

Designing dynamic alarm limits and adjusting manipulated variables for multivariate systems

Yan Yu, Jiandong Wang*, and Zijian Ouyang

Abstract—Alarms systems are of paramount importance for safe and efficient operations of industrial plants. This paper proposes a method to design dynamic alarm limits and adjust manipulated variables for multivariate systems. A hyper-ellipsoid model is built from historical normal data points to represent a normal operating zone. Dynamic alarm limits of each involved process variable are mathematically defined and calculated based on the normal operating zone. Adjustment amplitudes of some manipulated variables are determined in order to remove occurring alarms, by exploiting reduced normal operating zones in parallel coordinates. The effectiveness of the proposed method is illustrated via numerical and industrial examples.

Index Terms—Industrial alarm systems; dynamic alarm limits; alarm removal; parallel coordinates.

I. INTRODUCTION

AN alarm system is the collection of hardware and software that detects an alarm state, communicates the state to operators, and records changes in the alarm state [1, 2]. Alarm systems are of paramount importance for safe and efficient operations of industrial plants. The rationalization of alarm systems has received persisting attentions from both industry and academic communities [3–6].

Alarm limits, also known as alarm thresholds or alarm trippoints, are the critical parameters for generating alarms. The design of alarm limits for univariate systems has been extensively studied in literature (see, e.g., [7] and references therein). It is a well-known fact that many process variables are related to each other via physical connections, e.g., mass and energy conservation laws. Thus, alarm limits of these process variables should be related to each other; otherwise, false alarms and missed alarms may occur. As an illustration, Fig. 1 presents some data points for two related variables operating in a normal condition. If their alarm limits are constant values, then a rectangular area is formulated. As a result, false (missed) alarms may occur, shown as the circle (pentagram) points in Fig. 1.

For the design of alarm limits for multivariate systems, related studies are rather limited. Cheon *et al.* [8] used an expert system to design multivariate alarm systems in nuclear

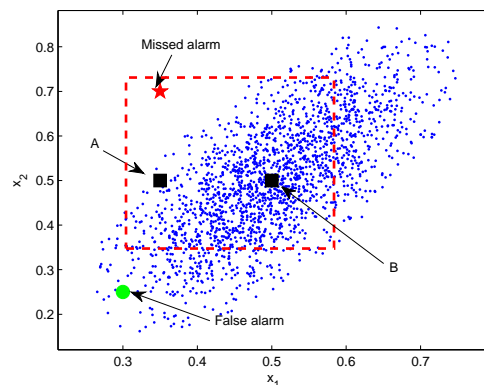


Fig. 1: Normal data points (blue dot) of x_1 and x_2 with fixed alarm limits (red dash), resulting the presence of false alarms (green circle) and missed alarms (red pentagram).

power plants. Fischer *et al.* [9] designed a Bayesian classifier to optimize alarm limits in thermal power plants. Yang *et al.* [10] advocated alarm limits achieving the maximum correlation coefficients among alarm and process variables. Gupta *et al.* [11] rationalized alarm limits for multivariate systems by integrating several techniques such as wavelet analysis and qualitative trend analysis. Alrowaie *et al.* [12] investigated a method to design alarm limits based on particle filters for multivariate nonlinear stochastic systems. Zhu *et al.* [13] claimed dynamic alarm limits being adaptive for different pre-defined operating stages. Han *et al.* [14] optimized alarm limits for multivariate systems by minimizing false and missed alarm probabilities based on joint probability densities of process variables.

A closely-related research topic is the multivariate statistical process monitoring (MSPM) [15–25], where many latent-variable based-methods have been proposed, such as the principal component analysis (PCA), the partial least squares and the independent component analysis. The PCA is of interest here, because it has a close relationship with the proposed method in this paper. Under the assumption that process variables are Gaussian-distributed, the PCA transforms original process variables into a set of orthogonal principal components, selects the principal components having largest variances, and constructs T^2 and SPE statistics based on the selected principal components. The T^2 statistic is a measure of the variation within the selected principal components, and the SPE statistic is a measure of the amount of variation not captured by the selected principal components. For a new

This work was supported by the National Natural Science Foundation of China under Grant No. 61433001, and the Research Fund for the Taishan Scholar Project of Shandong Province of China (Corresponding author: Jiandong Wang).

Y. Yu and Z. Ouyang are with the College of Engineering, Peking University, Beijing, China (e-mail: yuyan_coe@pku.edu.cn, oyzj496@pku.edu.cn). J. Wang (corresponding author; phone: +86 (532) 86057152) is with the College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao, Shandong Province, China (e-mail: jiandong@sdust.edu.cn).

data point, the corresponding T^2 and/or SPE statistics are calculated and compared to designed control limits, which represent the boundary of the normal operation zone of a multivariate process, to tell whether the data point is in the normal condition.

Once alarms are occurring, one next step is to take corrective actions to deal with abnormalities leading to occurring alarms. This research topic has been rarely studied in literature. Brooks *et al.* [26] presented a geometric process control method for defining and managing alarms for multivariate systems, but did not provide any technical details. A common practice is to adjust some variables by operators, mostly in a trial and error manner, based on process knowledge and work experience.

This paper proposes a method to design dynamic alarm limits and adjust manipulated variables for a multivariate system composed by Gaussian-distributed process variables. Such a system is often present in the process industry, where many MSPM methods (e.g., the PCA) have been successfully applied, e.g., Tennessee Eastman chemical reaction process [27], the boiler combustion system [28], the hydrotreating process [29], and the continuous stirred tank reactor [30]. Main contributions of the proposed method are two-fold: 1) Theoretical expressions of dynamic alarm limits for each process variable are developed based on the so-called normal operating zone (NOZ). The NOZ is described by a hyper-ellipsoid model built from historical normal data points, subject to a pre-defined false alarm rate. 2) To remove occurring alarms, manipulated variables to be adjusted are determined and the theoretical expressions of adjustment amplitudes are given. By fixing values of non-manipulated variables, a reduced NOZ is represented by hyperbolas in parallel coordinates. The manipulated variables to be adjusted are the ones being outside of the hyperbolas. The adjustment amplitudes of these manipulated variables are calculated in turns on the basis of the reduced NOZ.

The novelties of the proposed method are given as follows.

- Alarm limits of each involved process variable are dynamically changing: alarm limits of one process variable are dependent on the values of other process variables, so that alarm limits are changing with the positions of data points in the NOZ. Taking two data points A and B in Fig. 1 as examples, alarm limits of x_1 and x_2 for A and B should be very different, because A is close to the boundary of the NOZ, while B is next to the center of the NOZ. As a comparison, static alarm limits are omnipresent in contemporary industrial practices, that is, alarm limits of one process variable are isolated from other related process variables; as a result, false alarms and missed alarms may occur, as previously explained in Fig. 1. On the contrary, dynamic alarm limits from the proposed method yield less false/missed alarms, as illustrated later in Example 2 at Section V.
- The proposed method provides alarm limits for original process variables, whereas the MSPM methods design alarm limits for latent variables, not for original process variables. In particular, the PCA designs alarm limits for T^2 and/or SPE statistics based on selected principal components that are obtained by an orthogonal trans-

formation of original process variables. Doing so makes alarm limits not having the same physical meanings as original process variables, so that alarm limits are difficult for industrial plant operators to accept. By contrast, alarm limits from the proposed method have the same physical meanings as original process variables, so that they are easily acceptable.

- Manipulated variables to be adjusted and their adjustment amplitudes are determined by exploiting the reduced NOZ in parallel coordinates. The proposed method to adjust manipulated variables is inspired by the idea in [26]; however, no technical details were provided therein. Hence, to the best of our knowledge, the paper may be the first one providing a complete study on adjusting manipulated variables for alarm removal.

The rest of this paper is organized as follows. Section II describes the problem to be solved. Section III designs dynamic alarm limits of each process variable. Section IV is on the adjustment of manipulated variables. Examples are provided in Section V to illustrate the proposed method. Section VI makes some concluding remarks.

II. PROBLEM DESCRIPTION

Consider a multivariate system X composed by n analog process variables x_1, x_2, \dots, x_n . Historical data points in the normal condition are available, being represented as $\{X(t) \mid t \in [1, N]\}$ with $X(t) := [x_1(t), x_2(t), \dots, x_n(t)]$. Here t and N are the sampling time index and the number of data points, respectively. The process variables are assumed to be Gaussian-distributed. In addition, stationary processes are assumed, so that new data points of x_1, x_2, \dots, x_n in the normal condition should be distributed in a consistent way as the historical normal data points. Note that both Gaussian-distributed process variables and stationary processes are standard assumptions for the PCA as well as some other MSPM methods [15–25].

The first objective of this paper is to design dynamic alarm limits for each process variable based on the relationships among mutually-dependent process variables x_1, x_2, \dots, x_n . For a new data point $X(t_0)$, a comparison between $x_i(t_0)$ and the low alarm limit $l_i(t_0)$ or high alarm limit $h_i(t_0)$ of x_i should tell whether $X(t_0)$ is in the normal condition, and how far $X(t_0)$ is deviating away from the normal condition.

