Computers and Chemical Engineering 163 (2022) 107853

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

A novel fault detection and diagnosis approach based on orthogonal autoencoders

Davide Cacciarelli[a,b,*], Murat Kulahci[a,c]

[a] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Kgs. Lyngby, Denmark
[b] Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norway
[c] Department of Business Administration, Technology and Social Sciences, Luleå University of Technology, Luleå, Sweden

ARTICLE INFO

Article history:
Received 16 November 2021
Revised 31 March 2022
Accepted 22 May 2022
Available online 23 May 2022

Keywords:
Statistical process control
Unsupervised learning
Autoencoder
Fault detection and diagnosis
Deep Learning
Tennessee Eastman process

ABSTRACT

In recent years, there have been studies focusing on the use of different types of autoencoders (AEs) for monitoring complex nonlinear data coming from industrial and chemical processes. However, in many cases the focus was placed on detection. As a result, practitioners are encountering problems in trying to interpret such complex models and obtaining candidate variables for root cause analysis once an alarm is raised. This paper proposes a novel statistical process control (SPC) framework based on orthogonal autoencoders (OAEs). OAEs regularize the loss function to ensure no correlation among the features of the latent variables. This is extremely beneficial in SPC tasks, as it allows for the invertibility of the covariance matrix when computing the Hotelling T^2 statistic, significantly improving detection and diagnosis performance when the process variables are highly correlated. To support the fault diagnosis and identification analysis, we propose an adaptation of the integrated gradients (IG) method. Numerical simulations and the benchmark Tennessee Eastman Process are used to evaluate the performance of the proposed approach by comparing it to traditional approaches as principal component analysis (PCA) and kernel PCA (KPCA). In the analysis, we explore how the information useful for fault detection and diagnosis is stored in the intermediate layers of the encoder network. We also investigate how the correlation structure of the data affects the detection and diagnosis of faulty variables. The results show how the combination of OAEs and IG represents a compelling and ready-to-use solution, offering improved detection and diagnosis performances over the traditional methods.

1. Introduction

The increasing availability of large unlabeled datasets is reinforcing the interest of researchers and practitioners toward unsupervised learning (Locatello et al., 2019). In industrial and chemical processes, unlabeled data is more ubiquitous particularly for high-volume production, which often renders measuring the quality characteristics of every product unfeasible. This is certainly reflected in data analytics applications such as statistical process control (SPC) where the aim is to monitor a process over time and verify that it remains in a state of statistical control (MacGregor and Kourti, 1995). With the proliferation of automated data collection schemes and advances in sensorics, it is nowadays a common occurrence that multiple process variables are collected and used in summary statistics to monitor the performance of the processes. The most extensively adopted summary statistic for that purpose is the Hotelling's T^2 statistic (Hotelling, 1947), which requires computing the p-dimensional vector of sample means and the sample $p \times p$ covariance matrix, where p represents the number of process variables to be monitored.

In multivariate SPC applications, during the initial phase (Phase I), data is collected from the process under normal operating conditions, and mean vector and covariance matrix are estimated. These estimates are then used to calculate the T^2 statistic

$$T^2 = (x - \bar{x})^T S^{-1} (x - \bar{x}) \tag{1}$$

where $\bar{x}$ is the estimate of the mean vector and S is the estimate of the covariance matrix of p variables. It should be noted that for the $p \times 1$ observation vector x, T^2 statistic is a scalar irrespective of the number of variables, p. A T^2 control chart is then constructed using separate control limits for Phase I data and the prospective data collected in real-time, also called Phase II. These limits are calcu-

* Corresponding author at: Department of Applied Mathematics and Computer Science, Technical University of Denmark, Kgs. Lyngby, Denmark.
E-mail address: dcac@dtu.dk (D. Cacciarelli).

https://doi.org/10.1016/j.compchemeng.2022.107853

lated assuming a time-independent p-variate joint normal distribution for the data and are given respectively as

$$UCL_I = \frac{(m-1)^2}{m}\beta_{\alpha,p/2,(m-p-1)/2} \tag{2}$$

$$UCL_{II} = \frac{p(m+1)(m-1)}{m^2 - mp}F_{\alpha,p,m-p} \tag{3}$$

where m is the number of observations collected in Phase I, $\beta_{\alpha,p/2,(m-p-1)/2}$ and $F_{\alpha,p,m-p}$ the upper α-percentiles of the Beta and F-distributions with the associated degrees of freedom (Montgomery, 2017).

The most commonly used measure to assess the performance of a control chart is the average run length (ARL) (Box et al., 2003). The ARL for an in-control process, or ARL_0, represents the expected number of observations plotted before a false alarm is declared. On the other hand, ARL_1, the average run length for an out-of-control process corresponds to the expected number of observations plotted before the control chart detects the presence of a fault. ARL is a highly relevant metric in industrial contexts since it expresses the detection delay, which is crucial to understand whether the proposed method allows for a timely intervention. Another measure that is often used to assess the anomaly detection performance is the fault detection rate (FDR), along with the false alarm rate (FAR). FDR and FAR indicate, respectively, the proportion of correctly identified faults and the ratio of false alarms raised over a given time frame. When an out-of-control situation is detected, it is possible to inspect the signal by decomposing the T^2 statistic into components that express the relative contribution of each process variable to that statistic. Variables that report a high contribution can be investigated further by operators and engineers to determine the root cause of the alarm. A possible way of investigating the contribution of the ith variable to T^2 statistic is through the difference between the T^2 statistic and a second T^2 statistic, $T^2_{(i)}$,obtained by excluding the ith variable (Runger et al., 1996)

$$d_i = T^2 - T^2_{(i)} \tag{4}$$

The Hotelling T^2 control chart is extremely useful for detecting and identifying faults. A recent work (Ueda and Souza, 2022) shows how the T^2 decomposition technique can be effectively used in manufacturing processes to identify the sources of process variability. It is also widely used for the monitoring of chemical and pharmaceutical processes (Liu et al., 2017; Moreira et al., 2021; Silva et al., 2017). The main disadvantage of the Hotelling T^2 control chart is that it is not as effective when the variables of interest are highly correlated, which renders the sample covariance matrix S difficult to invert. Latent structure methods have been traditionally proposed to overcome this issue (MacGregor et al., 2005). An overview of these methods is offered in the upcoming section. Among the recent advances concerning the use of T^2 in production settings, in the latest years some authors have been focusing on the adaptation of the Hotelling T^2 charts for small production runs through the use of variable sample sizes (Chong et al., 2019); other researchers studied the impact of measurement errors on the detection performance in Phase II (Sabahno et al., 2018).

2. Preliminaries

Among the latent structure methods used in SPC, principal component analysis (PCA) is the most commonly employed unsupervised approach. Using PCA, the dimensionality of the dataset is reduced by creating new uncorrelated features that gradually maximize the variance (Jolliffe and Cadima, 2016). The original $n \times p$ data matrix X, where n is the number of observations, can be modeled as

$$X = Z_k C_k^{\mathrm{T}} + E \tag{5}$$

where $C_k \in \mathbb{R}^{p\times k}$ is the loading matrix, $Z_k \in \mathbb{R}^{n\times k}$ is the principal component matrix, k corresponds to the number of principal components (PCs) retained in the PCA model, and $E \in \mathbb{R}^{n\times p}$ is the residual matrix representing the "leftover" PCs (Vanhatalo et al., 2017). The process is then monitored using two control charts, a T^2 control chart for the k retained PCs, whose statistic for the ith observation is computed as

$$T_i^2 = \left(z_{k,i} - \bar{z}_k\right)^T S^{-1}\left(z_{k,i} - \bar{z}_k\right) \tag{6}$$

and a so-called Q or SPE chart for the error, whose ith statistic is given by

$$Q_i = e_i e_i^T \tag{7}$$

For the T^2 chart, the UCLs are obtained from Eqs. (2) and (3) while for the SPE chart, several limits have been proposed in the literature (Box, 1954; Frumosu and Kulahci, 2019; Jackson and Mudholkar, 1979).

Contributions of each variable on the T^2 statistics are obtained using

$$T^2_{Contr,i} = (x_i - \bar{x})^T C_k \Lambda_k^{1/2} C_k^{\mathrm{T}} \tag{8}$$

where Λ_k is the diagonal matrix of first k eigenvalues in descending order. Contributions for the Q chart are given by

$$Q_{Contr,i} = (x_i - \bar{x})^T \left(I - C_k C_k^T\right) \tag{9}$$

Despite the widespread use of PCA, its efficacy in fault detection is reduced when monitoring industrial processes characterized by complex nonlinear relationships among variables (Lee et al., 2004a). To overcome this issue, Kernel PCA (KPCA) has been proposed to obtain PCs in high-dimensional feature spaces by means of integral operators and nonlinear kernel functions (Cho et al., 2005; Choi et al., 2005). KPCA uses nonlinear mappings to project the input space into a higher-dimensional feature space and then computes the PCs. The original KPCA method (Schölkopf et al., 1998) does not offer a viable solution to reconstruct the input data, making it difficult to implement SPE charts in process monitoring. A solution to this problem is to compute the SPE directly in the feature space (Lee et al., 2004b). There are two main drawbacks in using KPCA for process monitoring. Firstly, the use of a fixed kernel function does not provide great flexibility to the model, making it difficult to select the right kernel for a given process. Secondly, KPCA cannot effectively isolate the faulty process variables in out-of-control situations (Yu and Zhao, 2020). One possible approach is to use a reconstruction-based index (Choi et al., 2005). Due to the infeasibility of the inverse mapping from the feature space to the input space in KPCA (Cheng et al., 2019), the contributions of the process variables for both T^2 and SPE charts are not easily obtained.

Besides the PCA-based methods, deep learning methods in SPC applications have also been proposed. Autoencoders (AEs) have been introduced in the early 90s (Bourlard and Kamp, 1988; Kramer, 1991) but they have lately gained popularity particularly in anomaly detection (Sakurada and Yairi, 2014; Shone et al., 2018; Zhou and Paffenroth, 2017a). AEs can be defined as the quintessential example of representation learning (LeCun et al., 2015) and may essentially be regarded as networks that try to approximate the identity function. An AE is composed of two parts: an encoder, which converts the input into a new representation, and a decoder, which tries to convert this new representation back into its original form. Several variants of the traditional AE are now available to practitioners, e.g., convolutional AEs, contractive AEs, denoising AEs, sparse AEs.

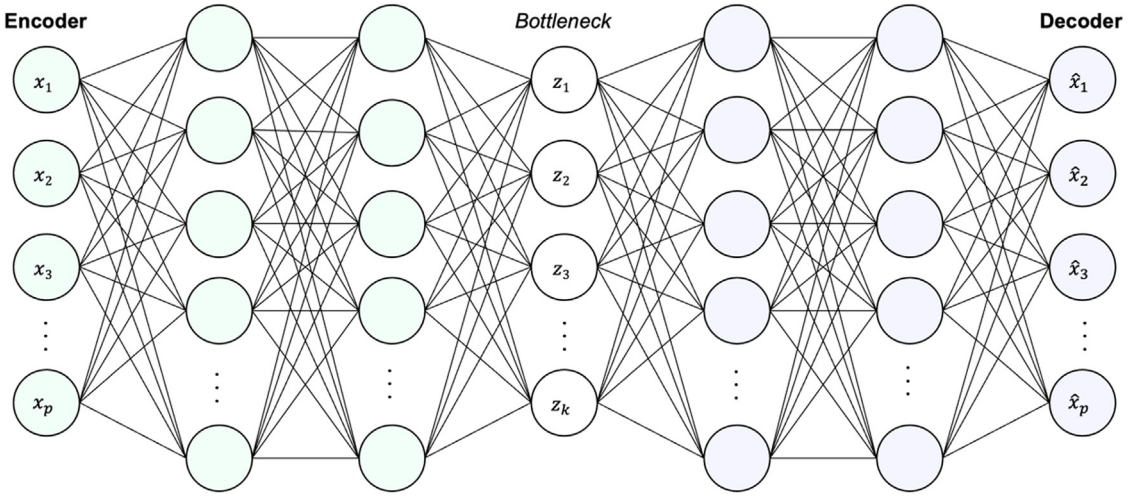

Fig. 1. Structure of a multi-layered AE.

Fig. 1 depicts a general deep AE. The bottleneck is the most interesting part of this network since it can help in revealing the underlying structures for a given dataset (Oring et al., 2020). Depending on the dimensionality of the bottleneck k, we can differentiate between two main types of AE. By using $k < p$ the AE is forced to learn and encode the salient features of the inputs in a lower-dimensional space, resulting in an undercomplete autoencoder. By using a bottleneck with dimensionality $k \geq p$ we obtain an overcomplete AE, that needs to be regularized or constrained in other ways. Otherwise, the network just copies its input features and is not able to generalize. With regards to the depth of the network, the number of hidden layers and the number of neurons in each layer can be very problem-specific and depend on the degree of complexity of the data. In the case of a simple AE, composed only by one hidden layer, the compression, or embedding, can be obtained from the input features as

$$z = \phi(W_e x + b_e) \tag{10}$$

where W_e represents the encoder weight matrix and b_e is the bias term. From the embedding z, the network tries to reconstruct the input features as

$$\hat{x} = \theta(W_d z + b_d) \tag{11}$$

where W_d and b_d are respectively the decoder weight matrix and the bias term. The functions ϕ and θ are generally referred to as activation functions. An AE that uses linear activation functions and minimizes the mean squared error (MSE), spans the same subspaces as PCA (Baldi and Hornik, 1989; Plaut, 2018). AEs with nonlinear encoder and decoder functions can learn a more powerful nonlinear generalization than PCA (LeCun et al., 2015).

The similarity between PCA and AEs motivates the attempts to replicate the traditional PCA-based process monitoring schemes, in the pursuit of a more reliable model for complex nonlinear data. Even though KPCA already offers a solution for nonlinear settings, there are several advantages to using AEs. First of all, in the way nonlinearities are taken into account, AEs offer a much more flexible solution to empirically approximate highly complex kernel functions, eliminating the need for deciding on a fixed kernel function. Other advantages can be found in the possibility of adjusting the structure of the network to efficiently deal with different kinds of data like images or time series (Cheng et al., 2019; Mehdiyev et al., 2017; Sun et al., 2020; Zimmerer et al., 2019).

The use of AEs to reduce the dimensionality of the data before computing the Hotelling T^2 scores is not straightforward. Issues to consider are:

- *Correlated features.* The main motivation behind the use of latent structure methods for SPC is to deal with the correlation in highly dimensional data. However, in the case of a regular AE, it is not guaranteed that the low dimensional latent space will be constituted by uncorrelated features. This represents a significant issue that is often overlooked by researchers and practitioners. Indeed, without warranties on the correlation among the extracted features, the issue of potential difficulties in inverting S in the calculation of T^2 statistic will remain.
- *Normality assumption.* The assumption of the Normal distribution of the p-variate data is rather fundamental in obtaining the UCLs in Eqs. (2) and (3). Yet real-life data may not comply with this assumption. Possible solutions to this problem include the use of variational AEs (VAEs), which are generative models that can reduce the original inputs into a low dimensional representation that follows a multivariate normal distribution (Lee et al., 2019a). Recently, another interesting work on VAEs (Bi and Zhao, 2021) showed how to use self-attention layers for dealing independently with the variable and temporal dimensions in multivariate time series.
- *Contribution plots.* In multivariate SPC, one of the most important steps after the detection of a fault is the diagnosis or root cause analysis. Being able to offer engineers a set of possible causes is crucial to perform timely maintenance and maintain high overall equipment effectiveness (OEE). Unfortunately, when using deep learning methods in this context, besides the possibility of using residuals to get contributions for the SPE chart (Heo and Lee, 2019), a complete solution to get contribution plots for the T^2 does not exist (Zhang et al., 2019). Despite the effectiveness of these deep learning-based methods in modeling nonlinear processes, they often fall short in offering an easy interpretation of the outcomes (Shah et al., 2020). This is also reflected in the multivariate SPC applications that rely on these methods.

