \{5\}$		1.00(<0.01)	6.44(5.37)	4.75(3.50)	9.35(9.07)	4.72(3.75)	7.60(7.37)
		0.69(0.01)	0.73(0.02)	0.64(0.01)	0.60(<0.01)	0.73(0.02)	0.72(0.01)	0.60(<0.01)	0.59(<0.01)	0.40(<0.01)	0.68(0.01)	0.99(0.01)	0.64(0.01)	0.99(0.01)	0.96(0.03)
		0.99(0.01)	1.00(0.01)	0.98(0.03)	1.00(<0.01)	0.95(0.03)	0.98(0.02)	0.98(0.07)	0.37(0.17)	0.40(<0.01)	0.99(0.02)	1.00(0.01)	0.99(0.02)	1.00(0.01)	0.99(0.01)
		0.23(0.03)	0.34(0.05)	0.10(0.02)	0.00(0 < 0.01)	0.35(0.04)	0.30(0.04)	0.01(0.01)	0.01(0.01)	1.00(<0.01)	0.2(0.03)	0.98(0.02)	0.11(0.02)	0.98(0.02)	0.96(0.03)
		0.37(0.04)	0.50(0.05)	0.17(0.03)	0.01(<0.01)	0.51(0.04)	0.46(0.04)	0.03(0.01)	0.02(0.01)	0.57(<0.01)	0.33(0.05)	0.99(0.01)	0.20(0.03)	0.99(0.01)	0.96(0.03)
ARL ₁ Accuracy Precision Recall F1-Score		2.15(1.45)	1.97(1.28)	2.90(2.17)	103.88(70.96)	1.82(1.32)	1.98(1.44)	$\alpha_i = 0.05, S_0 = \{5\}$		1.00(<0.01)	2.65(1.83)	2.15(1.21)	3.37(2.65)	2.08(1.25)	2.61(2.00)
		0.82(0.02)	0.90(0.02)	0.74(0.01)	0.6(<0.01)	0.82(0.02)	0.80(0.02)	0.62(0.01)	0.60(0.01)	0.40(<0.01)	0.82(0.02)	1.00(<0.01)	0.71(0.01)	1.00(<0.01)	0.99(0.01)
		1.00(0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.97(0.02)	0.99(0.01)	0.99(0.03)	0.45(0.17)	0.40(<0.01)	1.00(0.01)	1.00(0.01)	0.99(0.01)	1.00(0.01)	1.00(<0.01)
		0.55(0.04)	0.75(0.04)	0.34(0.03)	0.01(0.01)	0.57(0.04)	0.51(0.04)	0.05(0.02)	0.02(0.01)	1.00(<0.01)	0.54(0.05)	0.99(0.01)	0.29(0.03)	0.99(0.01)	0.98(0.02)
		0.71(0.03)	0.86(0.03)	0.51(0.04)	0.03(0.01)	0.71(0.03)	0.67(0.03)	0.10(0.03)	0.03(0.02)	0.57(<0.01)	0.70(0.04)	1.00(0.01)	0.45(0.03)	1.00(<0.01)	0.99(0.01)
ARL ₁ Accuracy Precision Recall F1-Score		1.24(0.52)	1.22(0.48)	1.35(0.70)	3.17(2.76)	1.29(0.63)	1.36(0.72)	$\alpha_i = 0.1, S_0 = \{5\}$		1.00(<0.01)	1.31(0.59)	1.29(0.55)	1.60(1.04)	1.26(0.53)	1.32(0.69)
		0.95(0.01)	1.00(<0.01)	0.91(0.01)	0.74(0.02)	0.9(0.01)	0.89(0.01)	0.72(0.01)	0.63(0.01)	0.40(<0.01)	0.98(0.01)	1.00(<0.01)	0.82(0.01)	1.00(<0.01)	1.00(<0.01)
		1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(0.01)	0.97(0.01)	0.99(0.01)	1.00(<0.01)	0.82(0.08)	0.40(<0.01)	1.00(0.01)	1.00(0.01)	1.00(0.01)	1.00(0.01)	1.00(<0.01)
		0.88(0.02)	0.99(0.01)	0.77(0.03)	0.35(0.04)	0.78(0.03)	0.73(0.03)	0.30(0.03)	0.10(0.02)	1.00(<0.01)	0.95(0.02)	1.00(<0.01)	0.55(0.03)	1.00(<0.01)	1.00(<0.01)
		0.94(0.01)	1.00(<0.01)	0.87(0.02)	0.52(0.04)	0.86(0.02)	0.84(0.02)	0.45(0.04)	0.18(0.04)	0.57(<0.01)	0.97(0.01)	1.00(<0.01)	0.71(0.02)	1.00(<0.01)	1.00(<0.01)
ARL ₁ Accuracy Precision Recall F1-Score		2.21(1.43)	2.40(1.60)	3.02(2.32)	86.77(66.27)	2.13(1.60)	2.15(1.72)	$\alpha_i = 0.03, S_0 = \{5, 6\}$		1.00(<0.01)	2.01(1.26)	2.19(1.19)	3.31(2.56)	2.11(1.20)	2.64(1.93)