If $X(t_0)$ is not in the normal condition, then some corrective actions should be taken to move X back to the normal condition. It is determined by some prior process knowledge that the first m variables $\{x_1, x_2, \dots, x_m\}$ can not be manipulated directly, while the rest $(n-m)$ variables $\{x_{m+1}, x_{m+2}, \dots, x_n\}$ are manipulated variables. The second objective of this paper is to decide the manipulated variables to be adjusted and their adjustment amplitudes. The first m variables are sorted in the order of importance. More critical variables are placed ahead of less critical ones, in order to ensure that critical alarms are not ignored. The order of the manipulated variables is based on some practical considerations such as difficulties and costs of changing these variables (see Example 3 appeared later in Section V for a concrete example; the appendix of

this paper proves that the order of process variables has no effects on the obtained results including dynamic alarm limits and adjustment amplitudes of manipulated variables). Physical devices to realize the adjustments of manipulated variables are usually known to industrial plant operators.

III. DESIGN OF DYNAMIC ALARM LIMITS

This section first establishes a hyper-ellipsoid model to represent the normal operating zone, and provides detailed steps to design dynamic alarm limits afterwards.

A. Establishment of the normal operating zone

Normal data points may formulate a cluster, referred to as the normal operating zone (NOZ) in the sequel. Establishing the NOZ is subject to a tradeoff between false and missed alarms. A false alarm is the one that is raised when the multivariate system X behaves normally, and a missed alarm occurs when X is in an abnormal condition but no alarm is raised. If the NOZ is too small, then false alarms would occur; however, if the NOZ is too large, then missed alarms might present. A general fact is that data points far away from the cluster center are sparsely distributed. Thus, it is a reasonable choice that the NOZ contains normal data points closer to the cluster center, not all data points.

There are a number of well known clustering algorithms in literature [31]. This paper uses the one proposed by Maestri *et al.* [32]. The algorithm of formulating the NOZ is given as follows.

Algorithm 1:

- 1) In order to avoid that some variables with large values erroneously play dominate roles, the algorithm preprocess the raw data by normalizing values of each variable into the range $[0, 1]$. For the raw data point $\tilde{X}(t)$, the normalized data point $X(t)$ is obtained as $X(t) := [x_1(t), x_2(t), \dots, x_n(t)]$, where

$$x_i(t) = \frac{\tilde{x}_i(t) - \min(\tilde{x}_i)}{\max(\tilde{x}_i) - \min(\tilde{x}_i)}.$$

Here $\min(\tilde{x}_i)$ and $\max(\tilde{x}_i)$ denote the minimum and maximum of the variable \tilde{x}_i with $i \in [1, n]$, respectively.

- 2) Determine the number r of data points to be included by the NOZ based on the false alarm rate (FAR) $p \in (0, 1)$ as $r = \lfloor N - pN \rfloor$. The floor function $\lfloor \cdot \rfloor$ returns the largest integer that is not greater than the operand.
- 3) Given a subset s of randomly selected r points from the normal data points, calculate the mean vector μ_s and covariance matrix W_s of s as

$$\mu_s = \frac{1}{r} \sum_{X(t) \in s} X(t), \quad (1)$$

and

$$W_s = \sum_{X(t) \in s} [X(t) - \mu_s]^T [X(t) - \mu_s]. \quad (2)$$

- 4) Compute the Mahalanobis distance from $X(t)$ to μ_s ,

$$d^2(t) = [X(t) - \mu_s] W_s^{-1} [X(t) - \mu_s]^T,$$

where $t = 1, 2, \dots, N$.

- 5) Sort the Mahalanobis distances $d^2(t)$ in the ascending order.
- 6) Construct the new subset s' , composed by r data points corresponding to the first r sorted Mahalanobis distances. Compute the new mean vectors $\mu_{s'}$ as (1) and covariance matrix $W_{s'}$ as (2).
- 7) If $\det(W_{s'}) \neq \det(W_s)$, where $\det(\cdot)$ represents the determinant of the operand matrix, then W_s is updated to $W_{s'}$, μ_s is updated to $\mu_{s'}$, and the algorithm repeats the iteration from Steps 3 to 7. Otherwise, the iteration is terminated and the algorithm goes to the next step.
- 8) Represent the NOZ by a hyper-ellipsoid model,

$$[X(t) - C] P [X(t) - C]^T = 1, \quad (3)$$

where the central vector C and the feature matrix P respectively are equal to μ_s and W_s^{-1} , i.e., $C = \mu_s$ and $P = W_s^{-1}$.

B. Definition of dynamic alarm limits

First, dynamic alarm limits are defined. The most common way in raising alarms is to compare the value of a process variable x_i to two constant values l_i and h_i as the low and high alarm limits respectively, i.e., an alarm occurs if $x_i(t) < l_i$ and/or $x_i(t) > h_i$ (see, e.g., [1, 2]). In this paper, dynamic alarm limits of each involved process variable are designed based on the NOZ. Note that the word ‘dynamic’ means that alarm limits may change with the position of data points, instead of indicating that the multivariate system X is time-varying. For a normal data point $X(t)$, its projections on the surface of the hyper-ellipsoid model in (3) are defined as the dynamic alarm limits. In this way, dynamic alarm limits of $X(t)$ reflect the distances of $X(t)$ away from the NOZ surface. When some process variables are close to their dynamic alarm limits, industrial plant operators should pay more attentions to the multivariate system for preventing abnormalities. If $X(t)$ is an abnormal data point that locates outside the NOZ, and its projections on the surface of the hyper-ellipsoid model might not exist, then dynamic alarm limits of a corresponding normal data point $X'(t)$ are defined as dynamic alarm limits of $X(t)$. Here $X'(t)$ is the intersection of a line segment L with the surface of the hyper-ellipsoid model, and L represents the line segment whose endpoints are $X(t)$ and the hyper-ellipsoid center C . Dynamic alarm limits of an abnormal data point can tell the distances of $X(t)$ deviating away from the NOZ.

Fig. 2(a) gives an example to illustrate dynamic alarm limits for a normal data point. The asterisk represents a normal data point $X(t)$, and four circles are the projections of $X(t)$ on an ellipse. The dynamic alarm limits of x_1 and x_2 are $l_1(t) = -0.90$, $h_1(t) = 0.90$, and $l_2(t) = -0.56$, $h_2(t) = 0.56$. Fig. 2(b) presents an example for an abnormal data point $X(t)$. The red pentagram represents the corresponding normal data point $X'(t)$ of $X(t)$, and two red circles are the projections of $X'(t)$ on the ellipse. The dynamic alarm limits of x_1 and x_2 are $l_1(t) = -0.49$, $h_1(t) = 0.49$, and $l_2(t) = -0.61$, $h_2(t) = 0.61$.

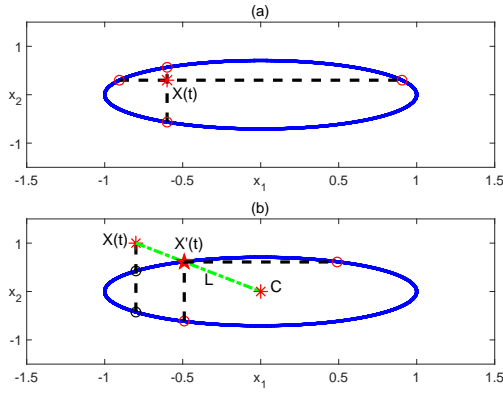


Fig. 2: (a) A normal data point (red asterisk) and its projections on the ellipse (red circles). (b) An abnormal data point $X(t)$ (red asterisk) and its corresponding normal data point $X'(t)$ (pentagram).

Second, dynamic alarm limits are calculated as follows. The n -dimensional hyper-ellipsoid model in (3) can be rewritten as

$$F(X(t)) = [X(t) \quad 1] \bar{P} \begin{bmatrix} X(t)^T \\ 1 \end{bmatrix} = 0, \quad (4)$$

where

$$\bar{P} = \begin{bmatrix} P & -PC^T \\ -CP & CPC^T - 1 \end{bmatrix}.$$

To obtain dynamic alarm limits of x_i , denote

$$U(t) = [x_1(t), x_2(t), \dots, x_{i-1}(t)],$$

and

$$V(t) = [x_{i+1}(t), x_{i+2}(t), \dots, x_n(t)].$$

Eq. (4) becomes

$$[U(t) \quad x_i(t) \quad V(t) \quad 1] \begin{bmatrix} \bar{P}_{11} & \bar{P}_{12} & \bar{P}_{13} & \bar{P}_{14} \\ \bar{P}_{12}^T & \bar{P}_{22} & \bar{P}_{23} & \bar{P}_{24} \\ \bar{P}_{13}^T & \bar{P}_{23}^T & \bar{P}_{33} & \bar{P}_{34} \\ \bar{P}_{14}^T & \bar{P}_{24}^T & \bar{P}_{34}^T & \bar{P}_{44} \end{bmatrix} \begin{bmatrix} U(t)^T \\ x_i(t) \\ V(t)^T \\ 1 \end{bmatrix} = 0, \quad (5)$$

where

$$\begin{cases} \bar{P}_{11} = \bar{P}(1:i-1, 1:i-1), \\ \bar{P}_{12} = \bar{P}(1:i-1, i), \\ \bar{P}_{13} = \bar{P}(1:i-1, i+1:n), \\ \bar{P}_{14} = \bar{P}(1:i-1, n+1), \\ \bar{P}_{22} = \bar{P}(i, i), \\ \bar{P}_{23} = \bar{P}(i, i+1:n), \\ \bar{P}_{24} = \bar{P}(i, n+1), \\ \bar{P}_{33} = \bar{P}(i+1:n, i+1:n), \\ \bar{P}_{34} = \bar{P}(i+1:n, n+1), \\ \bar{P}_{44} = \bar{P}(n+1, n+1). \end{cases}$$

Here the symbol $\bar{P}(i_1 : i_2, j_1 : j_2)$ represents the matrix of the elements in the i_1 -th, (i_1+1) -th, \dots , i_2 -th rows and the j_1 -th, (j_1+1) -th, \dots , j_2 -th columns of \bar{P} . Eq. (5) is a matrix equation of $x_i(t)$,

$$a_1 x_i^2(t) + 2a_2 x_i(t) + a_3 = 0, \quad (6)$$

where

$$\begin{cases} a_1 = \bar{P}_{22}, \\ a_2 = U(t)\bar{P}_{12} + V(t)\bar{P}_{23}^T + \bar{P}_{24}^T, \\ a_3 = U(t)\bar{P}_{11}U(t)^T + V(t)\bar{P}_{33}V(t)^T + \bar{P}_{44} \\ \quad + 2V(t)\bar{P}_{13}^T U(t)^T + 2U(t)\bar{P}_{14} + 2V(t)\bar{P}_{34}. \end{cases}$$

The solutions of (6) are

$$\begin{cases} l_i(t) = (-a_2 - \sqrt{a_2^2 - a_1 a_3})/a_1, \\ h_i(t) = (-a_2 + \sqrt{a_2^2 - a_1 a_3})/a_1. \end{cases} \quad (7)$$

Thus, dynamic alarm limits of $X(t)$ are obtained via (7) by setting the value of i to 1, 2, \dots , n .