We believe that providing a solution for tackling these obstacles may be highly beneficial for enhanced performance in terms of detection and diagnosis. Indeed, despite the fact that deep learning methods can be used to construct control charts and contribution plots in the residual space, we argue that there may be cases where the correlation structure of the process variables would make it easier to detect and identify faults in the feature space. A numerical study that synthetically reproduces similar circumstances is presented in Section 4.

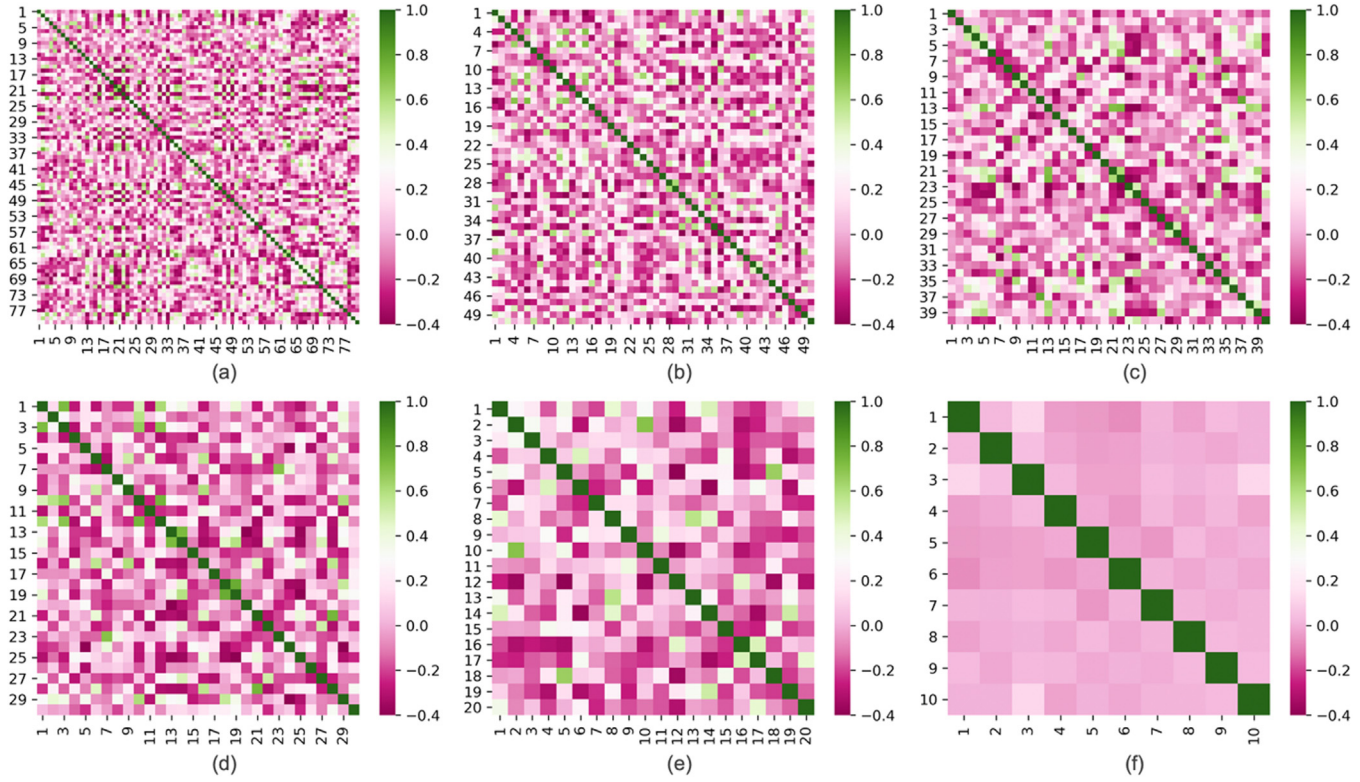

Fig. 2. Progressively learning orthogonality: correlation matrices for the output of the encoder network, from the first layer (a) to the bottleneck (f) with λ=1(one simulation).

3. Proposed approach

3.1. Orthogonal autoencoder

A simple solution to dramatically reduce the possibility of a singular S matrix in Eq. (1) is to impose the orthogonality of the learned representation. This can be done by modifying the loss function of the AE (Wang et al., 2019) to include a regularization term that penalizes the loss function when the learned low-dimensional embedding is not orthogonal

$$L\left(x,\hat{x}\right) = \|x-\hat{x}\|^2 + \lambda\|z^T z - I\|^2 \tag{12}$$

An AE that minimizes the following loss function is referred to as Orthogonal AE (OAE) and it provides quasi-uncorrelated features in its latent space. To investigate the progression of orthogonality in the encoded features, a deep OAE with encoder layer dimensions equal to [100, 80, 50, 40, 30, 20, 10], where 100 refers to the number of input features, was trained. We can see in Fig. 2 how the orthogonality enforced in the latent space is progressively learned by the various layers of the encoder network, and this allows to construct several Hoteling T^2 charts on the intermediate layers and not only on the bottleneck. This also allows for the exploration of the latent space of the encoder network for different dimensions of the extracted features.

The same network was trained without applying the orthogonality regularization to the encoded features, and the resulting correlation plots are shown in Fig. 3. This figure highlights how enforcing orthogonality becomes crucial for SPC purposes, as the encoded features report a high level of correlation, which would render the inversion of the covariance matrix for building an Hotelling T^2 control chart very difficult.

3.2. Upper control limits

Instead of assuming a multivariate normal distribution for the data to obtain the UCLs, we used a kernel density estimator to find the quantiles for the T^2 and SPE control charts (Yan et al., 2016). Given an observation $X_1, X_2, \ldots, X_n$ from an unknown distribution f and given a kernel function K and a positive number h, the kernel density estimator is defined as (Wasserman, 2004)

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h}K\left(\frac{x-X_i}{h}\right) \tag{13}$$

The bandwidth h is used to control the amount of smoothing while the kernel K is a smooth function such that $K(x) \geq 0$, $\int K(x)dx = 1$, $\int xK(x)dx = 0$ and $\sigma_K^2 \equiv \int x^2K(x)dx > 0$. For this study, the Gaussian (Normal) kernel, $K(x) = (2\,\pi)^{-1/2}e^{-x^2/2}$ is used. The UCLs are then defined as the α-upper percentile, with $\alpha = 0.05$. To allow a fair comparison between the proposed method and the traditional approaches, kernel density estimation has also been used to determine the UCLs for the T^2 and SPE control charts based on PCA and KPCA.

3.3. Contribution Plots

For the SPE charts, it is possible to perform a diagnostic analysis by considering the residuals, expressed as the difference between the original input x and the reconstruction $\hat{x}$. Since the OAE learns a regularized representation of the input space, it is expected to learn a salient k-dimensional representation of the p-dimensional input, without being allowed to perfectly replicate the initial input x. Hence, when training our model on Phase I data, we expect not to be able to perfectly reconstruct data points that are dissimilar in the p-dimensional space from the ones that it has seen during training. This means that if the error during the reconstruction is

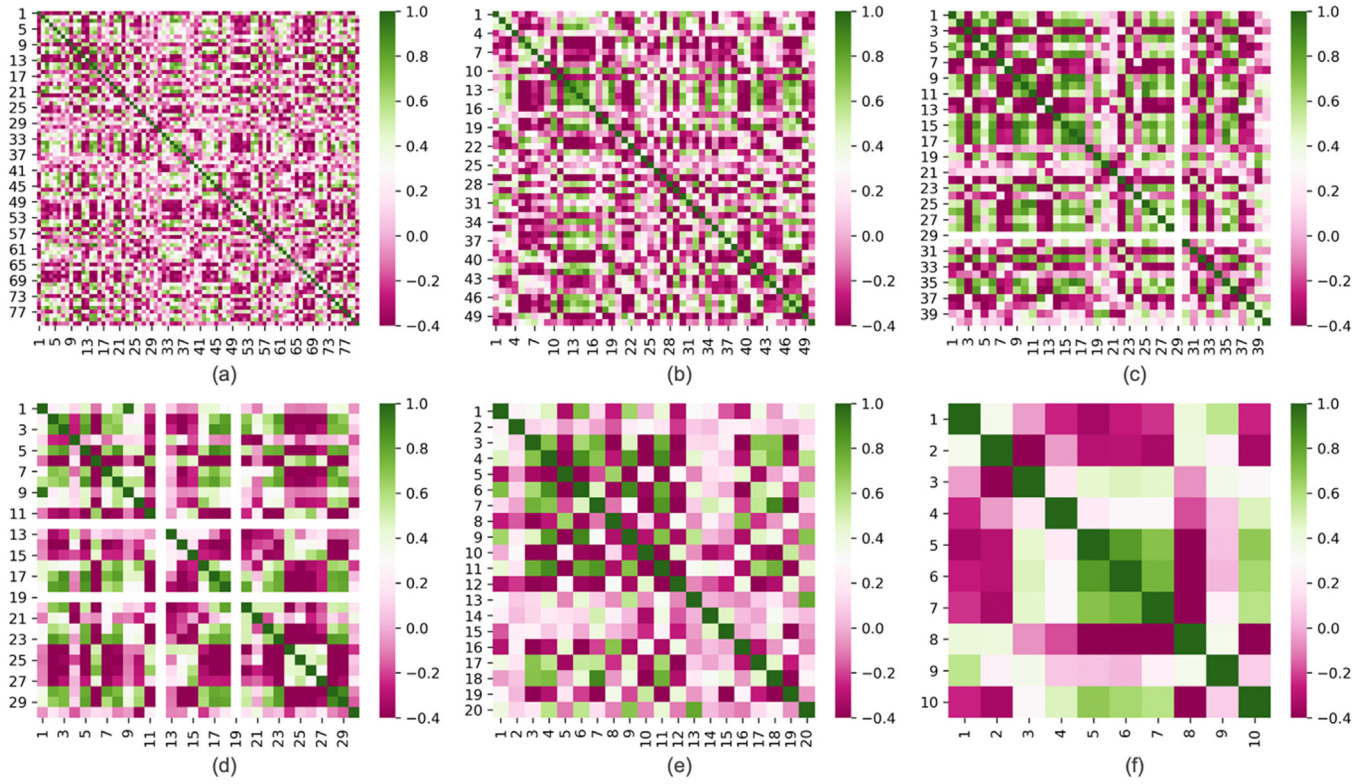

Fig. 3. Correlation matrices for the output of the encoder network, from the first layer (a) to the bottleneck (f) with λ=0 (one simulation).

large, the probability that it will be classified as a fault by the SPE chart will also be high. By decomposing the reconstruction error for each of the original features, we can obtain a contribution plot as in the case of PCA in Eq. (9).

This is however more complicated for the T^2 control chart. Some approaches to SPC with AEs can be found in Yu et al. (2019), Yu and Zhao (2020), and Zhang et al. (2019) but, to the best of our knowledge, this is the first work providing diagnosis measures for the Hotelling T^2 chart when using AEs. To do so, we propose an adaptation of the integrated gradients (IG) technique (Sundararajan et al., 2017), an approach mostly used for classification tasks in computer vision. In this case, we are interested in observing how much the low dimensional representation learned by the network changes as the input observations change. We could do so by computing the product of the feature values of the k-dimensional input and the gradient of our encoder function. However, this will not give us a reliable measure since the use of non-linear activation functions will not make the product sensitive to changes in the input features. For example, using a rectified linear unit (ReLU) as activation function will return exactly the same gradient for all the features with values lower than zero. This is because the ReLU will flatten every negative value to zero, making it impossible for a change in values to be traced by the gradient of the encoder. To tackle this issue, the IG method proposes the use of a baseline (a black image in image classification) to measure how much the output of the network changed from the baseline as the input changed from the baseline. IG are obtained by calculating the integral of the gradients between the baseline and the current point of interest. For SPC applications, we suggest the use of an in-control data point or the sample mean computed in Phase I as the baseline, b. The integral can be approximated with a Riemman sum as in

$$IG_i(x) = (x_i - b_i) \times \sum_{k=1}^{m} \frac{\partial F\left(b + \frac{k}{m} \times (x - b)\right)}{\partial x_i} \times \frac{1}{m} \tag{14}$$

The main difference with the contribution measures used for PCA is that Eqs. (8) and (9) are computed using the retained and disregarded PCs separately, thus representing complementary aspects of the PCA model. This is not true for the two measures proposed for the OAE since, while the IG index offers a view on the latent representation of the model, the SPE diagnosis is focused on the reconstruction errors. Hence, an extremely high value in a specific input feature will affect both the gradient of the encoder function and the specific reconstruction error. Nonetheless, we believe IG represents a highly efficient and effective approach to interpretability for deep neural networks since it allows for obtaining reliable contribution plots for the latent space without the need for using complex methods or algorithms. Moreover, being derived from the image analysis research (Sundararajan et al., 2017), it can be easily applicable in vision-based SPC as well.

4. Simulations

4.1. A motivating example

The following numerical example has the objective of highlighting a pathological case where the Hotelling T^2 control chart may significantly outperform a detection method solely based on the SPE chart. We examine a process with only two process variables, which are normally distributed with mean μ and covariance matrix Σ. We first consider a case where the two variables are not correlated and the variance-covariance matrix Σ is given by

$$\Sigma_u = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Phase I data is collected from the process in normal operating conditions, with μ corresponding to the zero-mean vector [0, 0]. In Phase II data, the process undergoes a fault, represented by a large shift in the process mean, which increases from [0, 0] to

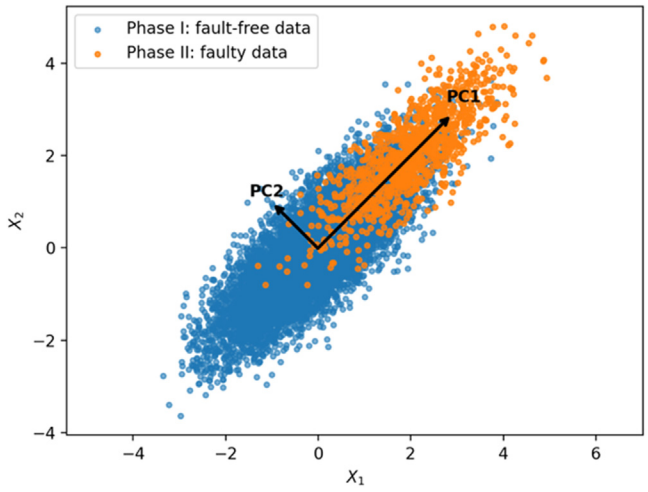

Fig. 4. Scatterplot of the 2D example (one simulation run from the scenario with correlated features).