		0.82(0.01)	0.85(0.02)	0.74(0.01)	0.60(<0.01)	0.80(0.02)	0.79(0.02)	0.63(0.01)	0.60(0.01)	0.40(<0.01)	0.89(0.02)	1.00(<0.01)	0.71(0.01)	1.00(<0.01)	0.99(0.01)
		1.00(0.01)	1.00(<0.01)	1.00(0.01)	1.00(<0.01)	0.96(0.02)	0.99(0.01)	0.99(0.02)	0.45(0.17)	0.40(<0.01)	1.00(0.01)	1.00(0.01)	0.99(0.01)	1.00(0.01)	1.00(<0.01)
		0.55(0.04)	0.62(<0.01)	0.35(0.04)	0.02(0.01)	0.52(0.04)	0.49(0.04)	0.09(0.02)	0.02(0.01)	1.00(<0.01)	0.72(0.04)	0.99(0.01)	0.29(0.03)	0.99(0.01)	0.98(0.03)
		0.71(0.03)	0.76(0.04)	0.51(0.04)	0.03(0.01)	0.67(0.03)	0.66(0.03)	0.16(0.03)	0.03(0.02)	0.57(<0.01)	0.84(0.03)	1.00(0.01)	0.45(0.03)	1.00(<0.01)	0.99(0.01)
ARL ₁ Accuracy Precision Recall F1-Score		1.38(0.71)	1.49(0.82)	1.55(0.94)	5.00(4.38)	1.50(0.92)	1.56(0.98)	$\alpha_i = 0.05, S_0 = \{5, 6\}$		1.00(<0.01)	1.29(0.61)	1.42(0.66)	1.86(1.31)	1.37(0.64)	1.49(0.90)
		0.93(0.01)	0.97(<0.01)	0.87(0.01)	0.69(0.01)	0.87(0.01)	0.87(0.01)	0.71(0.01)	0.62(0.01)	0.40(<0.01)	0.98(0.01)	1.00(<0.01)	0.79(0.01)	1.00(<0.01)	1.00(<0.01)
		1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(0.01)	0.97(0.01)	0.99(0.01)	1.00(0.01)	0.76(0.10)	0.40(<0.01)	1.00(0.01)	1.00(0.01)	1.00(0.01)	1.00(0.01)	1.00(<0.01)
		0.82(0.03)	0.94(0.02)	0.67(0.03)	0.23(0.03)	0.69(0.03)	0.67(0.04)	0.28(0.03)	0.07(0.02)	1.00(<0.01)	0.96(0.02)	1.00(<0.01)	0.49(0.03)	1.00(<0.01)	1.00(<0.01)
		0.90(0.02)	0.97(0.01)	0.80(0.02)	0.37(0.04)	0.81(0.02)	0.80(0.03)	0.43(0.04)	0.13(0.03)	0.57(<0.01)	0.98(0.01)	1.00(<0.01)	0.65(0.03)	1.00(<0.01)	1.00(<0.01)
ARL ₁ Accuracy Precision Recall F1-Score		1.08(0.28)	1.11(0.34)	1.12(0.35)	1.35(0.71)	1.19(0.47)	1.23(0.53)	$\alpha_i = 0.1, S_0 = \{5, 6\}$		1.00(<0.01)	1.06(0.24)	1.06(0.23)	1.14(0.46)	1.06(0.23)	1.07(0.26)
		0.99(0.01)	1.00(<0.01)	0.97(0.01)	0.91(0.01)	0.93(0.01)	0.93(0.01)	0.84(0.02)	0.73(0.02)	0.40(<0.01)	1.00(0.01)	1.00(<0.01)	0.89(0.01)	1.00(<0.01)	1.00(<0.01)
		1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.98(0.01)	0.99(0.01)	1.00(<0.01)	0.94(0.03)	0.40(<0.01)	1.00(0.01)	1.00(0.01)	1.00(<0.01)	1.00(0.0001)	1.00(<0.01)
		0.97(0.01)	1.00(<0.01)	0.93(0.02)	0.77(0.03)	0.85(0.03)	0.83(0.03)	0.62(0.04)	0.36(0.04)	1.00(<0.01)	1.00(0.01)	1.00(<0.01)	0.72(0.02)	1.00(<0.01)	1.00(<0.01)
		0.90(0.01)	1.00(<0.01)	0.96(0.01)	0.87(0.02)	0.91(0.02)	0.90(0.02)	0.76(0.03)	0.52(0.04)	0.57(<0.01)	1.00(0.01)	1.00(<0.01)	0.83(0.02)	1.00(<0.01)	1.00(<0.01)