If eq. (6) does not have two solutions, then $X(t)$ is an abnormal data point. Its dynamic alarm limits are the same as those of a corresponding normal data point $X'(t)$. It is a point in the line segment L between $X(t)$ and the hyper-ellipsoid center C ,

$$X'(t) = \theta X(t) + (1 - \theta)C, \quad 0 \leq \theta \leq 1. \quad (8)$$

It is also on the surface of the hyper-ellipsoid model,

$$[X'(t) - C]P[X'(t) - C]^T = 1. \quad (9)$$

Equations (8) and (9) lead to

$$[X(t) - C]P[X(t) - C]^T \theta^2 = 1. \quad (10)$$

For a certain time index t , (10) is a quadratic equation of θ , and its solutions are

$$\begin{cases} \theta_1 = -\sqrt{1/[(X(t) - C)P(X(t) - C)^T]}, \\ \theta_2 = \sqrt{1/[(X(t) - C)P(X(t) - C)^T]}. \end{cases}$$

Owing to the constraint $0 \leq \theta \leq 1$, the final solution is

$$\theta = \sqrt{1/[(X(t) - C)P(X(t) - C)^T]}. \quad (11)$$

Using θ in (11), $X'(t)$ in (9) is obtained and its dynamic alarm limits are available through (7).

Finally, after obtaining dynamic alarm limits for all process variables x_1, x_2, \dots, x_n , it is ready to tell whether $X(t)$ is in the normal condition. That is, if all variables in $X(t)$ are inside their own dynamic alarm limits, then $X(t)$ is a normal data point; otherwise, it is an abnormal data point. In addition, the dynamic alarm limits measure the distances between $X(t)$ and the NOZ surface.

Remark: Owing to the assumption that the process variables x_1, x_2, \dots, x_n are Gaussian-distributed, the NOZ can be represented by a hyper-ellipsoid model in (3). Thus, the proposed method has a close relationship with the PCA. In particular, the proposed method and the PCA have exactly the same performance such as the false alarm rate, if no dimension reduction is done for the PCA. The difference is that alarm limits $l_i(t)$ and $h_i(t)$ in (7) are designed for the i -th process variable x_i , whereas the PCA designs alarm limits for T^2 and/or SPE statistics based on selected principal components instead of the original process variables x_1, x_2, \dots, x_n . Such a difference is significant, since the alarm limits $l_i(t)$ and $h_i(t)$ have the same physical meanings as the i -th process variable x_i so that they are easily acceptable for

industrial plant operators, but the alarm limits for T^2 and/or SPE statistics do not have the same physical meanings as original process variables, so that they are difficult to accept.

IV. ADJUSTMENT OF MANIPULATED VARIABLES

This section presents the method to determine the manipulated variables to be adjusted and their adjustment amplitudes.

First, an abnormal data point $X(t)$ will return to the NOZ when it is adjusted to its corresponding normal data point $X'(t)$. However, if some process variables can not be manipulated directly, then $X(t)$ may not be adjusted to $X'(t)$. As an example, suppose that only x_2 in Fig. 2(b) is a manipulated variable and x_1 is not. Thus, $X(t)$ can not be adjusted to $X'(t)$ along the line L . Nevertheless, if x_2 is adjusted to the range $[-0.56, 0.56]$, then $X(t)$ will return back to the NOZ even though the value of x_1 is unchanged.

Second, as assumed in Section II, the first m variables can not be manipulated directly, and the rest $(n-m)$ variables are the manipulated variables. Denote

$$W(t) = [x_1(t), x_2(t), \dots, x_m(t)]$$

and

$$\bar{X}(t) = [x_{m+1}(t), x_{m+2}(t), \dots, x_n(t)].$$

Eq. (4) is rewritten as

$$[W(t) \ \bar{X}(t) \ 1] \begin{bmatrix} \bar{P}'_{11} & \bar{P}'_{12} & \bar{P}'_{13} \\ (\bar{P}'_{12})^T & \bar{P}'_{22} & \bar{P}'_{23} \\ (\bar{P}'_{13})^T & (\bar{P}'_{23})^T & \bar{P}'_{33} \end{bmatrix} \begin{bmatrix} W(t)^T \\ \bar{X}(t)^T \\ 1 \end{bmatrix} = 0, \quad (12)$$

where

$$\begin{cases} \bar{P}'_{11} = \bar{P}(1:m, 1:m), \\ \bar{P}'_{12} = \bar{P}(1:m, m+1:n), \\ \bar{P}'_{13} = \bar{P}(1:m, n+1), \\ \bar{P}'_{22} = \bar{P}(m+1:n, m+1:n), \\ \bar{P}'_{23} = \bar{P}(m+1:n, n+1), \\ \bar{P}'_{33} = \bar{P}(n+1, n+1). \end{cases}$$

By fixing the values of $W(t)$, eq. (12) leads to

$$[\bar{X}(t) \ 1] \bar{Q} \begin{bmatrix} \bar{X}(t)^T \\ 1 \end{bmatrix} = 0, \quad (13)$$

where

$$\bar{Q} = \begin{bmatrix} \bar{P}'_{22} & (\bar{P}'_{12})^T W(t)^T + \bar{P}'_{23} \\ W(t) \bar{P}'_{12} + (\bar{P}'_{23})^T & W(t) \bar{P}'_{11} W(t)^T + 2W(t) \bar{P}'_{13} + \bar{P}'_{33} \end{bmatrix}.$$

If a data point $X(t)$ is inside the n -dimensional hyper-ellipsoid in (12), then $\bar{X}(t)$ should be inside the $(n-m)$ -dimensional hyper-ellipsoid in (13). Thus, driving an abnormal data point $X(t)$ back to the NOZ is equivalent to moving $\bar{X}(t)$ back to the hyper-ellipsoid in (13).

Next, the $(n-m)$ -dimensional hyper-ellipsoid in (13) is projected into $(n-m-1)$ two-dimensional planes, namely, $x_{m+1}x_{m+2}$ plane, $x_{m+2}x_{m+3}$ plane, \dots , $x_{n-1}x_n$ plane. The $(n-m-1)$ projected ellipses are mapped into $(n-m-1)$ pairs of hyperbolas in parallel coordinates, and adjustment amplitudes of some manipulated variables are calculated. The details are given as follows.

1) *Projection of a hyper-ellipsoid in two-dimensional planes:* The projection of the $(n-m)$ -dimensional hyper-ellipsoid in (13) into the $x_i x_{i+1}$ plane in Cartesian coordinate is an ellipse [33],

$$f([x_i(t), x_{i+1}(t)]) = [x_i(t) \ x_{i+1}(t) \ 1] A \begin{bmatrix} x_i(t) \\ x_{i+1}(t) \\ 1 \end{bmatrix} = 0, \quad (14)$$

where

$$A = \begin{bmatrix} \bar{Q}_{22} - \bar{Q}_{12}^T \bar{Q}_{11}^{-1} \bar{Q}_{12} & \bar{Q}_{23} - \bar{Q}_{12}^T \bar{Q}_{11}^{-1} \bar{Q}_{13} \\ (\bar{Q}_{23} - \bar{Q}_{12}^T \bar{Q}_{11}^{-1} \bar{Q}_{13})^T & \bar{Q}_{33} - \bar{Q}_{13}^T \bar{Q}_{11}^{-1} \bar{Q}_{13} \end{bmatrix},$$

and

$$\begin{cases} \bar{Q}_{11} = \bar{Q}([1:i-1 \ i+2:n-m], [1:i-1 \ i+2:n-m]), \\ \bar{Q}_{12} = \bar{Q}([1:i-1 \ i+2:n-m], i:i+1), \\ \bar{Q}_{13} = \bar{Q}([1:i-1 \ i+2:n-m], n-m+1), \\ \bar{Q}_{22} = \bar{Q}(i:i+1, i:i+1), \\ \bar{Q}_{23} = \bar{Q}(i:i+1, n-m+1), \\ \bar{Q}_{33} = \bar{Q}(n-m+1, n-m+1). \end{cases}$$