Table 1
ARL_1 and FDR of the PCA-based control charts of the 2D example (100 simulations).

Scenario	ARL_1 T^2	ARL_1 SPE	FDR T^2	FDR SPE
Uncorrelated features	**8.900**	9.890	0.409	**0.442**
Correlated features	**0.790**	22.850	**0.559**	0.050

[2, 2]. A Hotelling T^2 and a SPE control chart can then be constructed by fitting a PCA-based monitoring system, where we keep one PC for the model space and leave the remaining one for the residual space. To test the detection rates of the two charts we repeated 100 times the following steps: To begin, 10000 fault-free data points are created in Phase I and used to fit a PCA model and estimate the UCLs. In Phase II, 1000 faulty observations are generated to calculate ARL and FDR.

In a second scenario, to assess the impact of the correlation on the detection performance, we assume the two variables to be highly correlated. To generate correlated variables, we simply assume the variance-covariance matrix being equal to

$$\Sigma_c = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

We can see from the scatterplot in Fig. 4 how the shift is always reflected along the axis of the largest PC, which will be the one included in the PCA model. Indeed, the out-of-control data points are well within the range of the in-control observations along the axis corresponding to the second PC, which is the one associated with the residual space. If we were to reduce the problem to one dimension, we would only be able to detect the shift by projecting the data points along the PC1 axis in Fig. 4.

Accordingly with this intuition, the T^2 and SPE control charts report substantially dissimilar detection results. An example of the two charts is reported in Fig. 5. It is worth emphasizing that the key factor that leads to this finding is that the mean shift affects both correlated variables; the situation would have been different if only one of the two variables was affected by a shift in the mean. The ARL and FDR of the PCA-based control charts from the 100 simulations are reported in Table 1. It should be noted how in the scenario with correlated variables, the detection performance of the SPE chart is dramatically worsened. Indeed, the ARL and the FDR are very close to the nominal value of 20 and 5%, which are the same values that are obtained with fault-free data.

The example has been shown using PCA because it is easier to visualize the PC directions in the two-dimensional plot. However, we repeated the same experiment with the AE-based control charts and obtained similar behavior in terms of detection. There was no need to enforce orthogonality in this basic situation because the bottleneck dimensionality was decreased to one. Hence, the regularization term has been omitted during the training by setting the λ shown in Eq. (12) equal to zero. Non-linearities are introduced in both the encoder and decoder layers by using point-wise hyperbolic tangent (Tanh) functions. The same procedure used for PCA has been replicated 100 times, training the network on 10000 observations from each scenario, with an early stop criterion on the validation loss, to arrest the learning procedure after 10 epochs without improvement and avoid overfitting. It is clear from the data in Table 2 that the PCA intuition still holds true in the deep learning context. With correlated data, the T^2 chart significantly outperforms the SPE one, despite the latter's improvement from the PCA-based scenario.

Table 2
ARL_1 and FDR of the AE-based control charts of the 2D example (100 simulations).

Scenario	ARL_1 T^2	ARL_1 SPE	FDR T^2	FDR SPE
Uncorrelated features	4.940	**3.730**	0.445	**0.475**
Correlated features	**0.720**	9.040	**0.555**	0.110

This simple example highlights how the feature space can be crucial, in terms of fault detection, when the process variables are highly correlated.

4.2. Simulation setup

We now test the performance of the OAE-based process control scheme against more conventional methods using high-dimensional simulated data. Several datasets have been generated to analyze detection and diagnosis performances in the presence of process mean shifts. Hence, the covariance matrix of the process has not been considered to be affected by faults and remained constant in both Phase I and Phase II. Without the mean shifts, the data is generated using

$$X = \mu_0 + Y \tag{15}$$

where μ_0 corresponds to the zero-mean vector $[0, 0, \ldots, 0]^T$ and $Y \sim N_p(0, \Sigma_0) Y \sim N_p(0, \Sigma_0)$. The matrix Σ_0 represents a unit variance-covariance matrix with randomly sized correlation blocks. The dimension of a block ranges from two variables to 15% of the number of variables p. $Cov(x_i, x_j) = 0.9$ if x_i and x_j belong to the same block and 0.2 otherwise. Faults have been introduced by a shift in the process mean, which is increased from μ_0 to

$$\mu_i = \mu_0 + \delta_i \cdot \sigma \quad \text{for} \quad i = 1, \ldots, 5 \tag{16}$$

where i corresponds to the ith level of fault magnitude introduced and δ_i goes from 0.2 to 1 in 0.2 increments.

Two cases are considered, the first one with 50 process variables and the second one with 100. The covariance matrices used in the two cases are reported in Fig. 6. Within each case, three different scenarios have been simulated to investigate the detection and diagnosis performance of the Hotelling T^2 and SPE charts for the considered methods. In scenario A, 10% of the variables belonging to different correlation blocks are affected by the mean shift. In scenario B for the case with 50 variables, all variables belonging to the first two correlation blocks are shifted. In scenario B for the case with 100 variables, the first three blocks are affected by the shift. Note that the number of blocks affected by the fault differs in order to attain similar ratios between faulty and non-faulty variables. Finally, scenario C combines scenarios A and B.

The training sets used for the 50 and 100 variables cases contain 100.000 observations each. The test sets contain 1.000 observations each and have been replicated 1.000 times to compute ARL, FDR, and FAR.

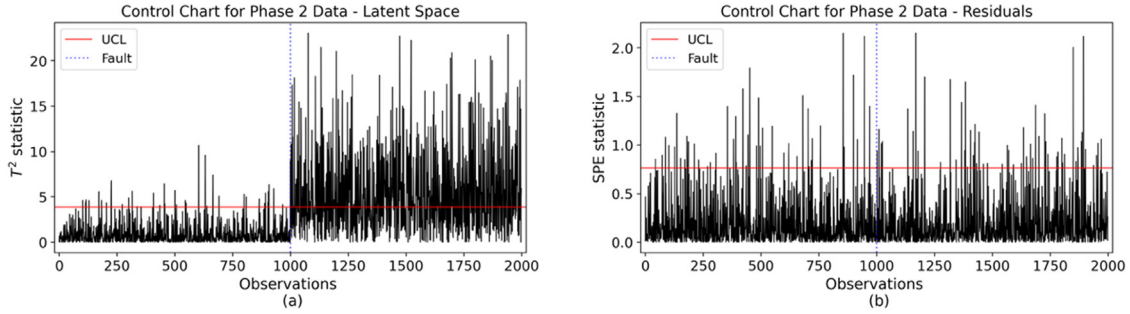

Fig. 5. PCA-based control charts of the 2D example (one simulation run from the scenario with correlated features): Hotelling T^2 (a) and SPE (b).

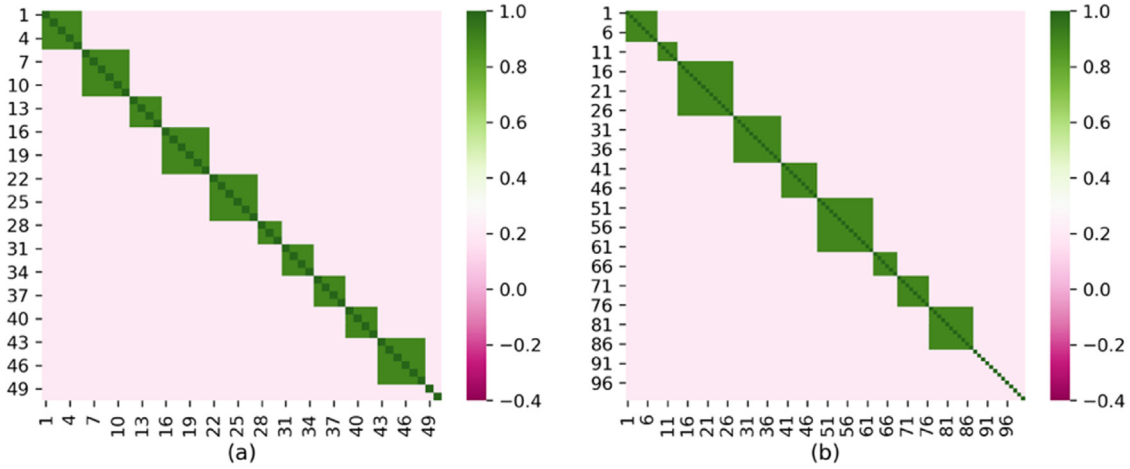

Fig. 6. Covariance matrices: 50 variables (a) and 100 variables (b).

4.3. Fault detection results

The proposed method is compared to PCA, KPCA, and an ordinary AE by observing ARL and FDR. The UCL is found by setting the false alarm rate to 5%, which corresponds to a nominal value of 20 for the ARL. With regards to the configuration of the models, OAE is the only one requiring hyperparameter tuning on the number and dimensionality of the layers and the number of extracted features (Cheng et al., 2019). In this case, we could opt for an undercomplete or overcomplete OAE, depending on the structure of the data. Even if we use an overcomplete AE, which is a network whose bottleneck has a dimensionality that is higher than the number of process variables p, the model could still be able to properly generalize given the orthogonality constraint enforced on the z_k extracted features. Despite most of the focus is often dedicated to dimensionality reduction, learning useful higher-level representation is also a task of interest for various problems (Ca et al., 2010) and it cannot be tackled with the use of PCA, given that the number of axes where the data can be projected is at most equal to the number of features p.

For PCA- and KPCA-based control charts, the number of retained PCs is the most important model configuration criterion. Following an established practice in multivariate SPC, we chose the number of principal components accounting for at least 80% of the total variance in Phase I data (Lee et al., 2019b). KPCA is implemented using a radial basis function kernel. The ordinary AE, hereinafter simply referred to as AE, is designed with linear activation functions and no regularization enforced on the bottleneck. The number of hidden layers is in this case kept at 1 since there is no value-added in stacking multiple linear layers, and the dimensionality of the bottleneck is the same as the number of retained PCs. It should be noted that the AE is only included in the analysis for illustrative purposes. Indeed, confirming the theory introduced in Section 2 (LeCun et al., 2015; Oring et al., 2020), the experiments will show how the detection results obtained with the AE converge to the ones obtained with PCA. The OAE is instead composed of several stacked nonlinear layers, with ReLU activation functions. We observed enhanced detection results by first expanding the feature space and then reducing it back to the dimension of the input space. The dimensionality of the layers of the encoder network for the scenarios with 50 and 100 variables is [50, 250, 100, 50] and [100, 500, 200, 100], respectively; the decoder is constructed in a symmetric fashion. The choice of keeping a higher dimension than the one chosen for PCA is motivated by some recent works (Li et al., 2021; Zhang et al., 2018), whose findings suggested how the detection and diagnosis performances of regularized AEs could be improved by increasing the size of the hidden layers and the bottleneck. This strategy is most effective when applied altogether with some regularization, otherwise there will be a significant risk of overfitting, as anticipated in Section 2 for the case of overcomplete AEs. Finally, the weight of the orthogonal regularization term λ was set to 1. With regards to the training details, we used the Adam optimizer with a learning rate of 0.001 to update the network parameters. As for the number of epochs, to avoid overfitting and favor the convergence of the loss function, instead of setting a fixed number of epochs, a patience on the loss decrease has been set using an early stopping approach for both the AE and the OAE. Indeed, according to this procedure, the training is stopped when the validation loss does not show any improvement for a certain number of consecutive iterations. The number of successive epochs for which no improvement is tolerated is also called patience and was in this case set equal to 10. The validation loss was measured on a separated validation set, which corresponds to 20% of the observations collected in Phase I.

Table 3
ARL_1 of different methods on the simulated dataset (1000 simulations).

Scenario	Process Variables	Shift Size	PCA		KPCA		AE		OAE	
			T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE
A	50	$\delta = 0.2$	19.441	18.614	19.509	16.923	19.792	19.005	**14.497**	15.507
		$\delta = 0.4$	19.01	16.035	18.910	11.871	19.205	16.069	**6.373**	12.048
		$\delta = 0.6$	17.581	11.856	18.073	6.489	17.433	11.596	**2.266**	6.946
		$\delta = 0.8$	17.909	8.012	18.417	2.722	17.603	8.106	**0.705**	3.683
		$\delta = 1.0$	17.589	4.827	17.311	0.958	17.46	4.944	**0.191**	1.865
	100	$\delta = 0.2$	19.081	16.816	17.974	16.555	19.16	16.726	**11.621**	17.902
		$\delta = 0.4$	18.298	11.68	19.338	10.966	18.343	11.663	**4.021**	16.892
		$\delta = 0.6$	18.998	6.673	19.280	5.610	18.826	6.653	**0.979**	13.965
		$\delta = 0.8$	17.487	2.763	17.705	2.282	17.186	2.758	**0.118**	8.647
		$\delta = 1.0$	17.629	1.028	19.186	0.772	17.563	1.000	**0.007**	4.966
B	50	$\delta = 0.2$	16.997	18.504	**16.885**	19.103	16.926	18.354	18.695	17.638
		$\delta = 0.4$	**14.636**	19.035	15.264	19.131	14.916	19.416	14.698	17.493
		$\delta = 0.6$	12.149	18.21	11.819	18.36	12.156	18.229	**11.292**	15.747
		$\delta = 0.8$	8.922	18.695	9.137	18.045	8.890	18.528	**8.626**	14.39
		$\delta = 1.0$	**5.906**	16.686	6.480	17.014	5.992	17.193	6.633	12.484
	100	$\delta = 0.2$	17.437	19.243	17.654	19.871	17.784	19.009	**16.168**	16.862
		$\delta = 0.4$	14.138	18.013	13.376	19.805	13.986	17.58	13.293	**13.116**
		$\delta = 0.6$	10.013	15.861	**9.384**	19.28	10.003	15.875	10.152	10.85
		$\delta = 0.8$	6.975	12.990	**6.149**	20.210	6.978	13.096	7.603	9.608
		$\delta = 1.0$	4.765	11.222	**3.928**	19.841	4.794	11.115	5.494	7.269
C	50	$\delta = 0.2$	18.305	18.009	18.389	18.071	18.653	18.303	**16.615**	17.150
		$\delta = 0.4$	15.713	16.177	15.717	14.261	15.733	16.297	**9.459**	12.766
		$\delta = 0.6$	12.792	14.02	13.666	10.254	12.954	13.997	**4.600**	8.783
		$\delta = 0.8$	8.726	10.789	9.386	5.611	9.052	11.048	**1.868**	4.896
		$\delta = 1.0$	6.602	7.705	7.055	2.996	6.699	7.919	**0.713**	2.714
	100	$\delta = 0.2$	18.776	16.835	17.66	17.98	18.442	16.862	**14.195**	14.984
		$\delta = 0.4$	14.157	12.12	13.645	12.294	14.168	11.921	**5.353**	12.702
		$\delta = 0.6$	10.769	7.418	10.233	7.834	10.776	7.571	**1.811**	10.151
		$\delta = 0.8$	7.835	3.865	7.218	3.802	7.939	3.794	**0.398**	7.318
		$\delta = 1.0$	5.203	1.845	4.462	1.895	5.330	1.845	**0.072**	4.419

From Tables 3 and 4 we can see how the stacked layers of the OAE and the regularization imposed on the orthogonality of the learned representation encourage the encoder network to learn significant features in the latent space, with compelling detection results on the T^2 charts. With regards to PCA and KPCA, the way faults are introduced deeply affects the performances of the two charts. In scenario A, when the faults are introduced to variables belonging to different correlation blocks, the SPE chart is the most effective control chart. Conversely, when whole blocks of correlated variables are shifted (scenario B), the Hotelling T^2 chart performs better. Finally, in the hybrid scenario, both charts report low ARLs, with the SPE being slightly superior. This behavior is further investigated in the fault identification phase of the analysis.