ARL ₁ Accuracy Precision Recall F1-Score		1.38(0.68)	1.71(0.99)	1.57(0.93)	4.89(4.57)	1.51(0.93)	1.51(0.96)	$\alpha_i = 0.03, S_0 = \{4, 5, 6, 7\}$		1.00(<0.01)	1.44(0.74)	1.43(0.67)	1.91(1.36)	1.38(0.64)	1.49(0.82)
		0.92(0.01)	0.93(0.01)	0.86(0.01)	0.69(0.01)	0.86(0.01)	0.86(0.01)	0.72(0.01)	0.62(0.01)	0.40(<0.01)	0.97(0.01)	1.00(<0.01)	0.79(0.01)	1.00(<0.01)	1.00(<0.01)
		1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(0.01)	0.97(0.01)	0.99(0.01)	1.00(0.01)	0.76(0.10)	0.40(<0.01)	1.00(0.01)	1.00(0.01)	1.00(0.01)	1.00(<0.01)	1.00(<0.01)
		0.80(0.03)	0.82(0.04)	0.64(0.03)	0.24(0.03)	0.67(0.04)	0.67(0.04)	0.31(0.04)	0.07(0.02)	1.00(<0.01)	0.92(0.02)	1.00(<0.01)	0.47(0.03)	1.00(<0.01)	1.00(<0.01)
		0.89(0.02)	0.90(0.02)	0.78(0.03)	0.38(0.04)	0.79(0.03)	0.80(0.03)	0.48(0.04)	0.13(0.03)	0.57(<0.01)	0.96(0.01)	1.00(<0.01)	0.64(0.03)	1.00(<0.01)	1.00(<0.01)
ARL ₁ Accuracy Precision Recall F1-Score		1.10(0.34)	1.21(0.47)	1.14(0.40)	1.56(0.97)	1.22(0.53)	1.27(0.57)	$\alpha_i = 0.05, S_0 = \{4, 5, 6, 7\}$		1.00(0.03)	1.12(0.35)	1.12(0.35)	1.28(0.67)	1.10(0.32)	1.13(0.41)
		0.98(0.01)	0.99(<0.01)	0.95(0.01)	0.86(0.02)	0.91(0.01)	0.92(0.01)	0.83(0.02)	0.69(0.01)	0.40(<0.01)	1.00(0.01)	1.00(<0.01)	0.86(0.01)	1.00(<0.01)	1.00(<0.01)
		1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.97(0.01)	0.99(0.01)	1.00(<0.01)	0.91(0.04)	0.40(<0.01)	1.00(0.01)	1.00(0.01)	1.00(<0.01)	1.00(0.01)	1.00(<0.01)
		0.94(0.02)	0.99(0.01)	0.88(0.02)	0.66(0.04)	0.80(0.03)	0.80(0.03)	0.57(0.04)	0.25(0.03)	1.00(<0.01)	1.00(0.01)	1.00(<0.01)	0.64(0.03)	1.00(<0.01)	1.00(<0.01)
		0.97(0.01)	0.99(<0.01)	0.93(0.01)	0.79(0.03)	0.88(0.02)	0.89(0.02)	0.72(0.03)	0.39(0.04)	0.57(<0.01)	1.00(0.01)	1.00(<0.01)	0.78(0.02)	1.00(<0.01)	1.00(<0.01)
ARL ₁ Accuracy Precision Recall F1-Score		1.02(0.14)	1.04(0.20)	1.02(0.16)	1.06(0.27)	1.14(0.39)	1.11(0.34)	$\alpha_i = 0.1, S_0 = \{4, 5, 6, 7\}$		1.11(0.38)	1.02(0.14)	1.02(0.14)	1.05(0.26)	1.02(0.14)	1.02(0.16)
		1.00(<0.01)	1.00(<0.01)	0.99(<0.01)	0.97(0.01)	0.95(0.01)	0.96(0.01)	0.93(0.01)	0.86(0.01)	0.37(0.01)	1.00(0.01)	1.00(<0.01)	0.92(0.01)	1.00(<0.01)	1.00(<0.01)
		1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	1.00(<0.01)	0.98(0.01)	0.99(<0.01)	1.00(<0.01)	0.96(0.01)	0.38(0.01)	1.00(0.01)	1.00(0.01)	1.00(<0.01)	1.00(0.01)	1.00(<0.01)
		0.99(0.01)	1.00(<0.01)	0.98(0.01)	0.94(0.02)	0.90(0.02)	0.90(0.02)	0.83(0.03)	0.67(0.03)	0.91(0.02)	1.00(0.01)	1.00(<0.01)	0.81(0.02)	1.00(<0.01)	1.00(<0.01)
		0.99(<0.01)	1.00(<0.01)	0.99(0.01)	0.97(0.01)	0.94(0.01)	0.95(0.01)	0.90(0.02)	0.79(0.02)	0.54(0.01)	1.00(0.01)	1.00(<0.01)	0.90(0.01)	1.00(<0.01)	1.00(<0.01)