2) *Representation of an ellipse in the parallel coordinate:* An ellipse in the $x_i x_{i+1}$ plane has a one-to-one mapping as a pair of hyperbolas in the parallel coordinate [33]. If the distance between two adjacent axes in the parallel coordinate is selected as 1, the mathematical model of the corresponding hyperbolas in the xy plane is

$$g([x(t), y(t)]) = [x(t) \ y(t) \ 1] \begin{bmatrix} d_1 & d_4 & d_5 \\ d_4 & d_2 & d_6 \\ d_5 & d_6 & d_3 \end{bmatrix} \begin{bmatrix} x(t) \\ y(t) \\ 1 \end{bmatrix} = 0, \quad (15)$$

where

$$\begin{cases} d_1 = a_3(a_1 + a_2 + 2a_4) - (a_5 + a_6)^2, \\ d_2 = a_1 a_2 - a_4^2, \\ d_3 = a_2 a_3 - a_6^2, \\ d_4 = a_6(a_1 + a_4) - a_5(a_2 + a_4), \\ d_5 = a_6^2 + a_5 a_6 - a_3(a_2 + a_4), \\ d_6 = a_2 a_5 - a_4 a_6, \end{cases}$$

and

$$\begin{cases} a_1 = A(1, 1), \ a_2 = A(2, 2), \ a_3 = A(3, 3), \\ a_4 = A(1, 2), \ a_5 = A(1, 3), \ a_6 = A(2, 3). \end{cases}$$

The variational range of x_i is obtained by setting $x(t)$ in (15) equal to 0. The resulting equation is

$$d_2 y(t)^2 + 2d_6 y(t) + d_3 = 0,$$

whose solutions are

$$\begin{cases} y_i^l(t) = (-d_6 - \sqrt{d_6^2 - d_2 d_3})/d_2, \\ y_i^h(t) = (-d_6 + \sqrt{d_6^2 - d_2 d_3})/d_2. \end{cases} \quad (16)$$

Similarly, the variational range of x_{i+1} is from (15) using $x(t) = 1$. The resulting equation is

$$d_2 y(t)^2 + 2(d_4 + d_6)y(t) + (d_1 + d_3 + 2d_5) = 0,$$

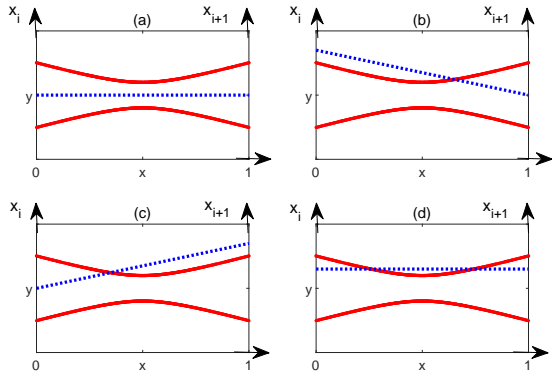


Fig. 3: Determination of the manipulated variables to be adjusted.

whose solutions are

$$\begin{cases} y_{i+1}^l(t) = \frac{-(d_4+d_6) - \sqrt{(d_4+d_6)^2 - d_2(d_1+d_3+2d_5)}}{d_2}, \\ y_{i+1}^h(t) = \frac{-(d_4+d_6) + \sqrt{(d_4+d_6)^2 - d_2(d_1+d_3+2d_5)}}{d_2}. \end{cases}$$

The manipulated variables to be adjusted are determined in the following algorithm.

Algorithm 2:

- If $x_i(t) \notin [y_i^l(t), y_i^h(t)]$ or $x_{i+1}(t) \notin [y_{i+1}^l(t), y_{i+1}^h(t)]$ as presented in Figs. 3(b) and (c), then x_i or x_{i+1} needs to be adjusted.
- If $x_i(t) \in [y_i^l(t), y_i^h(t)]$ and $x_{i+1}(t) \in [y_{i+1}^l(t), y_{i+1}^h(t)]$, then there are two different cases. If the line in the xy plane associated with the point $[x_i(t), x_{i+1}(t)]$ crosses over hyperbolas in (15), or equivalently, $f([x_i(t), x_{i+1}(t)])$ in (26) is larger than zero, as shown in Fig. 3(d), then x_i or x_{i+1} is the one to be adjusted. If there are no crossing points, i.e., $f([x_i(t), x_{i+1}(t)]) \leq 0$, as illustrated in Fig. 3(a), then x_i and x_{i+1} are not the ones to be adjusted.

3) *Calculation of adjustment amplitude ranges:* Without the loss of generality, let x_{i+1} with $i \in [m+1, n-1]$ be a manipulated variable to be adjusted. Given the value of $x_i(t)$, the adjustment amplitude of x_{i+1} can be obtained as follows. The tangent lines of hyperbolas in (15) through the point $[0, x_i(t)]$ is

$$y(t) = kx(t) + x_i(t). \quad (17)$$

Equations (15) and (17) lead to

$$\phi_1 x(t)^2 + 2\phi_2 x(t) + \phi_3 = 0$$

and

$$\begin{cases} \phi_1 = d_1 + d_2 k^2 + 2d_4 k, \\ \phi_2 = d_2 k x_i(t) + d_4 x_i(t) + d_5 + d_6 k, \\ \phi_3 = d_2 x_i(t)^2 + 2d_6 x_i(t) + d_3. \end{cases}$$

The tangent line and one hyperbola have one single intersection so that

$$(2\phi_2)^2 - 4\phi_1\phi_3 = 0,$$

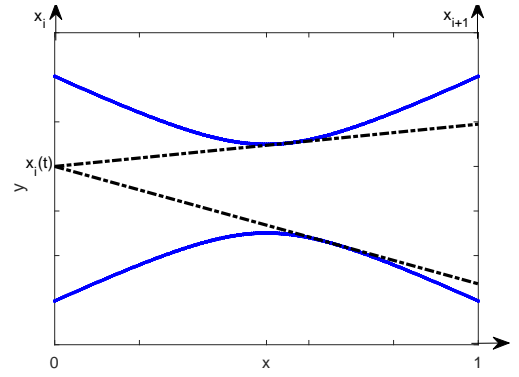


Fig. 4: Hyperbolas (blue solid) and tangent lines (black dot-dash) through the point $[0, x_i(t)]$ in the parallel coordinate.

which leads to

$$\psi_1 k^2 + 2\psi_2 k + \psi_3 = 0, \quad (18)$$

where

$$\begin{cases} \psi_1 = d_6^2 - d_2 d_3, \\ \psi_2 = d_2 d_5 x_i(t) + d_5 d_6 - d_4 d_6 x_i(t) - d_3 d_4, \\ \psi_3 = d_4^2 x_i(t)^2 + 2d_4 d_5 x_i(t) + d_5^2 \\ \quad - d_1 d_2 x_i(t)^2 - 2d_1 d_6 x_i(t) - d_1 d_3. \end{cases}$$

The solutions of (18) are

$$\begin{cases} k_1 = (-\psi_2 - \sqrt{\psi_2^2 - \psi_1\psi_3})/\psi_1, \\ k_2 = (-\psi_2 + \sqrt{\psi_2^2 - \psi_1\psi_3})/\psi_1. \end{cases} \quad (19)$$

From (17) and (19) with $x(t) = 1$, the adjustment amplitude range $[r_{i+1}^l(t), r_{i+1}^h(t)]$ of $x_{i+1}(t)$ are obtained,

$$\begin{cases} r_{i+1}^l(t) = k_1 + x_i(t), \\ r_{i+1}^h(t) = k_2 + x_i(t). \end{cases} \quad (20)$$

Fig. 4 illustrates the hyperbolas and tangent lines.

For an abnormal point $X(t) = [x_1(t), x_2(t), \dots, x_n(t)]$, the adjustments of manipulated variables are summarized in the following algorithm.

Algorithm 3:

- Determine the manipulated variables to be adjusted by Algorithm 2 as $\{x_{m_1}, x_{m_2}, \dots, x_{m_s}\}$.
- Calculate the adjustment amplitude range $[r_{m_i}^l(t), r_{m_i}^h(t)]$ in (20) for $m_i = m_1, m_2, \dots, m_s$. The adjustment amplitude is the middle point of the range, i.e.,

$$x'_{m_i}(t) = \frac{r_{m_i}^l(t) + r_{m_i}^h(t)}{2}. \quad (21)$$

If x_{m+1} is the first to be adjusted, then its adjustment amplitude range is the same as $[y_{m+1}^l(t), y_{m+1}^h(t)]$ in (16), and the adjustment amplitude is

$$x'_{m+1}(t) = \frac{y_{m+1}^l(t) + y_{m+1}^h(t)}{2}.$$

The computation complexity of the proposed method is analyzed here. The proposed method is composed by the offline and online stages. At the offline stage of establishing

the NOZ, the distance from each data point to the cluster center needs to be calculated, and these distances must be sorted in the ascending order; thus, the computation complexity is $O(N\log(N))$. Taking Example 3 appeared later in Section V for instance, the offline stage takes about 1.49 sec at a personal computer with Intel i7 3.4GHz CPU and 16.0 GB memory, to establish the NOZ from 20000 data points of six process variables. In the online stage, dynamic alarm limits are respectively calculated in (7) and (21); the computation complexity is $O(1)$, which imposes no difficulties for online implementations.

V. EXAMPLES

This section presents numerical and industrial examples to illustrate the proposed method.