Figs. 7 and 8 show the performance of the OAE-based control charts (with a false alarm rate equal to 5%) for one simulation run. The first 3 charts (a-c) are Hotelling T^2 charts applied to the output of the encoder layers of the network since the OAE can learn representations characterized by quasi-uncorrelated features also in its early layers. SPE chart is reported in the last plot (d). In these examples, Phase II data contains 2000 observations, and faults ($\delta = 1.0$) are introduced on 10% of the variables after the 1000^{th} observation. The process mean shifts are detected by all the charts, with the bottleneck being the most successful one. After the faults are introduced, almost all data points fall above the UCL in the third layer, which is the last layer of the encoder network. The performance of the OAE is similar for the two scenarios (50 and 100 process variables).

4.4. Fault identification results

Once a fault is detected, a contribution analysis based on the T^2 and SPE charts can be performed for all the methods. To better understand the detection behavior, we show the results of the diagnostic analysis for both control charts. The heatmaps shown in Figs. 9–11 report the average contribution of each variable to the faults over 1000 simulation runs for the last 1000 observations of Phase II data in which the process was out-of-control. We only present the results obtained by OAE and PCA since the results obtained with KPCA are very similar to the ones obtained with the PCA. Contributions for the Hotelling T^2 chart of the OAE are obtained by applying Eq. (14) to each observation of the faulty Phase II data. The score represents the approximated integral of the gradients computed on the interpolation between a fixed baseline and the observation of interest. The gradients represent the partial derivatives of the encoder function with respect to the inputs.

We would like to draw attention to a relevant aspect that connects detection delay and fault identification. In real-life applications, as soon as an out-of-control situation is detected by a control chart, we would expect to be able to intervene to investigate the possible root causes of the faults by looking at the contribution plot of the signaling control chart. We would hereinafter refer to this concept as detection-diagnosis coherence. In this regard, we can see from Figs. 9–11 how the OAE is particularly coherent for the T^2 chart. That is, whenever a fault is signaled by the T^2 control chart, we are immediately able to identify all the process disturbance sources from the IG scores. PCA, on the other hand, does not exhibit the same level of coherence. Indeed, regardless of which chart is signaling the fault, the shift related to a whole block of correlated variables is always recognized by the T^2 contribution plots. The behavior of the three scenarios is unaffected by the number of variables included in the simulations.

Besides better detection and diagnosis performances, OAE also has additional advantages, which are mostly related to its flexibility and versatility. First of all, in terms of function approximation, not having a fixed kernel K as in KPCA allows for a better generalization of complex nonlinear relationships. Secondly, the structure of the network may be efficiently adjusted to deal with sequential data or complex sets of images (Beggel et al., 2020; Li et al., 2020;

Table 4
FDR of different methods on the simulated dataset (1000 simulations).

Scenario	Process Variables	Shift Size	PCA		KPCA		AE		OAE	
			T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE
A	50	$\delta = 0.2$	0.050	0.056	0.050	0.059	0.050	0.056	**0.065**	0.060
		$\delta = 0.4$	0.051	0.077	0.051	0.093	0.050	0.077	**0.134**	0.078
		$\delta = 0.6$	0.052	0.131	0.052	0.194	0.052	0.132	**0.306**	0.121
		$\delta = 0.8$	0.053	0.260	0.053	0.431	0.052	0.262	**0.584**	0.207
		$\delta = 1.0$	0.056	0.516	0.055	0.755	0.055	0.518	**0.842**	0.355
	100	$\delta = 0.2$	0.050	0.058	0.049	0.059	0.052	0.056	**0.076**	0.054
		$\delta = 0.4$	0.050	0.092	0.049	0.094	0.055	0.085	**0.203**	0.058
		$\delta = 0.6$	0.051	0.176	0.050	0.185	0.061	0.157	**0.529**	0.069
		$\delta = 0.8$	0.052	0.364	0.050	0.387	0.069	0.324	**0.883**	0.099
		$\delta = 1.0$	0.054	0.665	0.052	0.702	0.081	0.615	**0.993**	0.176
B	50	$\delta = 0.2$	0.052	0.050	0.052	0.051	0.052	0.050	0.051	**0.053**
		$\delta = 0.4$	**0.061**	0.050	0.060	0.050	**0.061**	0.050	0.054	0.054
		$\delta = 0.6$	**0.077**	0.051	0.075	0.050	**0.077**	0.051	0.061	0.058
		$\delta = 0.8$	**0.100**	0.052	0.099	0.051	**0.100**	0.052	0.071	0.065
		$\delta = 1.0$	**0.135**	0.054	0.132	0.051	**0.135**	0.053	0.086	0.076
	100	$\delta = 0.2$	0.054	0.050	0.053	0.050	0.054	0.049	0.052	**0.059**
		$\delta = 0.4$	0.065	0.050	0.064	0.050	0.066	0.049	0.055	**0.067**
		$\delta = 0.6$	**0.088**	0.050	0.085	0.050	**0.088**	0.049	0.063	0.079
		$\delta = 0.8$	0.124	0.050	0.120	0.050	**0.125**	0.049	0.075	0.096
		$\delta = 1.0$	0.178	0.050	0.172	0.050	**0.179**	0.049	0.093	0.118
C	50	$\delta = 0.2$	0.052	0.054	0.052	0.055	0.052	0.054	**0.058**	0.057
		$\delta = 0.4$	0.061	0.066	0.060	0.072	0.060	0.066	**0.096**	0.071
		$\delta = 0.6$	0.075	0.091	0.074	0.112	0.074	0.093	**0.183**	0.102
		$\delta = 0.8$	0.098	0.142	0.095	0.205	0.096	0.146	**0.344**	0.162
		$\delta = 1.0$	0.131	0.243	0.126	0.382	0.128	0.252	**0.568**	0.265
	100	$\delta = 0.2$	0.053	0.056	0.051	0.056	0.054	0.055	**0.068**	0.059
		$\delta = 0.4$	0.064	0.078	0.062	0.079	0.067	0.075	**0.151**	0.071
		$\delta = 0.6$	0.083	0.128	0.079	0.133	0.089	0.121	**0.365**	0.091
		$\delta = 0.8$	0.115	0.232	0.109	0.244	0.125	0.218	**0.698**	0.127
		$\delta = 1.0$	0.162	0.417	0.153	0.442	0.180	0.392	**0.934**	0.187

Table 5
FAR of different methods on the simulated dataset (1000 simulations).

Scenario	Process Variables	PCA		KPCA		AE		OAE	
		T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE
Fault-free	50	0.050	0.050	0.050	0.051	0.050	0.050	0.050	0.054
	100	0.050	0.050	0.049	0.050	0.051	0.049	0.050	0.054

Table 6
Allocation of faults to variables in the simulated scenarios.

Scenario	Process Variables	Faulty Variables
A	50	1, 11, 21, 31, 41
	100	1, 11, 21, 31, 41, 51, 61, 71, 81, 91
B	50	From 1 to 11
	100	From 1 to 27
C	50	From 1 to 11 + 21, 31, 41
	100	From 1 to 27 + 31, 41, 51, 61, 71, 81, 91

Ma and Li, 2020; Zhou and Paffenroth, 2017b). Having a model that can deal with images or time-series data is particularly useful, especially in data-rich environments like smart factories, where IoT allows the simultaneous gathering of data in many different shapes and formats. Moreover, it has been shown how autocorrelation impacts PCA-based SPC by causing lower false alarm rates and delayed shift detection (Vanhatalo and Kulahci, 2016).

5. Tennessee Eastman process

The Tennessee Eastman Process (TEP) is a well-known process simulator that is often used as a benchmark in chemical engineering research. Its initial publication dates back to the early 90s (Downs and Vogel, 1993) and, since then, it has been commonly considered the gold standard testbed for comparing SPC approaches to fault detection and diagnosis (Capaci et al., 2019). It has also been extensively studied from a process dynamics and control perspective (Lawrence Ricker, 1996; Lyman and Georgakis, 1995; McAvoy and Ye, 1994; Ricker, 1995). Recently, an extended version of TEP has been published (Reinartz et al., 2021). In this application, we used the data available in Rieth et al. (2017). Including all the manipulated and control variables, 52 process variables are available and 20 different faults are introduced (Table 8). As in Gajjar et al. (2018), 33 variables (Table 7) have been used for predictive and diagnostic purposes. Composition measurements are excluded since they are not practical to be tracked during routine operations and agitator speed values have also been excluded since they are constant. In Table 7, XMEAS refers to continuous process measurements and XMV to process manipulated variables.

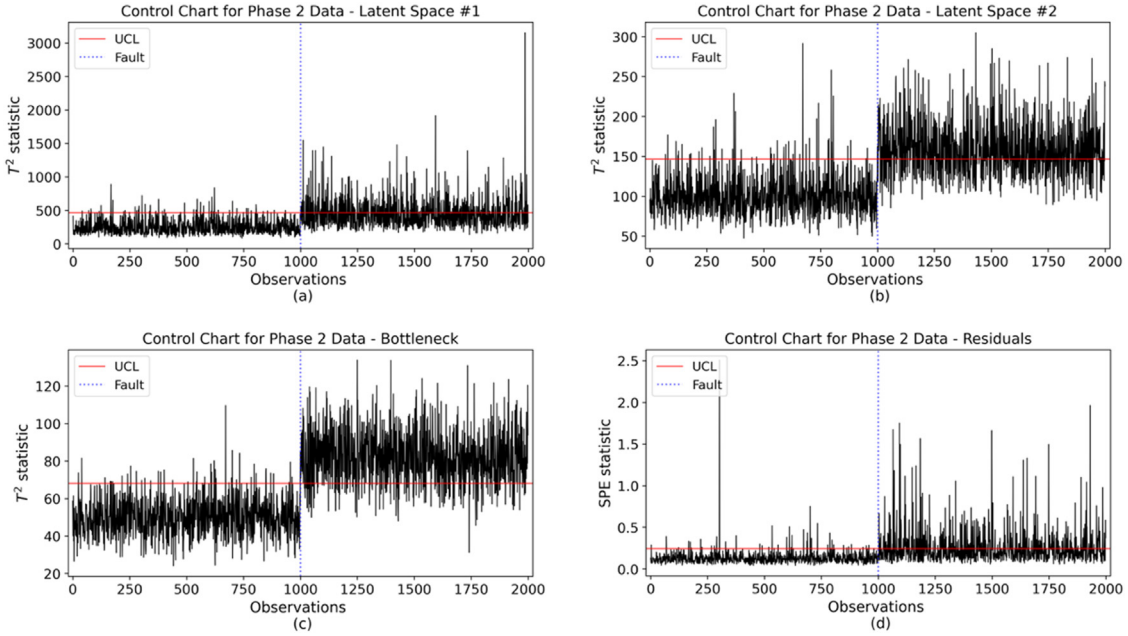

Fig. 7. OAE-based control charts (one simulation run from scenario A, with 50 variables and $\delta = 1.0$): Hotelling T^2 (a-c) and SPE (d).

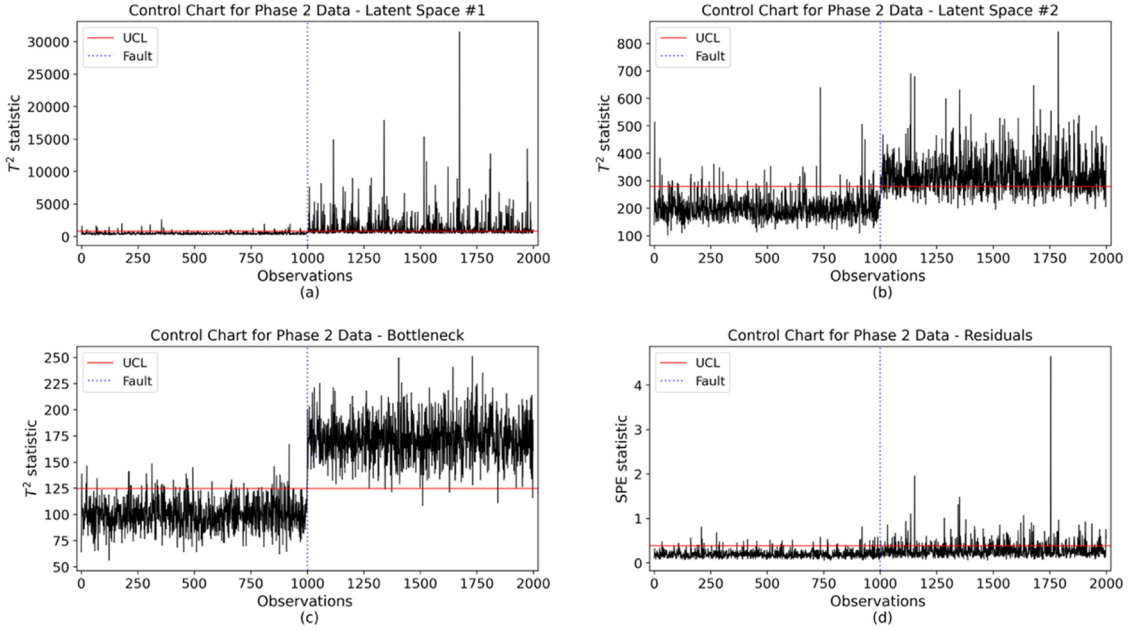

Fig. 8. OAE-based control charts (one simulation run from scenario A, with 100 variables and $\delta = 1.0$): Hotelling T^2 (a-c) and SPE (d).

5.1. Fault detection results

The different faults, shown in Table 8, are introduced in Phase II data after 8 simulation hours, which correspond to 160 observations. Figs. 13 to 15 show how observations begin to plot above the UCL only after the faults are introduced. The values in the plots are reported on a logarithmic scale as the statistics of out-of-control data points are several orders of magnitude higher than the ones corresponding to in-control observations. As in the numerical simulations, the faults could also be detected through the Hotelling T^2 charts based on the early layers of the encoder, given that the extracted features are sufficiently uncorrelated to allow for the inversion of the sample covariance matrix. For this case study, the dimensionality of the layers of the encoder network for the OAE is [33, 100, 75, 50, 25], with a symmetrically designed decoder. The training details for OAE and AE are the same as the ones specified for the numerical study. The number of retained PCs is set to 14 as in other related works (Gajjar et al., 2018). Also in this case, we used a higher dimensionality for the bottleneck of the OAE as other approaches based on regularized AEs reported better monitoring performances with increased width of the layers (Li et al., 2021; Zhang et al., 2018).

From Tables 9 and 10, which show the ARL and FDR results of the four methods, we can see how the OAE attains superior detection performances for many of the reported faults. The upcoming section will focus on the diagnosis phase of the analysis.