Note: Values in parentheses denote the standard deviations of the metrics.

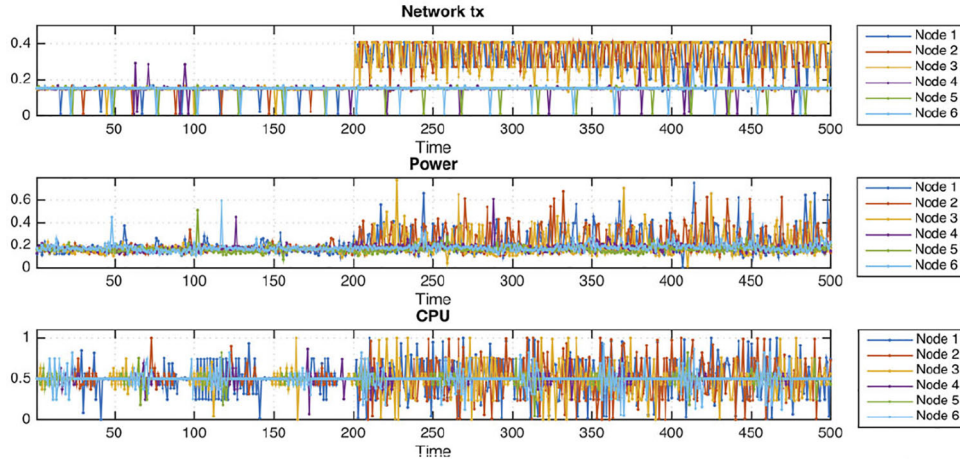


Figure 10. An example of the three types of the cyber and energy consumption data in the IoT network system. (Note: the data have been de-trended and standardized by 0-1 normalization.).

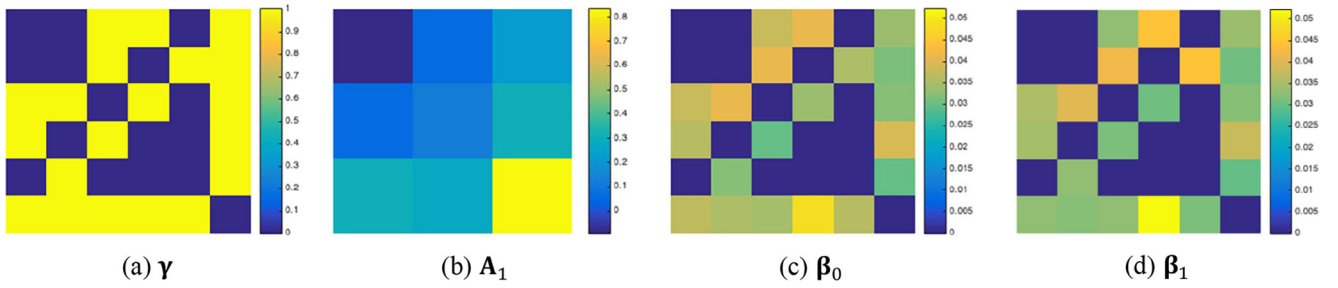


Figure 11. Estimated values of γ , A_1 , β_0 , and β_1 of the IoT network system.

as computation and communication units and geophones as sensing units. We run a distributed seismic ambient noise tomography imaging program in the testbed (Valero *et al.*, 2019). The comprehensive distributed program performs streaming data processing, RAM I/O (input/output) operations, Disk I/O operations, and communications (intermediate data and result transmissions). As shown in Figure 1, spatial information can be reflected based on the node relations of the CPS units. We collect and store cyber and energy consumption data generated by the nodes of the Raspberry Pis using widely available Application Programming Interfaces (APIs), such as collectd. At each CPS node, three types of cyber and energy consumption data are collected, including network tx, power, and CPU. Network tx indicates the network transmission data, whose pattern can help identify a cyber-attack over a network. Power is the energy consumption of the CPS units, which can reflect the hardware operating condition associated with cyber and physical attacks. CPU denotes the computation cost of the CPS units, which is highly related to both normal tasks and abnormal system burdens.