Example 1: Consider a multivariate system composed by three variables with a given hyper-ellipsoid model,

$$[X(t) - C_0]P_0[X(t) - C_0]^T \leq 1, \quad (22)$$

where

$$C_0 = [0.50 \quad 0.50 \quad 0.50],$$

and

$$P_0 = \begin{bmatrix} 13.39 & -6.01 & 9.88 \\ -6.01 & 9.93 & -5.99 \\ 9.88 & -5.99 & 13.12 \end{bmatrix}.$$

1000 historical normal data points are generated through (22) and are contaminated by three independent Gaussian white noises with zero mean and standard deviation 0.1.

First, Algorithm 1 is applied to the training data points for establishing the NOZ with the FAR $p = 0.05$. The estimated center vector is

$$C = [0.50 \quad 0.51 \quad 0.51],$$

and the estimated feature matrix is

$$P = \begin{bmatrix} 13.48 & -6.16 & 9.94 \\ -6.16 & 10.01 & -6.05 \\ 9.94 & -6.05 & 13.22 \end{bmatrix}.$$

The estimates are consistent with the real values.

Second, dynamic alarm limits in (7) are validated by comparing with the results obtained by projecting the NOZ into two-dimensional planes. Fig. 5(a) presents the NOZ surface $F([x_1(t), x_2(t), x_3(t)]) = 0$ in (4) and a new data point $X(t_0) = [0.7, 0.7, 0.4]$. In Fig. 5(b), the ellipse is the projection of $F([x_1(t), x_2(t), 0.4]) = 0$ on the x_1x_2 plane, and the red-asterisk point $[0.7, 0.7]$ is the projection of $X(t_0)$ on the x_1x_2 plane. The variational range of x_1 is $[0.46, 0.87]$ for $x_2 = 0.7$, and the range of x_2 is $[0.28, 0.84]$ for $x_1 = 0.7$. In Fig. 5(c), the ellipse is the projection of $F([0.7, x_2(t), x_3(t)]) = 0$ on the x_2x_3 plane, and the red-asterisk point $[0.7, 0.4]$ is the projection of $X(t_0)$ on the x_2x_3 plane. The variational range of x_2 is $[0.28, 0.84]$ for $x_3 = 0.4$, and the range of x_3 is $[0.23, 0.65]$ for $x_2 = 0.7$. Dynamic alarm limits of x_1 , x_2 and x_3 are calculated from (7) as $[0.46, 0.87]$, $[0.28, 0.84]$ and $[0.23, 0.65]$, respectively. Thus,

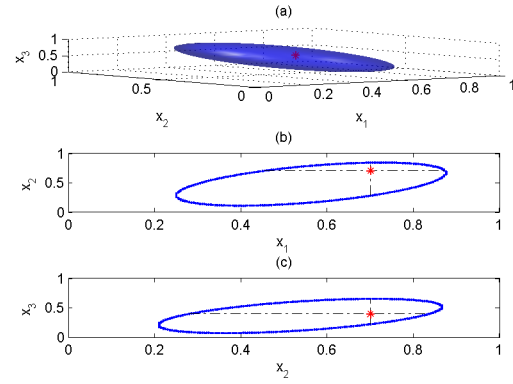


Fig. 5: A three-dimensional NOZ and a normal data point $X(t_0)$ for designing dynamic alarm limits in Example 1.

they are consistent with the results obtained by the projection method. By comparing $X(t_0)$ with the dynamic alarm limits, it is transparent to tell that $X(t_0)$ is inside the NOZ.

Third, let the variable x_3 be increased to 0.8 so that $X(t_1) = [0.7, 0.7, 0.8]$. Dynamic alarm limits of x_1 , x_2 and x_3 are obtained as $[0.18, 0.64]$, $[0.64, 0.77]$ and $[0.21, 0.71]$, respectively. Clearly, $x_1(t_1) = 0.7$ and $x_3(t_1) = 0.8$ are larger than their dynamic high alarm limits $h_1(t_1) = 0.64$ and $h_3(t_1) = 0.71$; thus, $X(t_1)$ is an abnormal data point. As presented in Fig. 6(a), $X(t_1)$ is indeed outside the NOZ.

Fourth, adjustments of manipulated variables are investigated. Suppose that x_1 can not be manipulated directly, while x_2 and x_3 are manipulated variables. In Fig. 6(b), the blue dotted polyline represents the abnormal data point $X(t_1)$, the red solid hyperbolas stand for the reduced NOZ composed of x_2 and x_3 from the model in (22) by fixing $x_1 = 0.7$, and the black dot-dash lines are tangent lines of the hyperbolas. Owing to $x_3(t_1) \notin [y_3^l(t_1), y_3^h(t_1)]$, x_3 is determined as the manipulated variable to be adjusted. The adjustment range of x_3 is calculated from (20) as $[0.23, 0.65]$, and the adjustment amplitude of x_3 is obtained from (21) as $x_3'(t_1) = 0.44$. Dynamic alarm limits of the point $X'(t_1) = [0.7, 0.7, 0.44]$ are $[0.42, 0.85]$, $[0.31, 0.85]$ and $[0.23, 0.65]$. Because all the three variables for $X'(t_1)$ are inside their own dynamic alarm limits, $X'(t_1)$ is a normal data point.

Finally, 100 Monte Carlo simulations are performed to validate the performance of the proposed method. In each simulation, 1000 new data points are generated by the same way as the historical normal data points. The sample mean and standard deviation of FARs from 100 simulations are calculated as 0.0329 and 0.0059. It is consistent with the performance specification that FAR is no larger than 0.05 in establishing the NOZ in the first step.

Example 2: This industrial example is used to compare the proposed method with the static alarm limits and the PCA-based method. In a large-scale thermal power generation unit, the inlet flow rate x_1 of a feedwater pump is configured with a high alarm limit 560 t/h. Fig. 7 presents two-hour historical data samples of x_1 and x_2 where the sampling period is 1 second. Clearly, 388 data samples of $x_1(t)$ for

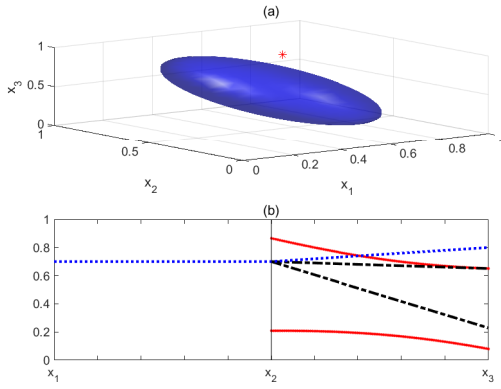


Fig. 6: A three-dimensional NOZ and an abnormal data point $X(t_1)$ for adjusting manipulated variables in Example 1.

$t \in [3301, 3363] \cup [3900, 4224]$ are larger than 560 t/h, leading to alarm occurrences. However, these alarms are not associated with any reported abnormalities. The increments of x_1 are actually induced by power plant operator's demand in normal operations. Hence, all the 388 alarms are indeed false alarms. A process knowledge provided by operators is: if the feedwater pump runs in the normal condition, then the electrical current x_2 of a booster pump should change in a synchronized manner with x_1 . As shown in Fig. 7, the relation between x_1 and x_2 is unchanged during the alarm occurrences. The static alarm limit of x_1 is isolated from the related variable x_2 , which results in false alarms.

The proposed method and the PCA-based method are applied to the bivariate system composed by x_1 and x_2 . The results are given in Figs. 8 and 9, where only the part for $t \in [3800, 4300]$ (between two vertical dot lines in Fig. 7) are presented for a better visualization. Both methods take a hyper-ellipsoid model as the NOZ. For a fair comparison, the same ellipsoid model in Fig. 9(a) is used for the two methods. Thus, the two methods are expected to have the same performance in terms of FAR. This is validated in Figs. 8 and 9(b), where the two methods generate two alarms at $t = 3994$ and $t = 4124$ due to measurement noises, much less than the 388 false alarms by using static alarm limits. However, the two methods are very different in the way of monitoring whether a data point is inside the NOZ. For the proposed method, the value of each process variable is compared with its dynamic alarm limits, i.e., $x_1(t)$ is compared with $l_1(t)$ and $h_1(t)$, and $x_2(t)$ is compared with $l_2(t)$ and $h_2(t)$. The dynamic alarm limits have the same physical meanings as the corresponding process variables. For the PCA-based method, the T^2 statistic is compared with its alarm limit associated with a confidence level 0.99. Here the T^2 statistic is obtained as $T^2 = 73.78x_1^2 - 146.56x_1x_2 + 73.78x_2^2$. As a result, the alarm limit associated with the T^2 statistic is difficult for power plant operators to accept.

Example 3: This is an industrial example from a large-scale thermal power generation unit. The distributed control system therein provides real-time data samples of 24079 process variables. The operating status of the power generation unit

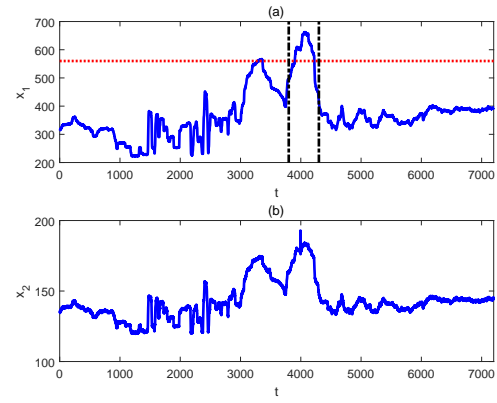


Fig. 7: Time sequence plots of process variables in Example 2: (a) the inlet flow rate x_1 of a feedwater pump, (b) the electrical current x_2 of a booster pump.