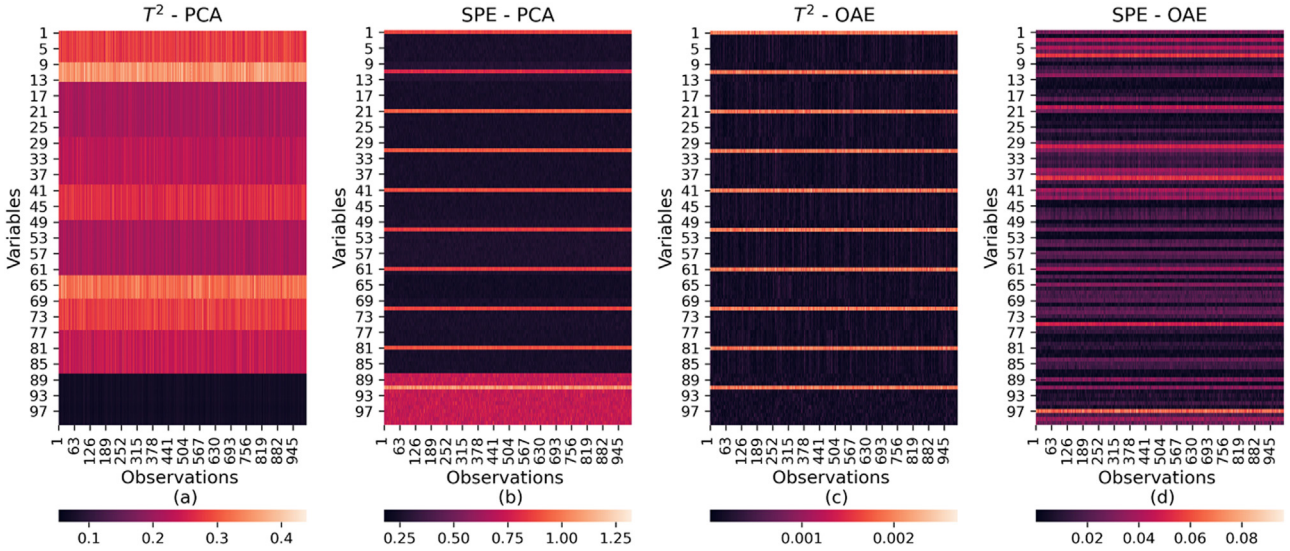

Fig. 9. Hotelling T^2 diagnosis and SPE diagnosis with PCA (a, b) and OAE (c, d) (1000 simulations from scenario A, with 100 variables and $\delta = 1.0$).

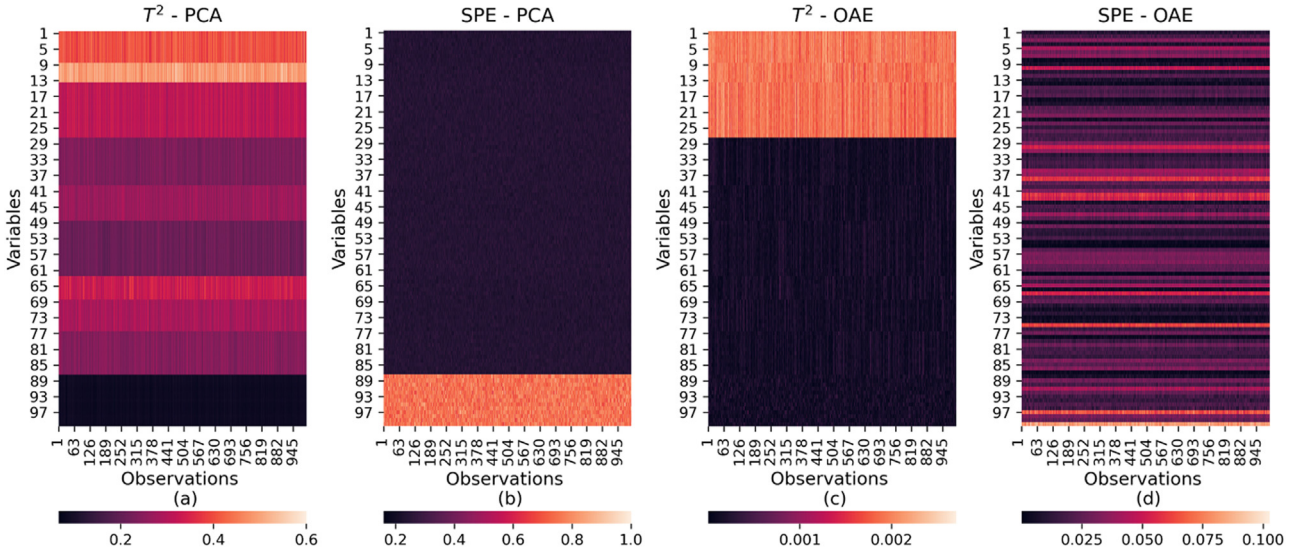

Fig. 10. Hotelling T^2 diagnosis and SPE diagnosis with PCA (a, b) and OAE (c, d) (1000 simulations from scenario B, with 100 variables and $\delta = 1.0$).

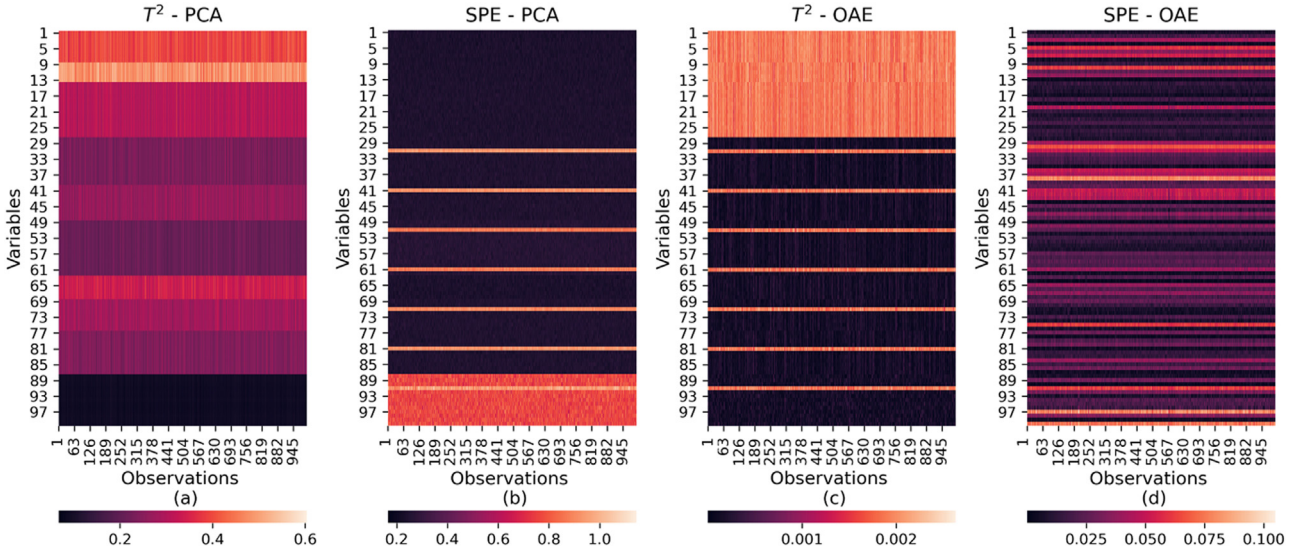

Fig. 11. Hotelling T^2 diagnosis and SPE diagnosis with PCA (a, b) and OAE (c, d) (1000 simulations from scenario C, with 100 variables and $\delta = 1.0$).

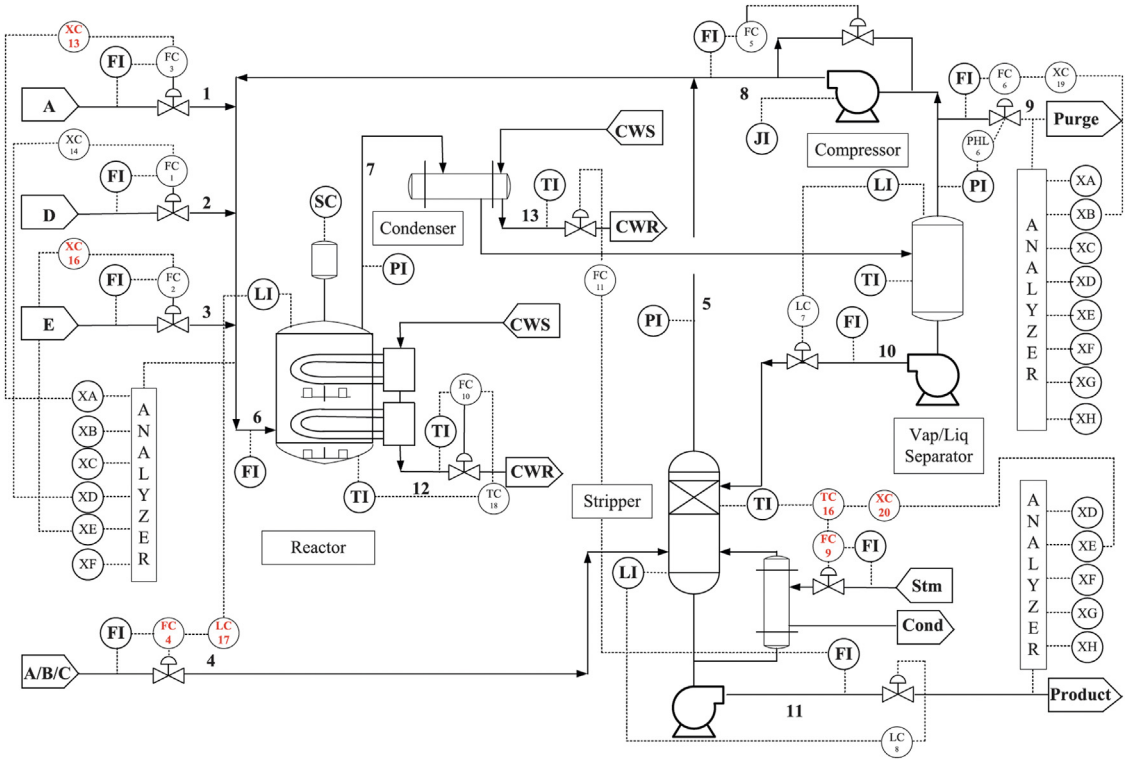

Fig. 12. Flowsheet of the TEP (Liu, 2012).

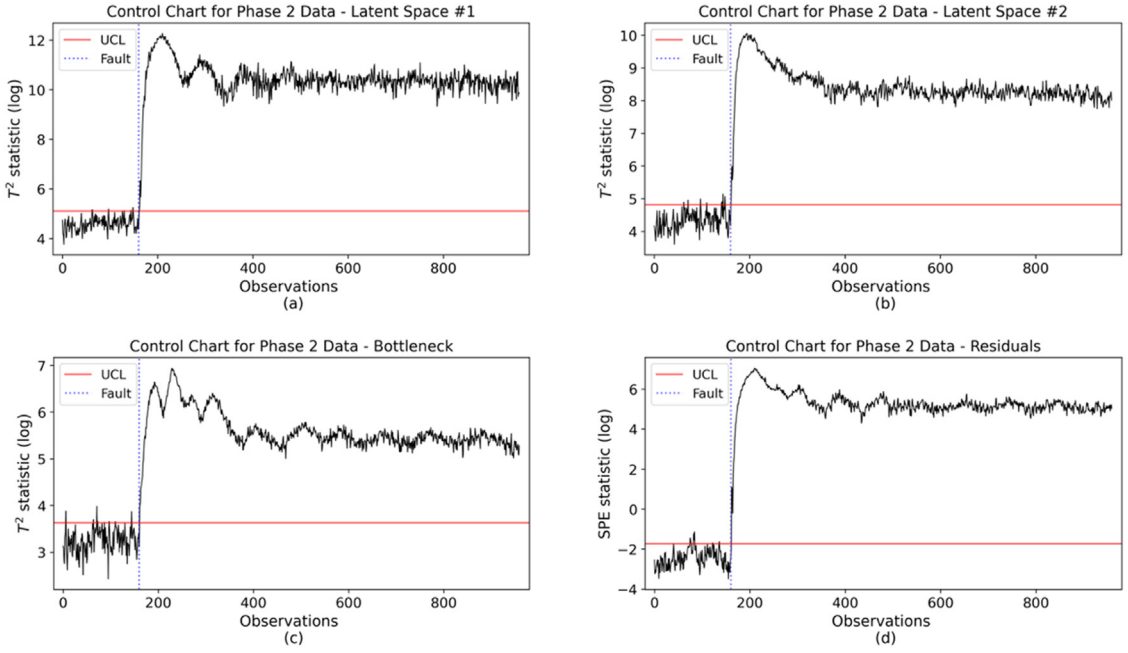

Fig. 13. OAE-based control charts (one simulation run with IDV 1): Hotelling T^2 (a-c) and SPE (d).

5.2. Fault identification results

As the TEP is affected by several types of faults, we dedicated the fault identification of analysis to three different kinds of process disturbances.

5.2.1. IDV 1 (step)

As expected, since there are no faults introduced before the 160*th* observation, no variable shows any strong contribution in those observations. After that, both PCA and OAE reveal the candidate process variables for the root cause analysis, as Fault 1 is a relatively easy fault to identify. Indeed, we can see from Fig. 16 that many individual variables go out-of-control. Some of these variables, as is the case of the A feed (variable 1) and the A feed flow (variable 25), assume a significantly high value and remain above the UCL throughout the monitoring period. This is reflected in all the contribution plots. During this time window, other variables such as the A and C feed (variable 4), the stripper temperature (variable 18), the stripper steam flow (variable 19), and the stripper steam valve (variable 31) remain in an out-of-

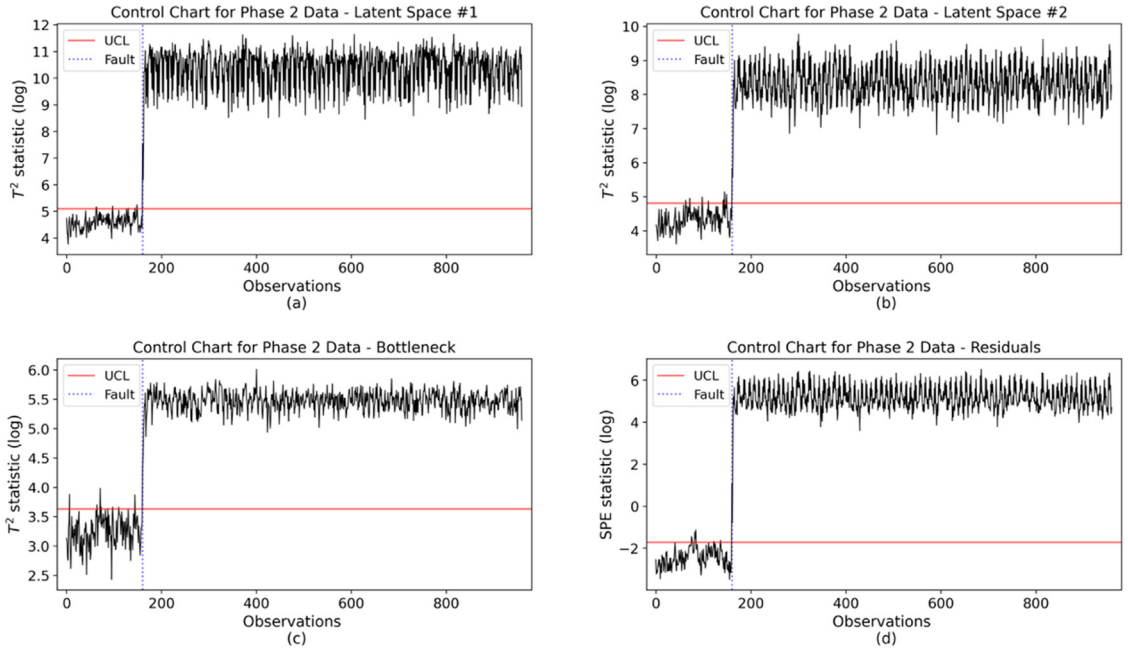

Fig. 14. OAE-based control charts (one simulation run with IDV 14): Hotelling T^2 (a-c) and SPE (d).