The IoT network system is in the normal state in the beginning, and becomes under attack at some time point and remains in the under-attack state. Note that there are six CPS nodes in the network system, and not all of them are under attack. Figure 10 shows an example of the three types of cyber and energy consumption data when the IoT system is in the normal state in the beginning and becomes under attack at one, two, and three nodes from the time

Table 4. Estimated values of σ_l^2 , with $l = 1, \dots, L$, of the IoT network system.

σ_l^2	σ_1^2	σ_2^2	σ_3^2
Estimated values	0.0024	0.0011	0.0014

point 200. We can see that the attack affects the patterns of the cyber and energy consumption data.

We first estimate the parameters of the proposed MSTA method using the observed data of the IoT network system in the normal state with $M = 6000$ time points. We select the order and the hyperprior-parameters using the same procedure in Section 4 with $Q = 1$, $q = 0.5$, $\tau_1^2 = 0.01$, $\tau_0^2 = 0.001$, and $c_0 = 0.01$. Figure 11 and Table 4 show the estimated values of γ , A_1 , β_0 , β_1 , and σ_l^2 , with $l = 1, \dots, L$. The estimated value of γ is demonstrated in the heat map in Figure 11(a), which represents the network structure of the IoT network system. The learnt network structure provides important engineering information of the IoT network system and can be further used in a wide range of engineering applications, e.g., optimum structural design, resource management, and task arrangement. The estimated A_1 , β_0 , and β_1 in Figure 11 and σ_l^2 , with $l = 1, \dots, L$, in Table 4 show the spatial dependence, the temporal association, and the overall variance of the IoT network system, respectively.

Before evaluating the monitoring performance, we determine the control limits h_1 and h_2 of the proposed TCUSUM-1 and TCUSUM-2 control charts based on the

Table 5. Modeling performances of the proposed and benchmark methods in the real case.

Metrics \ Control Charts	NONE + TC		GP + TC		VAR + TC		STCAR + TC		MSTA + TC	
	USUM-1	USUM-2	USUM-1	USUM-2	USUM-1	USUM-2	USUM-1	USUM-2	USUM-1	USUM-2
ARL_1	289.70	216.80	20.78	19.34	4.65	4.88	2.37	2.89	1.68	1.98
Accuracy	0.4226	0.5684	0.9824	0.9845	0.9899	0.9887	0.9974	0.9969	0.9983	0.9975
Precision	1.0000	1.0000	0.9945	0.9950	0.9977	0.9969	0.9995	0.9995	0.9995	0.9995
Recall	0.0377	0.2807	0.9899	0.9901	0.9945	0.9955	0.9971	0.9960	0.9977	0.9964
F1-Score	0.0725	0.4380	0.9922	0.9925	0.9961	0.9962	0.9983	0.9977	0.9986	0.9979

Table 6. Monitoring performances of the proposed and benchmark methods in the real case.

Control Charts														
Metrics	T^2	CCUSUM	EWMA	SVM	DT	RF	NN	KNN	SOM	WT	TCUSUM-1-Mean	TCUSUM-2-Mean	TCUSUM-1	TCUSUM-2
ARL_1	3.01	2.64	1.72	1.43	3.21	2.05	1.97	2.49	64.39	5.38	1.90	2.43	1.68	1.98
Accuracy	0.6848	0.9939	0.9901	0.8291	0.6514	0.6514	0.7403	0.6810	0.5823	0.5132	0.9978	0.9953	0.9983	0.9975
Precision	0.9999	0.9993	0.9997	0.9361	0.7527	0.5887	0.7624	0.6273	0.6659	0.4439	0.9994	0.9996	0.9995	0.9995
Recall	0.4747	0.9905	0.9839	0.5844	0.2970	0.7472	0.6802	0.5940	0.0479	0.3449	0.9970	0.9925	0.9977	0.9964
F1-Score	0.6432	0.9949	0.9917	0.7195	0.4259	0.6585	0.7189	0.6102	0.0894	0.3882	0.9982	0.9960	0.9986	0.9979