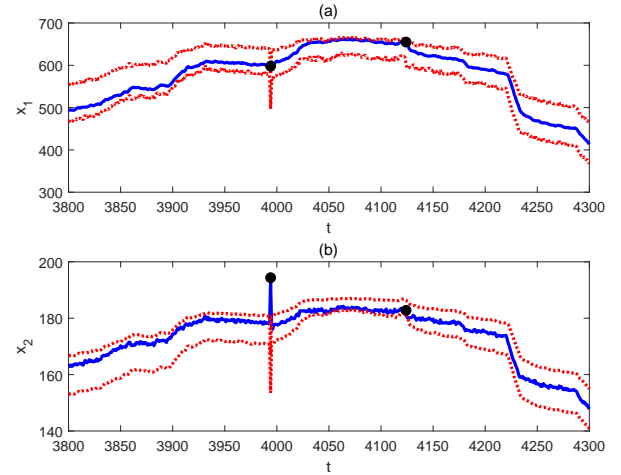


Fig. 8: (a) The time sequence plot of x_1 (blue solid), its dynamic alarm limits (red dot), and false alarms (black circle). (b) The time sequence plot of x_2 (blue solid), its dynamic alarm limits (red dot), and false alarms (black circle).

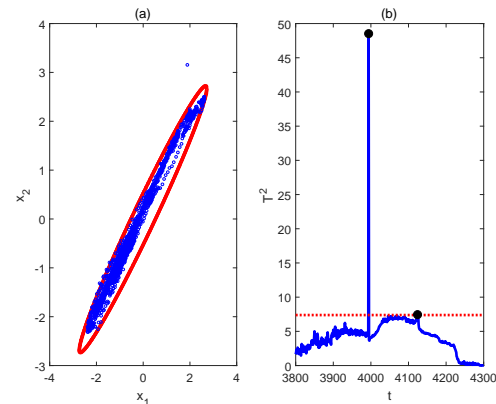


Fig. 9: (a) The scatter plot (blue dot) of x_1 and x_2 , and the estimated ellipsoid model (red dash). (b) The time sequence plot of T^2 statistic (blue solid), its alarm limit (red dot) and false alarms (black circle).

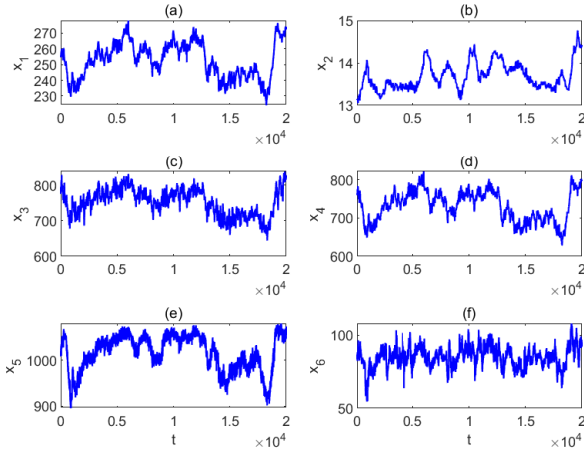


Fig. 10: Time sequence plots of (a) generated electrical power x_1 in MW, (b) steam pressure x_2 in MPa, (c) steam flow rate x_3 in t/h, water flow rate x_4 in t/h, (d) air flow rate x_5 in t/h and (e) coal flow rate x_6 in t/h in 5.5 hours of normal operations with the sampling period 1 second in Example 3.

as a whole can be described by six process variables, namely, the generated electrical power x_1 in the unit of MW, steam pressure x_2 in MPa, steam flow rate x_3 in t/h, water flow rate x_4 in t/h, air flow rate x_5 in t/h and coal flow rate x_6 in t/h. Fig. 10 presents time sequence plots of the six variables for data points in 5.5 hours of normal operations. The sampling period is 1 second.

The six process variables are selected on the basis of the following process knowledge. The main objective of the power generation unit is to convert the heat energy of combustion by burning coal powders mixed with air into the thermal energy of high-pressure high-temperature steam, and to generate the electricity via an electrical generator. When the generated electricity is changed according to the demand from power grids, the flow rates of the coal, air, water and steam have to be changed accordingly. The steam temperature usually varies within a small range, so that it is not considered. Based on related process knowledge, x_1 and x_2 are the ones to be controlled, while the other four variables are manipulated variables. The order of manipulating the four variables is determined as follows: the steam flow rate x_3 is the easiest to adjust by changing control valves and has a direct effect on the generated electrical power x_1 and steam pressure x_2 ; adjusting the water flow rate x_4 should be done right after changing x_3 to keep a balance between water and steam flow rates; the air flow rate x_5 is usually changed before adjusting the coal flow rate x_6 for safety concerns.

The proposed method is implemented as follows. First, the hyper-ellipsoid model in (3) is built by Algorithm 1 with the FAR $p = 0.05$. The performance of the proposed method is validated by a new set of data points. Fig. 12 presents the time sequence plots of 1000 new data points (blue dot lines) with their dynamic alarm limits (red solid lines). All the new data points are known to be in the normal condition.

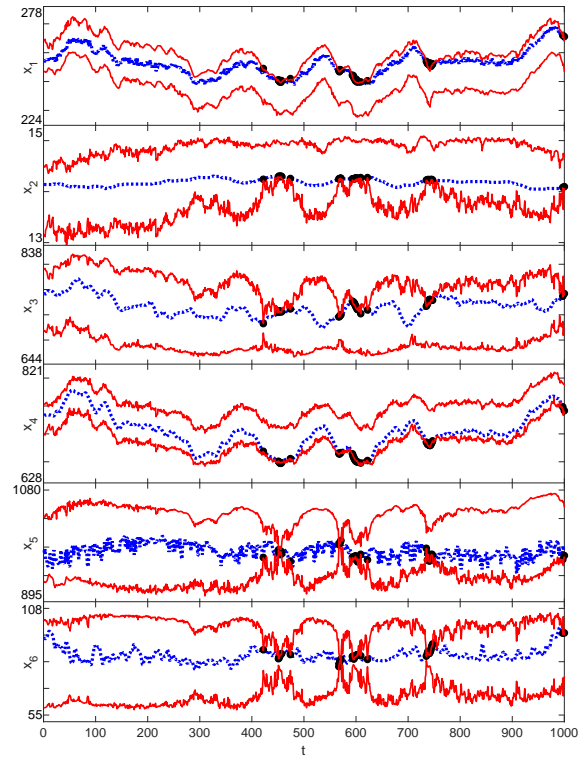


Fig. 11: Time sequence plots of 1000 new data samples of six process variables (blue dot) and their dynamic alarm limits (red solid) in the normal condition: (a) generated electrical power x_1 in MW, (b) steam pressure x_2 in MPa, (c) steam flow rate x_3 in t/h, water flow rate x_4 in t/h, (d) air flow rate x_5 in t/h and (e) coal flow rate x_6 in t/h (red curves) in Example 3.

In Fig. 12, black circles represent the new data points causing false alarms. There are 14 false alarms in total, so that the FAR is equal to 0.014. It is consistent with the performance specification that the FAR is no larger than 0.05.

Second, dynamic alarm limits are calculated. Fig. 12 presents the time sequence plots of 300 new data points (blue dot lines) with their dynamic alarm limits (red solid lines). In Fig. 12, black circles represent the new data points causing alarms. From the time instant $t = 271$ to the current time instant, there are 30 consecutive points of x_1 , x_2 , x_4 , x_5 and x_6 that are outside of their dynamic alarm limits. An abnormal condition is found to be present: the air flow rate x_5 and the coal flow rate x_6 are erroneously reduced too much, and the water flow rate x_4 is not in balance with x_5 and x_6 ; as a result, the coal burning process in the power generation unit yields less energies than the required, so that the steam pressure x_2 and the generated electrical power x_1 are lower than their normal values. The designed alarm system successfully detects such an abnormal condition.

Finally, the manipulated variables to be adjusted and their adjustment amplitudes are determined for alarm removals. Let us consider the current data point $X(t_0) = [256.47, 13.48, 770.44, 773.10, 959.89, 65.26]$ being associated with alarms. The red hyperbolas in Fig. 13 represents

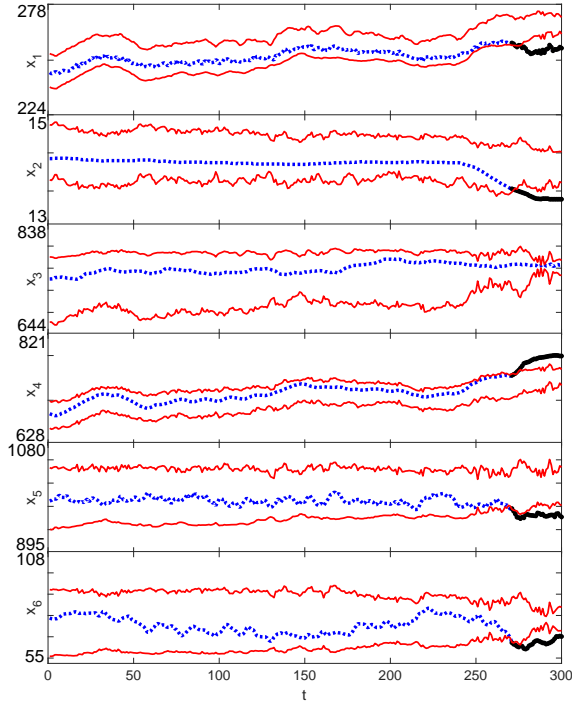


Fig. 12: Time sequence plots of 300 new data samples of six process variables (blue dot) and their dynamic alarm limits (red solid) with an abnormal condition: generated electrical power x_1 in MW, steam pressure x_2 in MPa, steam flow rate x_3 in t/h, water flow rate x_4 in t/h, air flow rate x_5 in t/h and coal flow rate x_6 in t/h (red curves) in Example 3 (from top to bottom).