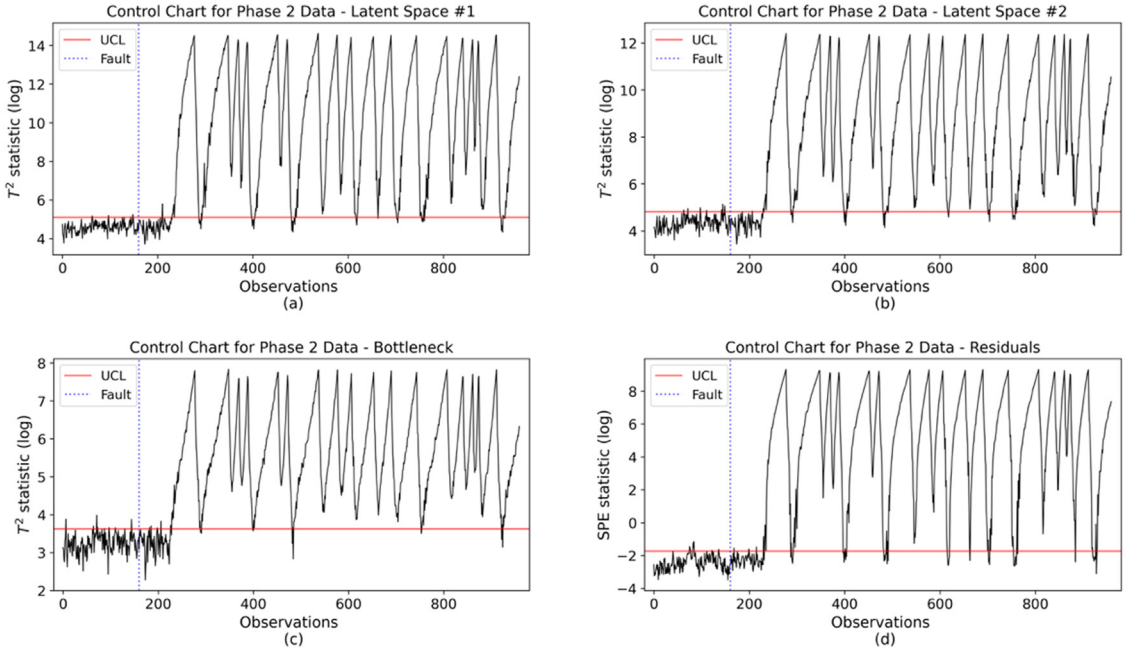

Fig. 15. OAE-based control charts (one simulation run with IDV 17): Hotelling T^2 (a-c) and SPE (d).

control state. It can be seen, however, that their respective values do not exceed the individual UCLs, resulting in lower contributions for both PCA and OAE.

5.2.2. IDV 14 (sticking)

Although this fault is still relatively straightforward to detect, it does provide some enticing fault identification insights. Fig. 18 shows that there are only three variables that exhibit an out-of-control behavior. These are the reactor temperature (variable 9), the reactor cooling water temperature (variable 21), and the reactor cooling water flow (variable 32). However, numerous additional variables report a large contribution to the fault in the PCA contribution plots. This issue is referred to in the literature as the smearing effect (Liu, 2012), and may lead to a misinterpretation of the root cause variables. Conversely, the IG scores of the OAE offer a clear view of the true causes of the process disturbance. It should be noted how the smearing effect also affects the contributions on the SPE side of the OAE. This highlights anew how exploring the contributions in the feature space might be highly beneficial for the fault diagnostic.

5.2.3. IDV 17 (unknown)

Despite being a significantly different fault in terms of detection performance, the variables that are out-of-control during IDV 17 and IDV 14 are the same. OAE is still coherent on the T^2 side and, although displaying lower contributions for reactor cooling water temperature and reactor cooling water flow, it still manages to only focus on the three variables that assume anomalous values.

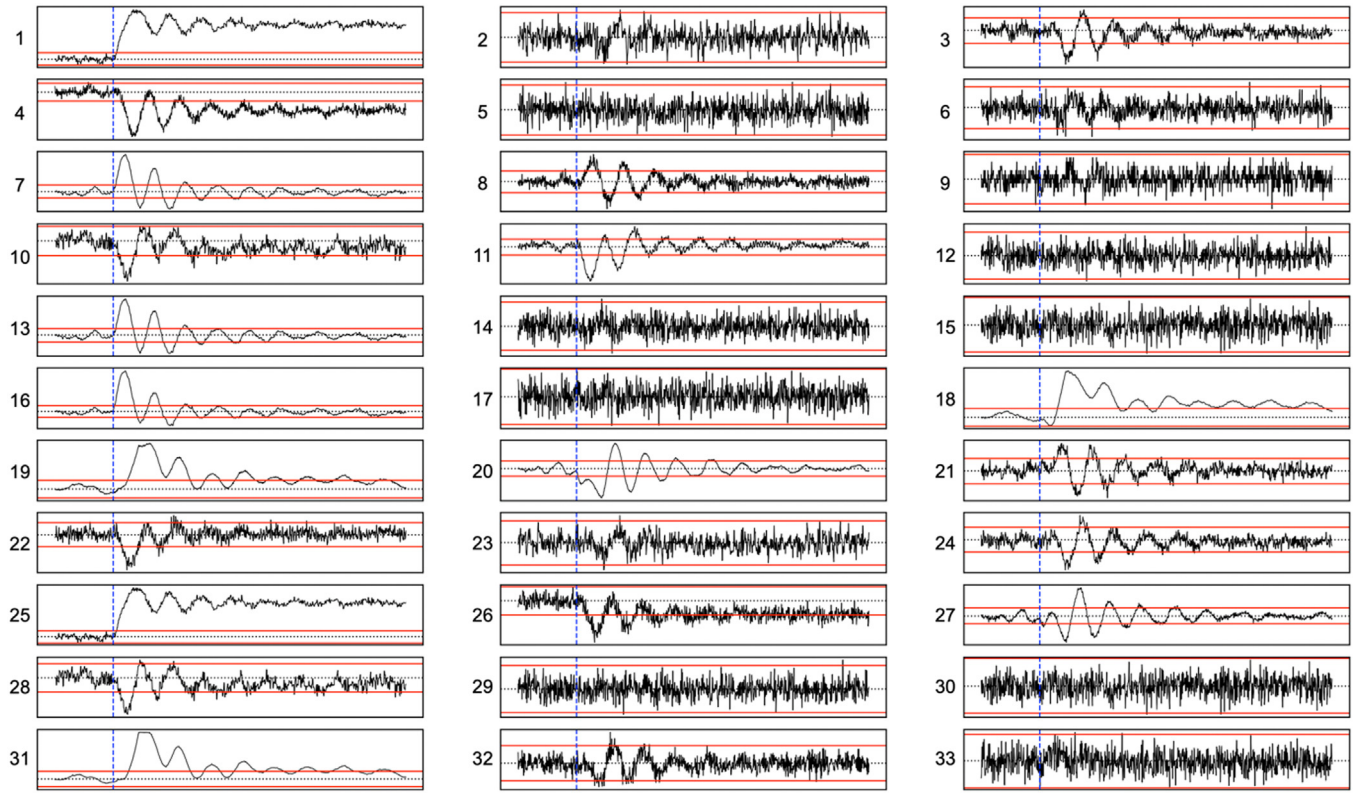

Fig. 16. Individual control charts for the 33 variables, the black dotted line represents the mean from Phase I and the red lines represent the control limits obtained as the Phase I mean ± 3 standard deviations, the blue dashed line indicates the fault start (one simulation run with IDV 1).

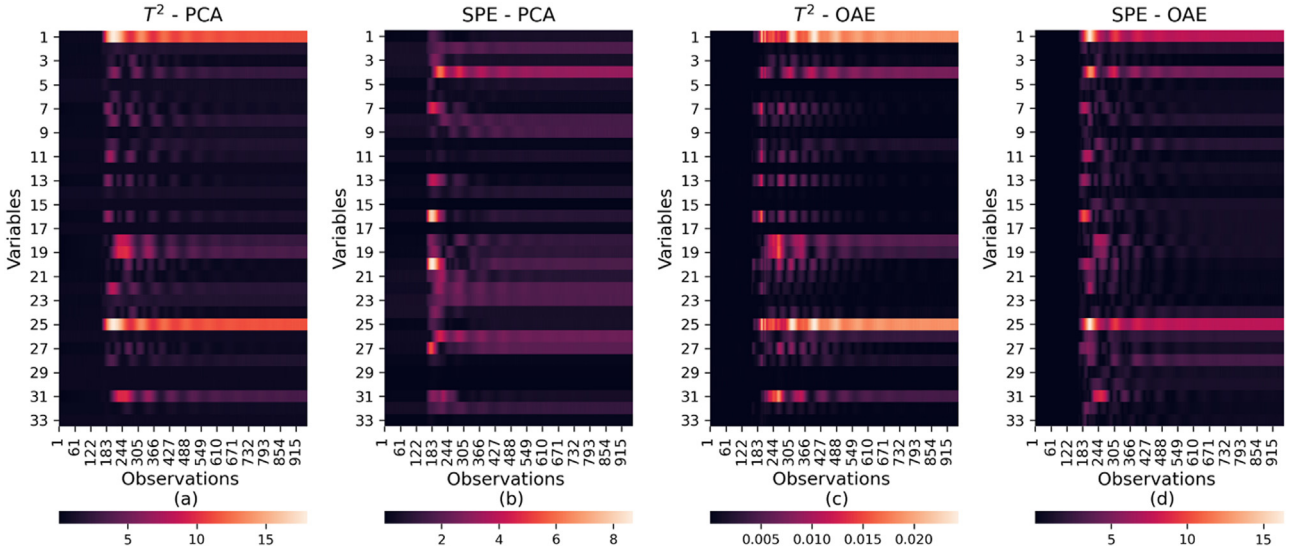

Fig. 17. Hotelling T^2 diagnosis and SPE diagnosis with PCA (a, b) and OAE (c, d) on the IDV 1 (500 simulations).

Table 7
Monitored variables of the TEP.

Process Variable	ID	Process Variable	ID	Process Variable	ID
A Feed (Stream 1)	XMEAS 1	Separator Level	XMEAS 12	D Feed Flow (Stream 2)	XMV 1
D Feed (Stream 2)	XMEAS 2	Separator Pressure	XMEAS 13	E Feed Flow (Stream 3)	XMV 2
E Feed (Stream 3)	XMEAS 3	Product Separator Underflow (Stream 10)	XMEAS 14	A Feed Flow (Stream 1)	XMV 3
A and C Feed (Stream 4)	XMEAS 4	Stripper Level	XMEAS 15	A and C Feed Flow (Stream 4)	XMV 4
Recycle Flow (Stream 8)	XMEAS 5	Stripper Pressure	XMEAS 16	Compressor Recycle Valve	XMV 5
Reactor Feed Rate (Stream 6)	XMEAS 6	Stripper Underflow (Stream 11)	XMEAS 17	Purge Valve (Stream 9)	XMV 6
Reactor Pressure	XMEAS 7	Stripper Temperature	XMEAS 18	Separator Pot Liquid Flow (Stream 10)	XMV 7
Reactor Level	XMEAS 8	Stripper Steam Flow	XMEAS 19	Stripper Liquid Product Flow (Stream 11)	XMV 8
Reactor Temperature	XMEAS 9	Compressor Work	XMEAS 20	Stripper Steam Valve	XMV 9
Purge Rate (Stream 9)	XMEAS 10	Reactor Cooling Water Outlet Temperature	XMEAS 21	Reactor Cooling Water Flow	XMV 10
Separator Temperature	XMEAS 11	Separator Cooling Water Outlet Temperature	XMEAS 22	Condenser Cooling Water Flow	XMV 11

Table 8
Process disturbances of the TEP.

IDV	Description	Type
1	A/C feed ratio, B composition constant (Stream 4)	Step
2	B composition, A/C ratio constant (Stream 4)	Step
3	D feed temperature (Stream 2)	Step
4	Reactor cooling water inlet temperature	Step
5	Condenser cooling water inlet temperature	Step
6	A feed loss (Stream 1)	Step
7	C header pressure loss-reduced availability (Stream 4)	Step
8	A, B, C feed composition (Stream 4)	Random variation
9	D feed temperature (Stream 2)	Random variation
10	C feed temperature (Stream 4)	Random variation
11	Reactor cooling water inlet temperature	Random variation
12	Condenser cooling water inlet temperature	Random variation
13	Reaction kinetics	Slow drift
14	Reactor cooling water valve	Sticking
15	Condenser cooling water valve	Sticking
16	Unknown	Unknown
17	Unknown	Unknown
18	Unknown	Unknown
19	Unknown	Unknown
20	Unknown	Unknown

Table 9
ARL_1 of different methods on the TEP simulation data (500 simulations).

IDV	PCA		KPCA		AE		OAE	
	T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE
1	4.349	1.545	1.663	3.577	3.603	1.563	**1.354**	2.273
2	8.707	9.553	7.918	12.701	8.585	9.874	**7.061**	10.242
3	30.741	21.655	31.160	21.066	28.303	21.281	**17.768**	45.586
4	2.062	1.018	1.339	1.064	1.443	1.076	**1.000**	1.313
5	**1.000**	1.036	**1.000**	1.768	1.004	1.002	**1.000**	**1.000**
6	4.399	**1.000**	3.691	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
7	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
8	13.876	10.752	11.585	12.926	13.086	11.170	**10.465**	13.232
9	30.345	21.403	31.285	20.894	28.760	21.232	**15.394**	39.788
10	26.164	16.437	26.491	15.617	21.090	15.637	**12.000**	13.626
11	5.982	4.569	5.062	5.499	5.353	5.106	**4.384**	5.939
12	5.273	5.685	5.299	6.537	5.555	5.451	**4.778**	5.333
13	20.703	16.852	20.022	17.581	20.002	17.265	**14.111**	24.232
14	1.335	1.036	1.110	1.090	1.188	1.060	**1.030**	1.051
15	26.090	20.200	27.341	19.854	25.575	19.635	**14.525**	39.798
16	24.208	12.032	23.056	11.986	17.669	11.417	**7.697**	7.717
17	18.068	14.822	17.912	14.731	17.725	15.160	**12.677**	20.354
18	22.419	17.715	22.906	17.138	21.862	17.782	**13.010**	26.152
19	5.044	3.477	5.459	3.663	5.242	2.980	1.424	**1.313**
20	22.531	17.279	22.214	17.297	21.182	17.399	**13.242**	21.576

Table 10
FDR of different methods on the TEP simulation data (500 simulations).