observed data of the IoT network system in the normal state by setting $ARL_0 = 1000$ and repeating the procedure 1000 times. Then, we evaluate the monitoring performance of our proposed method and the benchmark methods. We have 100 under-attack processes of the IoT network system. Each under-attack process has 500 time points, and is in the normal state in the beginning and becomes under attack at one, two, and three nodes starting from the time point 200. We monitor the under-attack processes using the proposed method and the benchmark methods. Tables 5 and 6 present the average values of the metrics of the proposed method and the benchmark methods based on the 100 under-attack processes. For network modeling in Table 5, the proposed method has smaller ARL_1 values and larger values of Accuracy, Precision, Recall, and F1-score than NONE + TCUSUM-1 and NONE + TCUSUM-2, GP + TCUSUM-1 and GP + TCUSUM-2 with a simple multivariate spatio-temporal covariance function, VAR + TCUSUM-1 and VAR + TCUSUM-2 that only consider multivariate temporal correlation, as well as STCAR + TCUSUM-1 and STCAR + TCUSUM-2 that omit the correlation among signals, which indicates the effectiveness of the proposed MSTa for modeling the multivariate spatio-temporal characteristics of the network system. For network monitoring in Table 6, the proposed method performs better than the benchmark methods, which indicates the timeliness and stability of the proposed method for network monitoring. In addition, by simultaneously considering the characteristic quantities of the mean change and the variance change, the proposed TCUSUM-1 and TCUSUM-2 perform better than TCUSUM-1-Mean and TCUSUM-2-Mean with lower ARL_1 values.

6. Conclusion

The modeling and monitoring of a network system has been reported to be a challenging task in the literature, due to the complex spatio-temporal characteristics, multiple types of sensor signals, and the unknown network structure. In this article, we propose a multivariate spatio-temporal modeling and monitoring methodology for a network system by using multiple types of sensor signals. For the modeling of the

network system, we first propose an MSTa model that integrates a GMRF and a vector autoregressive structure to fully characterize the spatio-temporal correlation of the network system. Specifically, we develop an iterative model learning algorithm, which combines the Bayesian inference, least squares, and a sum square error-based optimization method, to learn the network structure and estimate parameters in the proposed MSTa model. For the monitoring of the network system, we then propose two multivariate spatio-temporal control schemes, i.e., TCUSUM-1 and TCUSUM-2, which construct the control statistics by excluding the temporal dependence from the spatio-temporal characteristics. The results of the numerical experiments and a case study of an IoT network system demonstrate that the proposed method achieves a much better modeling and monitoring performance than a list of existing benchmark methods.

There are two related topics that are worth studying as future works. First, network systems are vulnerable to attacks with various strength levels in engineering practice. We will extend our proposed method for quantifying the strength levels of attacks when monitoring a network system. Second, network systems (e.g., IoT) leverage a diverse set of computing resources, exposing to various types of attacks that threaten network systems in various aspects. Identifying the attack type when monitoring network systems is a potential topic for future study based on our proposed method.

Funding

This work was supported in part by the National Science Foundation under Grant 1637772; in part by the National Science Foundation of China under Grant 72101148; and in part by the Shanghai Sailing Program under Grant 21YF1420100. The authors acknowledge the support from the funding agencies.

Notes on contributors

Di Wang received the BS degree in industrial engineering from Nankai University, Tianjin, China, in 2015, and the PhD degree in management science and engineering from Peking University, Beijing, China, in 2020. She is currently an assistant professor with the Department of Industrial Engineering and Management, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China. Her research interests include statistical modeling of spatiotemporal data

and artificial intelligence of complex engineering systems. Dr. Wang is also a member of IEEE, INFORMS, and IISE.

Fangyu Li received a BS degree in automatic control science and engineering from Beihang University in 2009, an MS degree in automation from Tsinghua University in 2013, and a PhD degree in computational geophysics from The University of Oklahoma in 2017 respectively. Dr. Li did his postdoc research at the School of Electrical and Computer Engineering, University of Georgia between 2017 and 2020. He is currently a professor with the Faculty of Information Technology, Beijing University of Technology. His research interests include IoT, dynamic system analysis, complex system modeling, data-driven methods, and distributed algorithms.

Kaibo Liu received a BS degree in industrial engineering and engineering management from the Hong Kong University of Science and Technology in 2009, and an MS degree in statistics and a PhD degree in industrial engineering from the Georgia Institute of Technology in 2011 and 2013, respectively. He is currently an associate professor with the Department of Industrial and Systems Engineering, University of Wisconsin-Madison, where he is also the associate director of the UW-Madison IoT Systems Research Center. His research interests include system informatics, industrial big data analytics, and data fusion for process modeling, monitoring, diagnosis, prognostics, and decision making. Dr. Liu is also a member of ASQ, INFORMS, SME, IEEE, and IISE.