the reduced NOZ by fixing $x_1 = 256.47$ and $x_2 = 13.48$; x_4 , x_5 and x_6 are determined by Algorithm 2 as manipulated variables to be adjusted. Applying (20) to x_4 , x_5 and x_6 in turns yields the adjustment ranges of x_4 , x_5 and x_6 as $[734.29, 762.09]$, $[973.79, 1087.50]$ and $[68.05, 106.50]$, and the adjustment amplitudes $x'_4(t_0) = 748.19$, $x'_5(t_0) = 1030.65$ and $x'_6(t_0) = 87.28$, respectively. When x_4 , x_5 and x_6 are respectively adjusted to 748.19, 1030.65 and 87.28, the corresponding data point $X'(t_0) = [256.47, 13.48, 770.44, 748.19, 1030.65, 87.28]$ is returned to the NOZ. This is validated by a fact that each value of $X'(t_0)$ is indeed inside the corresponding dynamic alarm limits $[252.70, 260.11]$, $[12.89, 14.24]$, $[702.79, 828.00]$, $[734.72, 761.73]$, $[974.70, 1087.28]$ and $[68.20, 106.30]$.

VI. CONCLUSION

This paper proposed a method to design dynamic alarm limits and adjust manipulated variables for multivariate systems. The hyper-ellipsoid model in (3) was built via Algorithm 1 by clustering historical normal data points as the NOZ. Dynamic alarm limits in (7) were developed based on the NOZ. A comparison of current values of process variables and their dynamic alarm limits could tell whether the multivariate system was in the normal condition and how

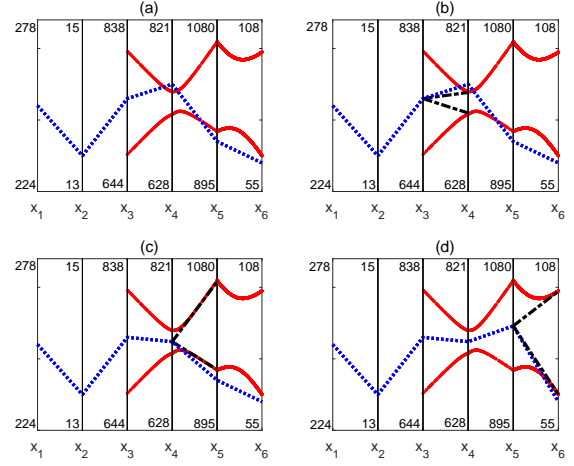


Fig. 13: The data point $X(t_0)$ (blue dot), the hyperbolas (red solid) and tangent lines (black dot-dash) of the reduced NOZ in the parallel coordinate in Example 3.

far it was deviating away from the NOZ. If alarms were occurring, manipulated variables to be adjusted were chosen via Algorithm 2 as the ones outside of the reduced NOZ, which was developed by fixing non-manipulated variables to their current values. Adjustment amplitudes of manipulated variables were calculated by Algorithm 3 from the reduced NOZ represented in the parallel coordinate. Numerical and industrial examples validated the proposed method.

The proposed method has two limitations to be dealt with in the future work. First, it is assumed in Section II that all new data points in the normal condition are distributed in a consistent way as the historical normal data points. Such an assumption may not be valid for time-varying and/or nonlinear processes. As one future work, a mechanism is needed to monitor whether new data points in the normal condition are distributed in a consistent way as the historical normal data points. If the answer is negative, then the NOZ has to be updated in some manner. Second, a hyper-ellipsoid model is used here to describe the NOZ formulated by historical normal data points. For some cases, the NOZ may be better described by other types of models such as a convex-hull model or multiple hyper-ellipsoid models. Hence, another future work is to devise a modeling approach being able to automatically choose model types and the number of sub-models in order to fit with historical normal data points.

APPENDIX

This appendix proves that the order of process variables has no effects on the dynamic alarm limits and adjustment amplitudes of manipulated variables. First, assume that the n process variables are sorted as $X(t) = [U_1(t), x_i(t), U_2(t), x_j(t), U_3(t)]$, where $U_1(t) = [x_1(t), x_2(t), \dots, x_{i-1}(t)]$, $U_2(t) = [x_{i+1}(t), x_{i+2}(t), \dots, x_{j-1}(t)]$, $U_3(t) = [x_{j+1}(t), x_{j+2}(t), \dots, x_n(t)]$. Then, the hyper-ellipsoid model in (4)

can be written as

$$\begin{bmatrix} U_1(t) \\ x_i(t) \\ U_2(t) \\ x_j(t) \\ U_3(t) \\ 1 \end{bmatrix}^T \begin{bmatrix} E_{11} & E_{12} & E_{13} & E_{14} & E_{15} & E_{16} \\ E_{11} & E_{22} & E_{23} & E_{24} & E_{25} & E_{26} \\ E_{13} & E_{23} & E_{33} & E_{34} & E_{35} & E_{36} \\ E_{14} & E_{24} & E_{34} & E_{44} & E_{45} & E_{46} \\ E_{15} & E_{25} & E_{35} & E_{45} & E_{55} & E_{56} \\ E_{16} & E_{26} & E_{36} & E_{46} & E_{56} & E_{66} \end{bmatrix} \begin{bmatrix} U_1(t) \\ x_i(t) \\ U_2(t) \\ x_j(t) \\ U_3(t) \\ 1 \end{bmatrix} = 0, \quad (23)$$

where $\bar{E}_{11} = \bar{P}(1:i-1, 1:i-1)$, $\bar{E}_{12} = \bar{P}(1:i-1, i)$, $\bar{E}_{13} = \bar{P}(1:i-1, i+1:j-1)$, $\bar{E}_{14} = \bar{P}(1:i-1, j)$, $\bar{E}_{15} = \bar{P}(1:i-1, j+1:n)$, $\bar{E}_{16} = \bar{P}(1:i-1, n+1)$, $\bar{E}_{22} = \bar{P}(i, i)$, $\bar{E}_{23} = \bar{P}(i, i+1:j-1)$, $\bar{E}_{24} = \bar{P}(i, j)$, $\bar{E}_{25} = \bar{P}(i, j+1:n)$, $\bar{E}_{26} = \bar{P}(i, n+1)$, $\bar{E}_{33} = \bar{P}(i+1:j-1, i+1:j-1)$, $\bar{E}_{34} = \bar{P}(i+1:j-1, j)$, $\bar{E}_{35} = \bar{P}(i+1:j-1, j+1:n)$, $\bar{E}_{36} = \bar{P}(i+1:j-1, n+1)$, $\bar{E}_{44} = \bar{P}(j, j)$, $\bar{E}_{45} = \bar{P}(j, j+1:n)$, $\bar{E}_{46} = \bar{P}(j, n+1)$, $\bar{E}_{55} = \bar{P}(j+1:n, j+1:n)$, $\bar{E}_{56} = \bar{P}(j+1:n, n+1)$, and $\bar{E}_{66} = \bar{P}(n+1, n+1)$. Eq. (23) can be expanded as

$$\begin{aligned} & E_{11}U_1^2(t) + E_{22}x_i(t)^2 + E_{33}U_2^2(t) + E_{44}x_j(t)^2 + E_{55}U_3^2(t) \\ & + 2E_{12}U_1(t)x_i(t) + 2E_{13}U_1(t)U_2(t) + 2E_{14}U_1(t)x_j(t) \\ & + 2E_{15}U_1(t)U_3(t) + 2E_{23}x_i(t)U_2(t) + 2E_{24}x_i(t)x_j(t) \\ & + 2E_{25}x_i(t)U_3(t) + 2E_{34}U_2(t)x_j(t) + 2E_{35}U_2(t)U_3(t) \\ & + 2E_{45}x_j(t)U_3(t) + 2E_{16}U_1(t) + 2E_{26}x_i(t) + \\ & 2E_{36}U_2(t) + 2E_{46}x_j(t) + 2E_{56}U_3(t) + E_{66} = 0. \end{aligned} \quad (24)$$

If the order of the process variables is changed to $X(t)' = [U_1(t), x_j(t), U_2(t), x_i(t), U_3(t)]$, then the hyper-ellipsoid model in (4) becomes

$$\begin{bmatrix} U_1(t) \\ x_j(t) \\ U_2(t) \\ x_i(t) \\ U_3(t) \\ 1 \end{bmatrix}^T \begin{bmatrix} E_{11} & E_{14} & E_{13} & E_{12} & E_{15} & E_{16} \\ E_{14} & E_{44} & E_{34} & E_{24} & E_{45} & E_{46} \\ E_{13} & E_{34} & E_{33} & E_{23} & E_{35} & E_{36} \\ E_{11} & E_{24} & E_{23} & E_{22} & E_{25} & E_{26} \\ E_{15} & E_{45} & E_{35} & E_{25} & E_{55} & E_{56} \\ E_{16} & E_{46} & E_{36} & E_{26} & E_{56} & E_{66} \end{bmatrix} \begin{bmatrix} U_1(t) \\ x_j(t) \\ U_2(t) \\ x_i(t) \\ U_3(t) \\ 1 \end{bmatrix} = 0. \quad (25)$$

Eq. (25) leads to

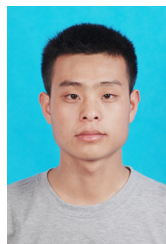
$$\begin{aligned} & E_{11}U_1^2(t) + E_{22}x_i(t)^2 + E_{33}U_2^2(t) + E_{44}x_j(t)^2 + E_{55}U_3^2(t) \\ & + 2E_{12}U_1(t)x_i(t) + 2E_{13}U_1(t)U_2(t) + 2E_{14}U_1(t)x_j(t) \\ & + 2E_{15}U_1(t)U_3(t) + 2E_{23}x_i(t)U_2(t) + 2E_{24}x_i(t)x_j(t) \\ & + 2E_{25}x_i(t)U_3(t) + 2E_{34}U_2(t)x_j(t) + 2E_{35}U_2(t)U_3(t) \\ & + 2E_{45}x_j(t)U_3(t) + 2E_{16}U_1(t) + 2E_{26}x_i(t) + \\ & 2E_{36}U_2(t) + 2E_{46}x_j(t) + 2E_{56}U_3(t) + E_{66} = 0. \end{aligned} \quad (26)$$

Obviously, eq. (26) is the same as eq. (24). Dynamic alarm limits in (7) and adjustment amplitudes in (21) are derived from (24) or (26), so that the obtained results remain the same.