IDV	PCA		KPCA		AE		OAE	
	T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE
1	0.993	**0.998**	**0.998**	0.994	0.993	**0.998**	**0.998**	0.997
2	0.986	0.973	0.987	0.962	0.986	0.974	**0.988**	0.983
3	0.053	0.054	0.055	0.052	0.053	0.053	**0.066**	0.057
4	0.317	**1.000**	0.678	**1.000**	0.508	**1.000**	**1.000**	0.996
5	0.299	0.229	0.308	0.205	0.322	0.272	**1.000**	**1.000**
6	0.992	**1.000**	0.994	**1.000**	0.999	**1.000**	**1.000**	**1.000**
7	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
8	0.967	0.917	0.978	0.854	0.967	0.922	**0.979**	0.976
9	0.055	0.055	0.057	0.053	0.055	0.054	**0.069**	0.060
10	0.344	0.419	0.410	0.337	0.351	0.430	0.807	**0.907**
11	0.461	**0.869**	0.594	0.788	0.542	0.847	0.860	0.740
12	0.984	0.958	0.986	0.938	0.985	0.951	**0.991**	**0.991**
13	0.943	0.931	0.950	0.900	0.943	0.931	**0.954**	0.949
14	0.986	**1.000**	0.999	0.999	0.998	**1.000**	**1.000**	**1.000**
15	0.062	0.057	0.064	0.054	0.062	0.056	**0.075**	0.070
16	0.175	0.351	0.219	0.298	0.186	0.380	0.820	**0.940**
17	0.775	0.942	0.846	0.921	0.791	0.941	**0.949**	0.920
18	0.934	0.942	0.933	0.942	0.935	0.942	**0.945**	0.940
19	0.310	0.280	0.271	0.328	0.345	0.274	0.849	**0.916**
20	0.370	0.593	0.501	0.524	0.383	0.592	0.751	**0.800**

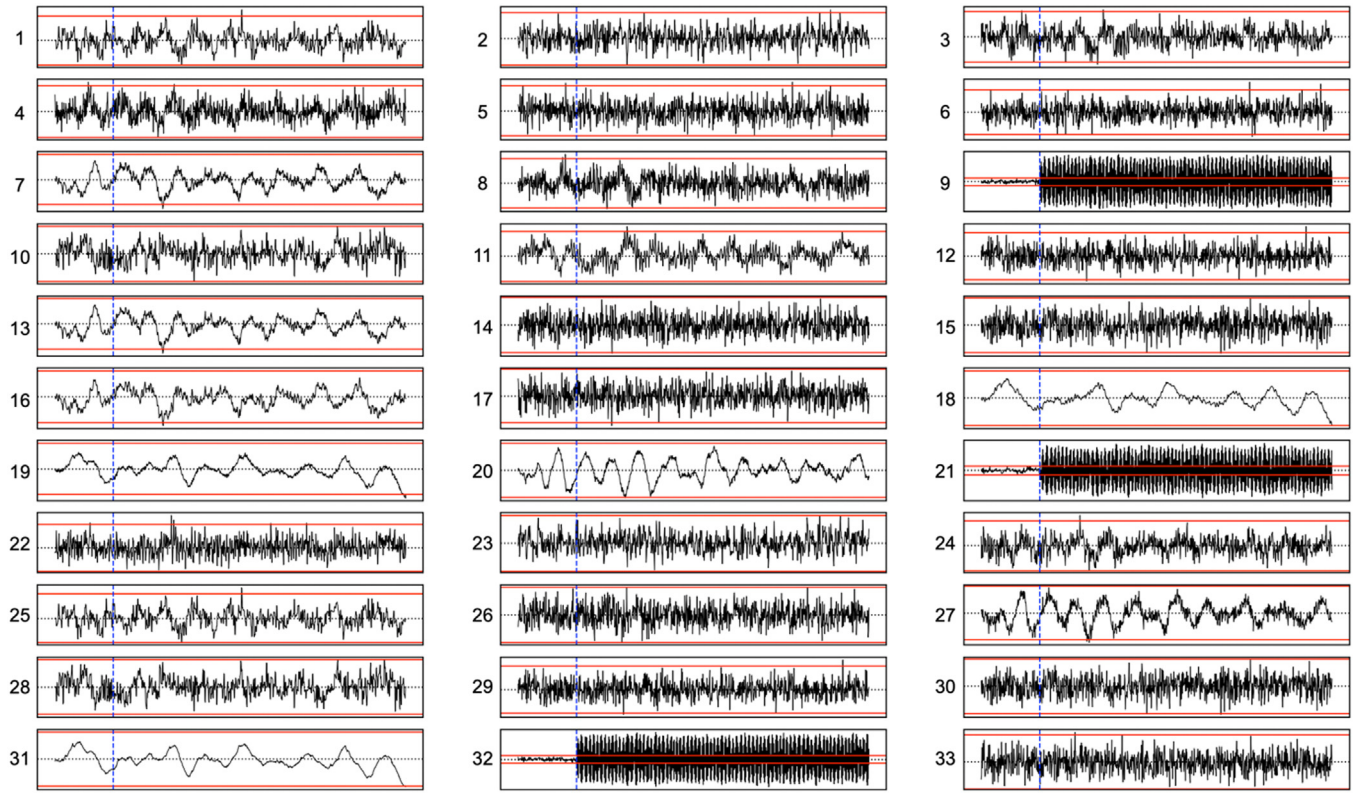

Fig. 18. Individual control charts for the 33 variables, the black dotted line represents the mean from Phase I and the red lines represent the control limits obtained as the Phase I mean ± 3 standard deviations, the blue dashed line indicates the fault start (one simulation run with IDV 14).

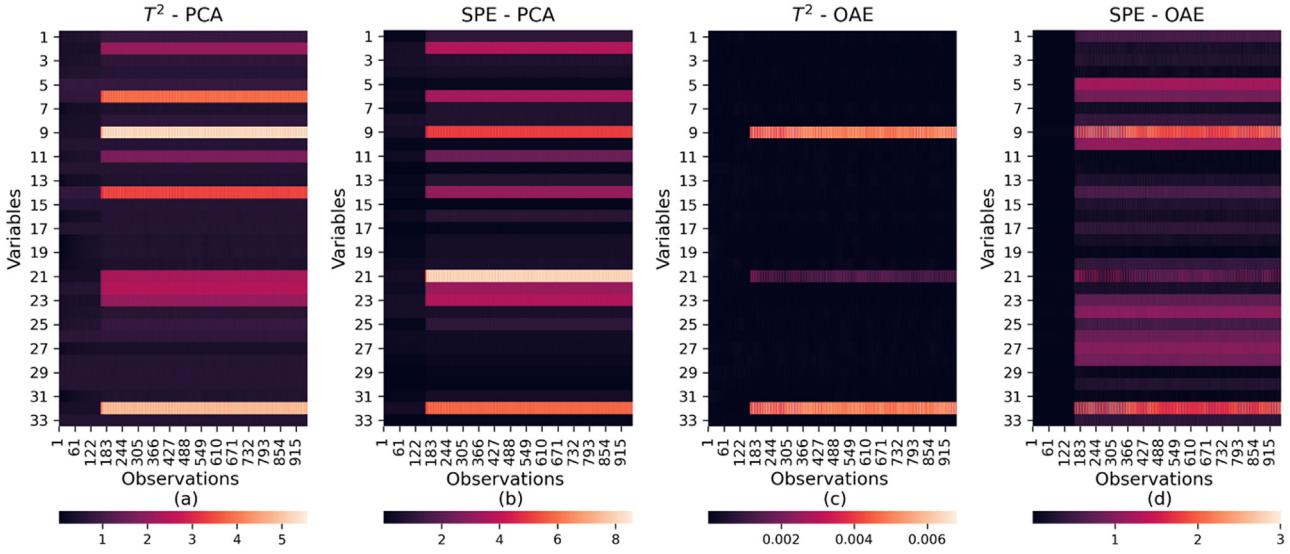

Fig. 19. Hotelling T^2 diagnosis and SPE diagnosis with PCA (a, b) and OAE (c, d) on the IDV 14 (500 simulations).

The smearing effect still exists in PCA, and the contributions of variables that do not alter their value from in-control observations are highlighted just as much as the contributions of process variables 9, 21, and 32. Indeed, PCA suggests the D feed (variable 2), reactor feed rate (variable 6), separator temperature (variable 11), product separator underflow (variable 14), and D feed flow (variable 23) as plausible root causes in this case. The OAE's SPE contributions accurately identify the reactor cooling water temperature, but they are unable to correctly isolate the other two out-of-control variables.

6. Conclusions

Deep learning algorithms are becoming extremely popular in high-dimensional industrial problems. In SPC applications, AEs can be used, similarly to PCA, to construct a Hotelling T^2 control chart on the latent space and a SPE chart on the residuals. However, when process variables are highly correlated, constructing a Hotelling T^2 control chart on the latent space of an AE is not reliable as the covariance matrix of the extracted features may not be invertible. Modifying the loss function of multi-layered AEs,

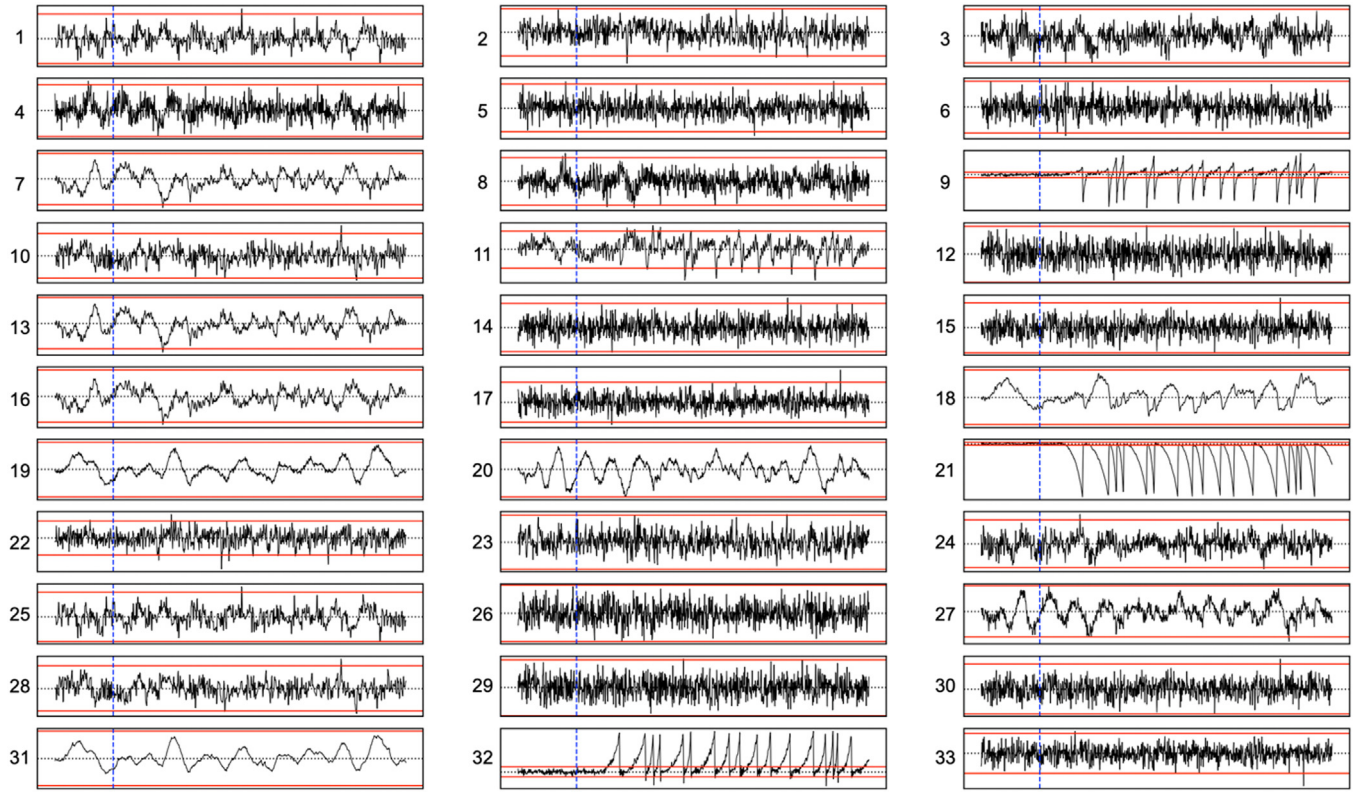

Fig. 20. Individual control charts for the 33 variables, the black dotted line represents the mean from Phase I and the red lines represent the control limits obtained as the Phase I mean ± 3 standard deviations, the blue dashed line indicates the fault start (one simulation run with IDV 17).

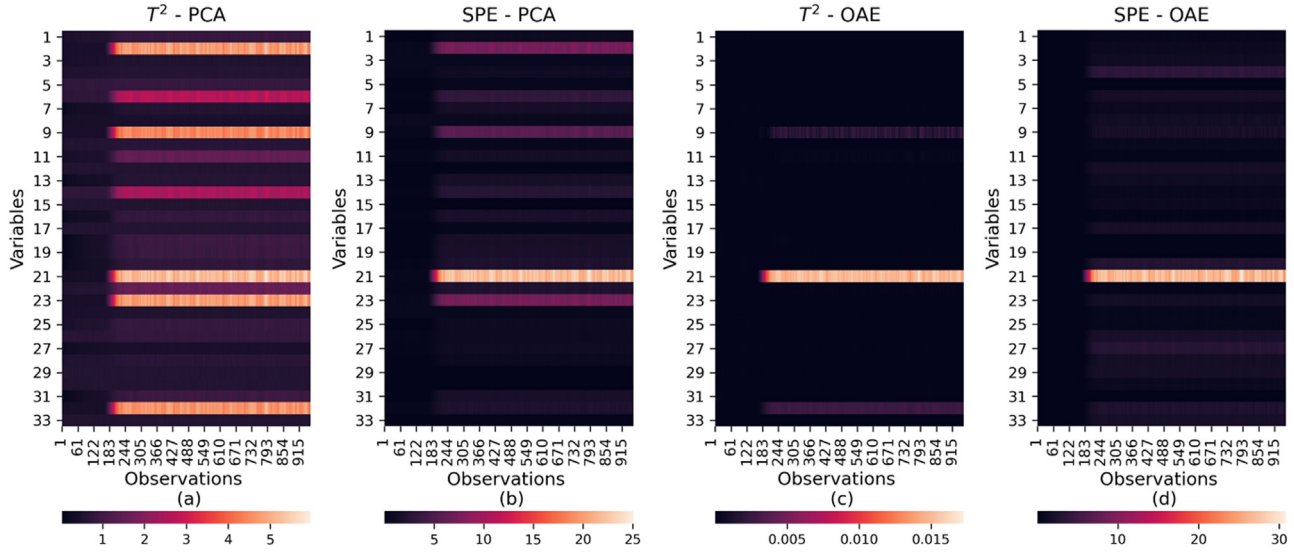

Fig. 21. Hotelling T^2 diagnosis and SPE diagnosis with PCA (a, b) and OAE (c, d) on the IDV 17 (500 simulations).

Table 11
FAR of different methods on the TEP simulation data (500 simulations).