ORCID

Di Wang  <http://orcid.org/0000-0001-7030-6521>
 Fangyu Li  <http://orcid.org/0000-0003-2340-3622>
 Kaibo Liu  <http://orcid.org/0000-0003-2863-5748>

References

- Altman, N.S. (1992) An introduction to kernel and nearest-neighbor non-parametric regression. *American Statistician*, **46**(3), 175–185.
- Ambikavathi, C. and Srivatsa, S.K. (2020) Predictor selection and attack classification using random forest for intrusion detection. *Journal of Scientific and Industrial Research*, **79**(5), 365–368.
- Bodnar, O. and Schmid, W. (2017) CUSUM control schemes for monitoring the covariance matrix of multivariate time series. *Statistics*, **51**(4), 722–744.
- Boullousa-Falces, D., Barrera, J.L., Lopez-Arraiza, A., Menendez, J. and Solaetxe, M.A.G. (2017) Monitoring of fuel oil process of marine diesel engine. *Applied Thermal Engineering*, **127**, 517–526.
- Cannon, J., Krokmal, P.A., Chen, Y. and Murphey, R. (2011) Detection of temporal changes in psychophysiological data using statistical process control methods. *Computer Methods and Programs in Biomedicine*, **107**(3), 367–381.
- Cressie, N. and Johannesson, G. (2008) Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society: Series B*, **70**(1), 209–226.
- Crosier, R.B. (1998) Multivariate generalizations of cumulative sum quality-control schemes. *Technometrics*, **30**(3), 291–303.
- Di Giacinto, V. (2010) On vector autoregressive modeling in space and time. *Journal of Geographical Systems*, **12**, 125–154.
- Dobra, A., Hans, C., Jones, B., Nevins, J.R., Yao, G. and West, M. (2004) Sparse graphical models for exploring gene expression data. *Journal of Multivariate Analysis*, **90**(1), 196–212.
- Friedman, J., Hastie, T. and Tibshirani, R. (2008) Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, **9**(3), 432–441.
- Furrer, R., Genton, M.G. and Nychka, D. (2006) Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, **15**(3), 502–503.
- Gao, W. and Ye, W. (2019) A min-max conditional covariance algorithm for structure learning of Gaussian graphical models. *Statistical Analysis and Data Mining*, **12**(1), 12–22.
- Garthoff, R. and Otto, P. (2017) Control charts for multivariate spatial autoregressive models. *Asia Advances in Statistical Analysis*, **101**, 67–94.
- George, E.I. and McCulloch, R.E. (1997) Approaches for Bayesian variable selection. *Statistica Sinica*, **7**(1997), 339–373.
- Gombay, E. and Serban, D. (2009) Monitoring parameter change in AR(p) time series models. *Journal of Multivariate Analysis*, **100**(4), 715–725.
- Grimshaw, S.D., Blades, N.J. and Miles, M.P. (2013) Spatial control charts for the mean. *Journal of Quality Technology*, **45**(2), 130–148.
- ISS Source (2019) IoT attack losses at \$330K and rising. <https://www.issource.com/iot-attack-losses-at-330k-and-rising/>. (accessed 31 May, 2019)
- Jiang, W., Han, S.W., Tsui, K.L. and Woodall, W.H. (2011) Spatio-temporal surveillance methods in the presence of spatial correlation. *Statistics in Medicine*, **30**(5), 569–583.
- Kehoe, B., Patil, S., Abbeel, P. and Goldberg, K. (2015) A survey of research on cloud robotics and automation. *IEEE Transactions on Automation Science and Engineering*, **12**(2), 398–409.
- Kontar, R., Zhou, S. and Horst, J. (2016) Estimation and monitoring of key performance indicators of manufacturing systems using the multi-output Gaussian process. *International Journal of Production Research*, **55**(8), 2304–2319.
- Lee, J., Hur, Y., Kim, S.H. and Wilson, J.R. (2012) Monitoring nonlinear profiles using a wavelet-based distribution-free CUSUM chart. *International Journal of Production Research*, **50**(22), 6574–6594.
- Li, F. and Zhang, N.R. (2010) Bayesian variable selection in structured high-dimensional covariate spaces with applications in genomics. *Journal of the American Statistical Association*, **105**(491), 1202–1214.
- 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**(3), 5224–5231.
- Lin, Z., Wang, T., Yang, C. and Zhao, H. (2017) On joint estimation of Gaussian graphical models for spatial and temporal data. *Biometrics*, **73**(3), 769–779.
- Liou, S., Kurniadi, D., Zheng, B., Xie, W., Tien, C. and Jong, G. (2018) Classification of biomedical signal on IoT platform using support vector machine, in *IEEE International Conference on Applied System Invention*, IEEE Press, Piscataway, NJ, pp. 50–53.
- Liu, K. and Shi, J. (2015) Internet of Things (IoT)-enabled system informatics for service decision making: Achievements, trends, challenges, and opportunities. *IEEE Intelligent Systems*, **30**(6), 18–21.
- Liu, X., Gopal, V. and Kalagnanam, J. (2018) A spatio-temporal modeling approach for weather radar reflectivity data and its applications in tropical southeast Asia. *The Annals of Applied Statistics*, **12**(1), 378–407.
- Lucas, J.M. and Saccucci, M.S. (1990) Exponentially weighted moving average control schemes: Properties and enhancements. *Technometrics*, **32**(1), 1–12.
- Mariella, L. and Tarantino, M. (2010) Spatial temporal conditional auto-regressive model: A new autoregressive matrix. *Austrian Journal of Statistics*, **39**(3), 223–244.
- Montgomery, D.C. (2009) *Introduction to Statistical Quality Control*, 6th ed., Wiley, New York, NY.
- Moon, D., Im, H., Kim, I. and Park, J.H. (2017) DTB-IDS: An intrusion detection system based on decision tree using behavior analysis for preventing APT attacks. *The Journal of Supercomputing*, **73**(7), 2881–2895.
- Shao, X. and Zhang, X. (2010) Testing for change points in time series. *Journal of the American Statistical Association*, **105**(491), 1228–1240.
- Spiegelhalter, D., Sherlaw-Johnson, C., Bardsley, M., Blunt, I., Wood, C. and Grigg, O. (2012) Statistical methods for healthcare regulation: Rating, screening and surveillance. *Journal of the Royal Statistical Society: Series A*, **175**(1), 1–47.
- Valero, M., Li, F., Wang, S., Lin, F. and Song, W. (2019) Real-time cooperative analytics for ambient noise tomography in sensor networks. *IEEE Transactions on Signal and Information Processing over Networks*, **5**(2), 375–389.
- Vesanto, J. and Alhoniemi, E. (2000) Clustering of the self-organizing map. *IEEE Transactions on Neural Networks*, **11**(3), 586–600.