REFERENCES

- [1] ISA, *ANSI/ISA-18.2: Management of Alarm Systems for the Process Industries*. Durham County: International Society of Automation, Jun. 2009.
- [2] EEMUA, *EEMUA-191: Alarm Systems - A Guide to Design, Management and Procurement*. London: Engineering Equipment and Materials Users Association, Sep. 2013.
- [3] D. S. Kirschen and B. F. Wollenberg, "Intelligent alarm processing in power systems," *Proc. IEEE*, vol. 80, no. 5, pp. 663–672, May. 1992.
- [4] D. H. Rothenberg, *Alarm Management for Process Control*. New York: Momentum Press, Aug. 2009.
- [5] B. Hollifield and E. Habibi, *The Alarm Management Handbook*. Fremont: PAS Inc., Aug. 2010.
- [6] J. Wang, F. Yang, T. Chen, and S. L. Shah, "An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 1045–1061, Apr. 2016.
- [7] Y. Yu, J. Wang, and Z. Yang, "Design of alarm trip-points for univariate analog process variables based on alarm probability plots," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 8, pp. 6496–6505, Aug. 2017.
- [8] S. W. Cheon, S. H. Chang, and H. Y. Chung, "Development strategies of an expert system for multiple alarm processing and diagnosis in nuclear power plants," *IEEE Nuclear and Plasma Sciences Society*, vol. 40, no. 1, pp. 21–30, Feb. 1993.
- [9] D. Fischer, B. Szabados, and W. F. S. Poehlman, "Using a bayes classifier to optimize alarm generation to electric power generator stator overheating," *IEEE Transactions on Instrumentation and Measurement*, vol. 52, no. 3, pp. 703–709, Jun. 2003.
- [10] F. Yang, S. L. Shah, and D. Xiao, "Correlation analysis of alarm data and alarm limit design for industrial processes," in *American Control Conference (ACC)*, pp. 5850–5855, Jun. 2010.
- [11] A. Gupta, A. Giridhar, V. Venkatasubramanian, and G. V. Reklaitis, "Intelligent alarm management applied to continuous pharmaceutical tablet manufacturing: An integrated approach," *Industrial & Engineering Chemistry Research*, vol. 52, no. 35, pp. 12 357–12 368, Mar. 2013.
- [12] F. Alrowaie, R. B. Gopaluni, and K. E. Kwok, "Alarm design for nonlinear stochastic systems," in *11th World Congress on Intelligent Control and Automation (WCI-CA)*, pp. 473–479, Jun. 2014.
- [13] J. Zhu, Y. Shu, J. Zhao, and F. Yang, "A dynamic alarm management strategy for chemical process transitions," *Journal of Loss Prevention in the Process industries*, vol. 30, pp. 207–218, Jul. 2014.
- [14] L. Han, H. Gao, Y. Xu, and Q. Zhu, "Combining fap, map and correlation analysis for multivariate alarm thresholds optimization in industrial process," *Journal of Loss Prevention in the Process Industries*, vol. 40, no. 10, pp. 471–478, Mar. 2016.
- [15] J. F. MacGregor and T. Kourti, "Statistical process control of multivariate processes," *Control Engineering Practice*, vol. 3, no. 3, pp. 403–414, Mar. 1995.
- [16] L. H. Chiang, E. L. Russell, and R. D. Braatz, *Fault Detection and Diagnosis in Industrial Systems*. London: Springer, Dec. 2001.
- [17] S. J. Qin, "Statistical process monitoring: basics and beyond," *Journal of Chemometrics*, vol. 17, no. 8, pp. 480–502, Sep. 2003.
- [18] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri,

- and K. Yin, "A review of process fault detection and diagnosis: Part iii: Process history based methods," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 327–346, Mar. 2003.
- [19] S. J. Qin, "Survey on data-driven industrial process monitoring and diagnosis," *Annual Reviews in Control*, vol. 36, no. 2, pp. 220–234, Dec. 2012.
- [20] J. MacGregor and A. Cinar, "Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods," *Computers & Chemical Engineering*, vol. 47, no. 20, pp. 111–120, Dec. 2012.
- [21] U. Kruger and L. Xie, *Statistical Monitoring of Complex Multivariate Processes: With Applications in Industrial Process Control*. Chichester: John Wiley & Sons, Oct. 2012.
- [22] Z. Ge, Z. Song, and F. Gao, "Review of recent research on data-based process monitoring," *Industrial & Engineering Chemistry Research*, vol. 52, no. 10, pp. 3543–3562, Feb. 2013.
- [23] S. Yin, S. X. Ding, X. Xie, and H. Luo, "A review on basic data-driven approaches for industrial process monitoring," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 11, pp. 6418–6428, Jan. 2014.
- [24] Z. Ge, "Process data analytics via probabilistic latent variable models: a tutorial review," *Industrial & Engineering Chemistry Research*, vol. 57, no. 38, pp. 12 646–12 661, Aug. 2018.
- [25] Q. P. He and J. Wang, "Statistical process monitoring as a big data analytics tool for smart manufacturing," *Journal of Process Control*, vol. 67, pp. 35–43, Jul. 2018.
- [26] R. Brooks, R. Thorpe, and J. Wilson, "A new method for defining and managing process alarms and for correcting process operation when an alarm occurs," *Journal of Hazardous Materials*, vol. 115, no. 1, pp. 169–174, Nov. 2004.
- [27] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Computers & Chemical Engineering*, vol. 17, no. 3, pp. 245–255, Mar. 1993.
- [28] J. F. MacGregor, H. Yu, S. G. Munoz, and J. Flores-Cerrillo, "Data-based latent variable methods for process analysis, monitoring and control," *Computers & Chemical Engineering*, vol. 29, no. 6, pp. 1217–1223, May. 2005.
- [29] S. Bezergianni and A. Kalogianni, "Application of principal component analysis for monitoring and disturbance detection of a hydrotreating process," *Industrial & Engineering Chemistry Research*, vol. 47, no. 18, pp. 6972–6982, Aug. 2008.
- [30] L. G. Bergh and D. Gomez, "On line monitoring and diagnosis of the operation of a hybrid cstr by using pca models," *Computer Aided Chemical Engineering*, vol. 27, no. 9, pp. 1203–1208, 2009.
- [31] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, Jun. 2010.
- [32] M. Maestri, A. Farall, P. Groisman, M. Cassanello, and G. Horowitz, "A robust clustering method for detection of abnormal situations in a process with multiple steady-state operation modes," *Computers & Chemical Engineering*, vol. 34, no. 2, pp. 223–231, Feb. 2010.
- [33] A. Inselberg, *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*. New York: Springer, Sep. 2009.



Yan Yu received the B.E. Degree in mechanical design, manufacturing and automation from Anhui University, Anhui, China, in 2015. He is currently working toward the Ph.D. degree at Peking University, Beijing, China. His research interest is the advanced alarm system management.



Jiandong Wang is presently a full Professor of College of Electrical Engineering and Automation at the Shandong University of Science and Technology, Qingdao, Shandong Province, China. He received the B.E. degree in automatic control from Beijing University of Chemical Technology, Beijing, China, in 1997, and the M.Sc and Ph.D. degrees in Electrical and Computer Engineering from the University of Alberta, Canada, in 2003 and 2007, respectively. From 1997 to 2001, he was a Control Engineer with the Beijing Tsinghua Energy Simulation Company, Beijing, China. From February 2006 to August 2006, he was a Visiting Scholar at the Department of System Design Engineering at the Keio University, Japan. From December 2006 to October 2016, he was an assistant/associate/full Professor with the College of Engineering, Peking University, China.

His research interests include industrial alarm systems, process monitoring and management, system identification and their applications to industrial problems. He has served as an Associate Editor/Guest Editor for Systems and Control Letters, and Control Engineering Practice, and is currently serving as an Associate Editor for Journal of the Franklin Institute.



Zijian Ouyang is currently working as an engineer in the Management Committee of Future Technology City, Hangzhou, Zhejiang. He received the B.E. Degree in automatic control from Wuhan University, Hubei, China, in 2010, and M.Sc. Degree in control engineering from Peking University, Beijing, China, in 2013. His research interest is the advanced alarm system management.