IDV	PCA		KPCA		AE		OAE	
	T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE
0	0.049	0.051	0.051	0.050	0.049	0.051	0.059	0.052

we can obtain a quasi-orthogonal representation of the input features in various latent spaces of the network and subsequently use the Hotelling T^2 control chart. The modified loss function also allows the OAE to learn salient features in its bottleneck, resulting in highly favorable detection results. We also showed how to perform root cause analysis for the OAE-based control charts through the interpretation of the latent space of the network using the IG scores. Moreover, this diagnosis approach offers compelling performances, being able to identify the faulty variables in each simulated scenario and reducing the smearing effect on contribution plots. In that regard, OAE offers a promising alternative to traditional methods for both fault detection and diagnosis.

Author contributions

Author 1: Davide Cacciarelli

- Conceived and designed the analysis
- Contributed data or analysis tools (implementation of the computer code)
- Performed the analysis
- Wrote the paper

Author 2: Murat Kulahci

- Conceived and designed the analysis
- Performed the analysis
- Wrote the paper
- Provided guidance and supervision

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Baldi, P., Hornik, K., 1989. Neural networks and principal component analysis: learning from examples without local minima. Neural Netw..

Beggel, L., Pfeiffer, M., Bischl, B., 2020. Robust Anomaly Detection in Images Using Adversarial Autoencoders doi:10.1007/978-3-030-46150-8_13.

Bi, X., Zhao, J., 2021. A novel orthogonal self-attentive variational autoencoder method for interpretable chemical process fault detection and identification. Process Saf. Environ. Prot. 156, 581–597. doi:10.1016/j.psep.2021.10.036.

Bourlard, H., Kamp, Y., 1988. Auto-association by multilayer perceptrons and singular value decomposition. Biol. Cybern. 59. doi:10.1007/BF00332918.

Box, G., Bisgaard, S., Graves, S., Kulahci, M., Marko, K., James, J., van Gilder, J., Ting, T., Zatorski, H., Wu, C., 2003. Performance evaluation of dynamic monitoring systems: the waterfall chart. Qual. Eng. 16. doi:10.1081/QEN-120024006.

Box, G.E.P., 1954. Some theorems on quadratic forms applied in the study of analysis of variance problems, i. effect of inequality of variance in the one-way classification. Ann. Math. Stat. 25. doi:10.1214/aoms/1177728786.

Ca, P.V., Edu, L.T., Lajoie, I., Ca, Y.B., Ca, P.-A.M., 2010. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion pascal vincent hugo larochelle yoshua bengio pierre-antoine manzagol. J. Mach. Learn. Res..

Capaci, F., Vanhatalo, E., Kulahci, M., Bergquist, B., 2019. The revised Tennessee Eastman process simulator as testbed for SPC and DoE methods. Qual. Eng. 31. doi:10.1080/08982112.2018.1461905.

Cheng, F., He, Q.P., Zhao, J., 2019. A novel process monitoring approach based on variational recurrent autoencoder. Comput. Chem. Eng. 129. doi:10.1016/j.compchemeng.2019.106515.

Cho, J.H., Lee, J.M., Wook Choi, S., Lee, D., Lee, I.B., 2005. Fault identification for process monitoring using kernel principal component analysis. Chem. Eng. Sci. 60. doi:10.1016/j.ces.2004.08.007.

Choi, S.W., Lee, C., Lee, J.-M., Park, J.H., Lee, I.-B., 2005. Fault detection and identification of nonlinear processes based on kernel PCA. Chemom. Intell. Lab. Syst. 75. doi:10.1016/j.chemolab.2004.05.001.

Chong, N.L., Khoo, M.B.C., Haq, A., Castagliola, P., 2019. Hotelling's T2 control charts with fixed and variable sample sizes for monitoring short production runs. Qual. Reliab. Eng. Int. 35, 14–29. doi:10.1002/qre.2377.

Downs, J.J., Vogel, E.F., 1993. A plant-wide industrial process control problem. Comput. Chem. Eng. 17. doi:10.1016/0098-1354(93)80018-I.

Frumosu, F.D., Kulahci, M., 2019. Outliers detection using an iterative strategy for semi-supervised learning. Qual. Reliab. Eng. Int. 35. doi:10.1002/qre.2522.

Gajjar, S., Kulahci, M., Palazoglu, A., 2018. Real-time fault detection and diagnosis using sparse principal component analysis. J. Process Control 67, 112–128. doi:10.1016/j.jprocont.2017.03.005.

Heo, S., Lee, J.H., 2019. Statistical process monitoring of the Tennessee Eastman process using parallel autoassociative neural networks and a large dataset. Processes 7. doi:10.3390/pr7070411.

Hotelling, H., 1947. Multivariate quality control. Techn. Stat. Anal..

Jackson, J.E., Mudholkar, G.S., 1979. Control procedures for residuals associated with principal component analysis. Technometrics 21. doi:10.1080/00401706.1979.10489779.

Jolliffe, I.T., Cadima, J., 2016. Principal component analysis: a review and recent developments. Philos. Trans. R. Soc., A 374. doi:10.1098/rsta.2015.0202.

Kramer, M.A., 1991. Nonlinear principal component analysis using autoassociative neural networks. AIChE J. 37. doi:10.1002/aic.690370209.

Lawrence Ricker, N., 1996. Decentralized control of the Tennessee Eastman challenge process. J. Process Control 6. doi:10.1016/0959-1524(96)00031-5.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521. doi:10.1038/nature14539.

Lee, J.-M., Yoo, C., Choi, S.W., Vanrolleghem, P.A., Lee, I.-B., 2004a. Nonlinear process monitoring using kernel principal component analysis. Chem. Eng. Sci. 59. doi:10.1016/j.ces.2003.09.012.

Lee, J.-M., Yoo, C., Choi, S.W., Vanrolleghem, P.A., Lee, I.-B., 2004b. Nonlinear process monitoring using kernel principal component analysis. Chem. Eng. Sci. 59. doi:10.1016/j.ces.2003.09.012.

Lee, S., Kwak, M., Tsui, K.L., Kim, S.B., 2019a. Process monitoring using variational autoencoder for high-dimensional nonlinear processes. Eng. Appl. Artif. Intell. 83, 13–27. doi:10.1016/j.engappai.2019.04.013.

Lee, S., Kwak, M., Tsui, K.L., Kim, S.B., 2019b. Process monitoring using variational autoencoder for high-dimensional nonlinear processes. Eng. Appl. Artif. Intell. 83, 13–27. doi:10.1016/j.engappai.2019.04.013.

Li, N., Shi, H., Song, B., Tao, Y., 2020. Temporal-spatial neighborhood enhanced sparse autoencoder for nonlinear dynamic process monitoring. Processes 8. doi:10.3390/pr8091079.

Li, Y., Chai, Y., Yin, H., 2021. Autoencoder embedded dictionary learning for nonlinear industrial process fault diagnosis. J. Process Control 101, 24–34. doi:10.1016/j.jprocont.2021.02.002.

Liu, J., 2012. Fault diagnosis using contribution plots without smearing effect on non-faulty variables. J. Process Control 22, 1609–1623. doi:10.1016/j.jprocont.2012.06.016.

Liu, Y.J., André, S., saint Cristau, L., Lagresle, S., Hannas, Z., Calvosa, É., Devos, O., Duponchel, L., 2017. Multivariate statistical process control (MSPC) using Raman spectroscopy for in-line culture cell monitoring considering time-varying batches synchronized with correlation optimized warping (COW). Anal. Chim. Acta 952, 9–17. doi:10.1016/j.aca.2016.11.064.

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., Bachem, O., 2019. Challenging common assumptions in the unsupervised learning of disentangled representations. In: Chaudhuri, K., Salakhutdinov, R. (Eds.), Proceedings of the 36th International Conference on Machine Learning, Proceedings of Machine Learning Research. PMLR, pp. 4114–4124.

Lyman, P.R., Georgakis, C., 1995. Plant-wide control of the Tennessee Eastman problem. Comput. Chem. Eng. 19. doi:10.1016/0098-1354(94)00057-U.

Ma, Y., Li, H., 2020. GRU-auto-encoder neural network based methods for diagnosing abnormal operating conditions of steam drums in coal gasification plants. Comput. Chem. Eng. 143. doi:10.1016/j.compchemeng.2020.107097.

MacGregor, J.F., Kourti, T., 1995. Statistical process control of multivariate processes. Control Eng. Pract. 3. doi:10.1016/0967-0661(95)00014-L.

MacGregor, J.F., Yu, H., Muñoz, S.G., Flores-Cerrillo, J., 2005. Data-based latent variable methods for process analysis, monitoring and control, in: computers and Chemical Engineering. Comput. Chem. Eng. 1217–1223. doi:10.1016/j.compchemeng.2005.02.007.

McAvoy, T.J., Ye, N., 1994. Base control for the Tennessee Eastman problem. Comput. Chem. Eng. 18. doi:10.1016/0098-1354(94)88019-0.

Mehdiyev, N., Lahann, J., Emrich, A., Enke, D., Fettke, P., Loos, P., 2017. Time series classification using deep learning for process planning: a case from the process industry. In: Procedia Comput. Sci.. Elsevier B.V., pp. 242–249. doi:10.1016/j.procs.2017.09.066.

Montgomery, DC, 2017. Introduction to Statistical Quality Control, 7th ed. John Wiley & Sons.

Moreira, B.R., de, A., Cruz, V.H., Oliveira, M.L.C., Viana, R., da, S., 2021. Full-scale production of high-quality wood pellets assisted by multivariate statistical process control. Biomass Bioenergy 151. doi:10.1016/j.biombioe.2021.106159.

Oring, A., Yakhini, Z., Hel-Or, Y., 2020. Autoencoder Image Interpolation by Shaping the Latent Space.

Plaut, E., 2018. From Principal Subspaces to Principal Components with Linear Autoencoders.

Reinartz, C., Kulahci, M., Ravn, O., 2021. An extended Tennessee Eastman simulation dataset for fault-detection and decision support systems. Comput. Chem. Eng. 149. doi:10.1016/j.compchemeng.2021.107281.

Ricker, N.L., 1995. Optimal steady-state operation of the Tennessee Eastman challenge process. Comput. Chem. Eng. 19. doi:10.1016/0098-1354(94)00043-N.

Rieth, C.A., Amsel, B.D., Tran, R., Cook, M.B., 2017. Additional Tennessee Eastman process simulation data for anomaly detection evaluation. Harvard Dataverse.

Runger, G.C., Alt, F.B., Montgomery, D.C., 1996. Contributors to a multivariate statistical process control chart signal. Commun. Stat. Theory Methods 25. doi:10.1080/03610929608831832.

Sabahno, H., Amiri, A., Castagliola, P., 2018. Evaluating the effect of measurement errors on the performance of the variable sampling intervals Hotelling's T2 control charts. Qual. Reliab. Eng. Int. 34, 1785–1799. doi:10.1002/qre.2370.

Sakurada, M., Yairi, T., 2014. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In: Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis - MLSDA'14. ACM Press, New York, New York, USA doi:10.1145/2689746.2689747.

Schölkopf, B., Smola, A., Müller, K.R., 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput. 10. doi:10.1162/089976698300017467.

Shah, D., Wang, J., He, Q.P., 2020. Feature engineering in big data analytics for IoT-enabled smart manufacturing – Comparison between deep learning and statistical learning. Comput. Chem. Eng. 141. doi:10.1016/j.compchemeng.2020.106970.

Shone, N., Ngoc, T.N., Phai, V.D., Shi, Q., 2018. A deep learning approach to network intrusion detection. IEEE Trans. Emerg. Top. Comput. Intell. 2. doi:10.1109/TETCI.2017.2772792.

Silva, A.F., Sarraguça, M.C., Fonteyne, M., Vercruysse, J., de Leersnyder, F., Vanhoorne, V., Bostijn, N., Verstraeten, M., Vervaet, C., Remon, J.P., de Beer, T., Lopes, J.A., 2017. Multivariate statistical process control of a continuous pharmaceutical twin-screw granulation and fluid bed drying process. Int. J. Pharm. 528, 242–252. doi:10.1016/j.ijpharm.2017.05.075.

Sun, W., Paiva, A.R.C., Xu, P., Sundaram, A., Braatz, R.D., 2020. Fault detection and identification using Bayesian recurrent neural networks. Comput. Chem. Eng. 141. doi:10.1016/j.compchemeng.2020.106991.

Sundararajan, M., Taly, A., Yan, Q., 2017. Axiomatic Attribution for Deep Networks.

Ueda, R.M., Souza, A.M., 2022. An effective approach to detect the source(s) of out-of-control signals in productive processes by vector error correction (VEC) residual and hotelling's T2 decomposition techniques. Expert Syst. Appl. 187. doi:10.1016/j.eswa.2021.115979.

Vanhatalo, E., Kulahci, M., 2016. Impact of autocorrelation on principal components and their use in statistical process control. Qual. Reliab. Eng. Int. 32, 1483–1500. doi:10.1002/qre.1858.

Vanhatalo, E., Kulahci, M., Bergquist, B., 2017. On the structure of dynamic principal component analysis used in statistical process monitoring. Chemom. Intell. Lab. Syst. 167, 1–11. doi:10.1016/j.chemolab.2017.05.016.

Wang, W., Yang, D., Chen, F., Pang, Y., Huang, S., Ge, Y., 2019. Clustering with orthogonal autoEncoder. IEEE Access 7, 62421–62432. doi:10.1109/ACCESS.2019.2916030.

Wasserman, L., 2004. All of Statistics. Springer, New York, NY doi:10.1007/978-0-387-21736-9.

Yan, W., Guo, P., gong, L., Li, Z., 2016. Nonlinear and robust statistical process monitoring based on variant autoencoders. Chemom. Intell. Lab. Syst. 158, 31–40. doi:10.1016/j.chemolab.2016.08.007.

Yu, J., Zheng, X., Wang, S., 2019. A deep autoencoder feature learning method for process pattern recognition. J. Process Control 79, 1–15. doi:10.1016/j.jprocont.2019.05.002.

Yu, W., Zhao, C., 2020. Robust monitoring and fault isolation of nonlinear industrial processes using denoising autoencoder and elastic net. IEEE Trans. Control Syst. Technol. 28, 1083–1091. doi:10.1109/TCST.2019.2897946.

Zhang, Z., Jiang, T., Li, S., Yang, Y., 2018. Automated feature learning for nonlinear process monitoring – an approach using stacked denoising autoencoder and k-nearest neighbor rule. J. Process Control 64, 49–61. doi:10.1016/j.jprocont.2018.02.004.

Zhang, Z., Jiang, T., Zhan, C., Yang, Y., 2019. Gaussian feature learning based on variational autoencoder for improving nonlinear process monitoring. J. Process Control 75, 136–155. doi:10.1016/j.jprocont.2019.01.008.

Zhou, C., Paffenroth, R.C., 2017a. Anomaly detection with robust deep autoencoders. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York, New York, USA doi:10.1145/3097983.3098052.

Zhou, C., Paffenroth, R.C., 2017b. Anomaly detection with robust deep autoencoders. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York, New York, USA doi:10.1145/3097983.3098052.

Zimmerer, D., Petersen, J., Kohl, S.A.A., Maier-Hein, K.H., 2019. A Case for the Score: Identifying Image Anomalies using Variational Autoencoder Gradients.