- Wang, D., Liu, K. and Zhang X. (2019) Modeling of a three-dimensional dynamic thermal field under grid-based sensor networks in grain storage. *IISE Transactions*, **51**(5), 531–546.
- Wang, D. and Zhang, X. (2019) Modeling of a 3D temperature field by integrating a physics-specific model and spatiotemporal stochastic processes, *Applied Sciences*, **9**(10), 1–13.
- Wang, D., Liu, K. and Zhang, X. (2020a) Spatiotemporal multitask learning for 3-D dynamic field modeling. *IEEE Transactions on Automation Science and Engineering*, **17**(2), 708–721.
- Wang, D., Liu, K. and Zhang, X. (2020b) A spatiotemporal prediction approach for a 3D thermal field from sensor networks. *Journal of Quality Technology*. doi: [10.1080/00224065.2020.1851618](https://doi.org/10.1080/00224065.2020.1851618).
- William, W.S.W. (2019) Space-time series models. *Multivariate Time Series Analysis and Applications*, John Wiley and Sons Ltd, Hoboken, NJ, pp. 261–299.
- Woodall, W.H. (2006) The use of control charts in health-care and public-health surveillance. *Journal of Quality Technology*, **38**(2), 89–104.
- Woodall, W.H. and Ncube, M.M. (1985) Multivariate CUSUM quality-control procedures. *Technometrics*, **27**(3), 285–292.
- Wu, N. and Wang, Q. (2011) Experimental studies on damage detection of beam structures with wavelet transform. *International Journal of Engineering Science*, **49**(3), 253–261.
- Xian, X., Ye, H., Wang, X. and Liu, K. (2020) Spatiotemporal modeling and real-time prediction of origin-destination traffic demand. *Technometrics*, **63**(1), 77–89.
- Xu, L. and Huang, Q. (2012) Modeling the interactions among neighboring nanostructures for local feature characterization and defect detection. *IEEE Transactions on Automation Science and Engineering*, **9**(4), 745–754.
- Xu, Y. and Choi, J. (2012) Spatial prediction with mobile sensor networks using Gaussian processes with built-in Gaussian Markov random fields. *Automatica*, **48**(8), 1735–1740.
- Zhang, B., Sang, H. and Huang, J.Z. (2015) Full-scale approximations of spatio-temporal covariance models for large datasets. *Statistica Sinica*, **25**(1), 99–114.
- Zhou, M., Fortino, G., Shen, W., Mitsugi, J., Jobin, J. and Bhattacharyya, R. (2016) Guest editorial special section on advances and applications of Internet of Things for smart automated systems. *IEEE Transactions on Automation Science and Engineering*, **13**(3), 1225–1229.