*Article*

# A Three-Step Framework for Multimodal Industrial Process Monitoring Based on DLAN, TSQTA, and FSBN

Hao Wu [1,*] , Wangan Fu [2], Xin Ren [1], Hua Wang [1] and Enmin Wang [1]

1    China Huaneng Clean Energy Research Institute, Beijing 102209, China
2    Clean Energy Branch of Huaneng (Zhejiang) Energy Development Co., Ltd., Hangzhou 310011, China
*    Correspondence: h_wu@qny.chng.com.cn

**Abstract:** The process monitoring method for industrial production can technically achieve early warning of abnormal situations and help operators make timely and reliable response decisions. Because practical industrial processes have multimodal operating conditions, the data distributions of process variables are different. The different data distributions may cause the fault detection model to be invalid. In addition, the fault diagnosis model cannot find the correct root cause variable of system failure by only identifying abnormal variables. There are correlations between the trend states of the process variables. If we do not consider these correlations, this may result in an incorrect fault root cause. Therefore, multimodal industrial process monitoring is a tough issue. In this paper, we propose a three-step framework for multimodal industrial process monitoring. The framework aims for multimodal industrial processes to detect the faulty status timely and then find the correct root variable that causes the failure. We present deep local adaptive network (DLAN), two-stage qualitative trend analysis (TSQTA), and five-state Bayesian network (FSBN) to implement fault detection, identification, and diagnosis step by step. This framework can detect the system failure timely, identify abnormal variables, and find the root cause variable and the fault propagation path. The case studies on the Tennessee Eastman simulation and a practical chlorobenzene production process are provided to verify the effectiveness of the proposed framework in multimodal industrial process monitoring.

**Keywords:** industrial safety; multimodality; process monitoring; deep neural network; Bayesian network

## 1. Introduction

The development of process industry has brought convenience to society, but it has also brought many industrial accidents, which endanger our property and life. The safety of process industry has always been a concern of government and community. Chemical industry is one of the most serious industries because it involves lots of hazardous chemicals and chemical processes. Tables 1 and 2 list the major chemical accidents and fatality in the Chinese chemical industry in 2017–2019. According to the analysis, production has the most accidents and fatalities. Because it involves more chemicals, hazardous processes, and staff, major accidents will cause more serious property losses and more casualties.

**Table 1.** The major chemical accidents in Chinese chemical industry in 2017–2019 [1–3].

| Year | Storage | Transportation | Maintenance | Production | Other |
|------|---------|----------------|-------------|------------|-------|
| 2017 | 0 | 1 | 8 | 7 | 1 |
| 2018 | 0 | 0 | 5 | 8 | 0 |
| 2019 | 1 | 1 | 3 | 8 | 0 |
| Total | 1 | 2 | 16 | 23 | 1 |
| Ratio | 2.3% | 4.7% | 37.2% | 53.5% | 2.3% |

**Table 2.** The fatalities of the major chemical accidents in Chinese chemical industry in 2017–2019 [1–3].

| Year | Storage | Transportation | Maintenance | Production | Other |
|------|---------|----------------|-------------|------------|-------|
| 2017 | 0 | 10 | 28 | 36 | 3 |
| 2018 | 0 | 0 | 19 | 70 | 0 |
| 2019 | 78 | 3 | 23 | 44 | 0 |
| Total | 78 | 13 | 70 | 150 | 3 |
| Ratio | 24.8% | 4.1% | 22.3% | 47.8% | 1.0% |

Industrial process monitoring is an important technical method to ensure process safety and product quality. It has important effects on the operation of complex dynamical systems specific to modern industry applications, such as chemical engineering, industrial electronics, business management systems, energy, and public sectors [4]. It has great theoretical and application value for improving the level of smart manufacturing in industry and preventing serious production accidents. The industrial process monitoring method can technically achieve early warning of abnormal situations and assist operators to make timely and reliable response decisions.

Chemical processes include model variable failure, physical device failure, and control system failure (see Figure 1). Model variable failure refers to the fact that some disturbances cause the devices to deviate from the normal status, such as flow, level, temperature, and pressure disturbances. Physical device failure refers to pipeline leakage, pump failure, compressor failure, condenser failure, etc. Control system failure refers to the fact that errors or faults occur in sensors, controllers, actuators, etc. This paper focuses on model variable failure in chemical processes. Industrial process monitoring includes three parts:

(1) Fault detection: detect the faulty status.
(2) Fault identification: identify the abnormal variables.
(3) Fault diagnosis: diagnose the root cause variable and the fault propagation path.
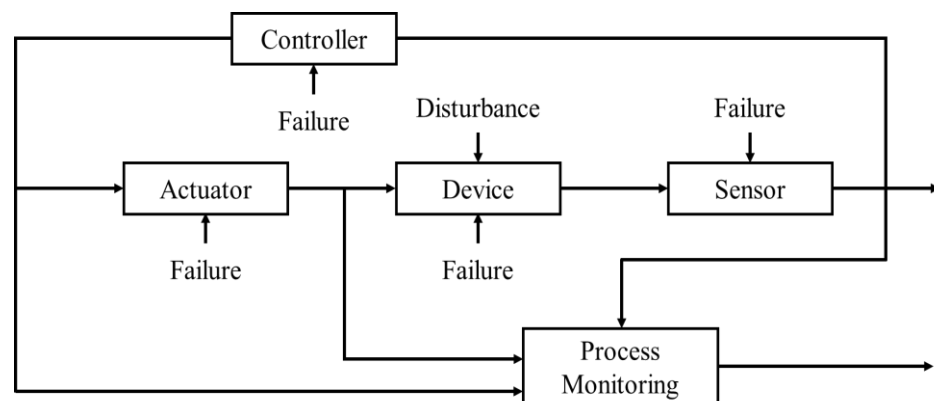


**Figure 1.** Failures in chemical processes.

Because practical industrial processes have multimodal operating conditions, the data distributions of process variables are different. The different data distributions may cause the fault detection to be model invalid. In addition, the fault diagnosis model cannot find the correct root cause variable of system failure by only identifying abnormal variables. There are correlations between the trend states of process variables. If we do not consider these correlations, this may result in the incorrect fault root cause. Therefore, multimodal industrial process monitoring is a tough issue. In this paper, we propose a three-step framework for multimodal industrial process monitoring. We present the deep local adaptive network (DLAN), two-stage qualitative trend analysis (TSQTA), and five-state Bayesian network (FSBN) to implement fault detection, identification, and diagnosis step by step. The framework aims for multimodal industrial processes to detect the faulty status

timely and then to find the correct root variable that causes the failure. It has wide and effective application prospects in practical multimodal industrial process monitoring.

The rest of the paper is organized as follows: Section 2 reviews the relevant literature about fault detection, identification, and diagnosis. Section 3 introduces the basic theory of DLAN, TSQTA, and FSBN. Section 4 presents the three-step framework for multimodal industrial process monitoring in detail. Section 5 shows the case studies of the proposed process monitoring framework, including TEP simulation and a practical chlorobenzene production process. Section 6 summarizes the whole paper.

## 2. Literature Review

### 2.1. Fault Detection

The purpose of fault detection is to detect the system status by using the multivariable time series process data. Fault detection includes qualitative model-based methods, quantitative model-based methods, and data-driven methods [5–7]. With the development and promotion of information technology, the widespread application of distributed control systems (DCSs), advanced process control systems (APCs), and industrial data storage platforms, data-driven methods have become the main research direction in the field of industrial process monitoring.

Data-driven methods include multivariate statistics and machine learning. Multivariate statistical methods map the high-dimensional data into principal component space and residual space and then use Hotelling $T^2$ and squared prediction error (SPE) to estimate the system status. Multivariate statistical analysis methods do not require the acquisition of process mechanism knowledge; they only require the use of historical data to build models [8]. Lu et al. developed two monitoring statistics based on the Wasserstein distance for fault detection [9]. The common methods are principal component analysis (PCA), partial least squares (PLS), independent component analysis (ICA), canonical correlation analysis (CCA), statistical pattern analysis, and kernel based methods [10–13]. Yin et al. compared the multivariate statistical methods for fault detection by using the Tennessee Eastman process (TEP) [14]. The research indicates that the average fault detection rate is only 50~60% for TEP simulation data.

Machine learning methods usually have better performance than multivariate statistical methods. Auto-encoder (AE) as a neural network can map the raw data into low-dimension space and then reconstruct the raw data. Because the modeling of AE is an unsupervised learning procedure and only needs normal data, it has great application prospects in practical industrial processes. So far, AE has been applied in many fault detection fields, such as spacecraft telemetry [15], gear and bearing failure [16], and chemical process failure [17–19].

The traditional methods were designed to operate under the assumptions of independently and identically distributed random variable characteristics of static stationary processes [20]. Fault detection methods of industrial processes under the single stable mode have achieved satisfactory performance. However, because of raw materials, products, markets, and environments, practical industrial processes often shift the operating conditions among several modes. Because several modes have different operating conditions, the data distributions of process variables are different. Multivariate statistics and machine learning both require the same data distribution between modeling and application. To solve this problem, we propose a method based on local adaptive standardization (LAS) and variational auto-encoder bidirectional long short-term memory (VAE-BiLSTM) for the fault detection of multimodal industrial processes in our previous work [21]. This method uses LAS for mapping the data in all the modes into the same feature space, so that the data have the same distribution. Then, it uses VAR-BiLSTM to build and train a fault detection model. It was proven that the LAS-VAE-BiLSTM method can be applied to multimodal industrial processes, and it works especially well even if the operating condition shifts to a new mode that has never occurred in history. In this paper, we call it the deep local

adaptive network (DLAN) and use it as the fault detection method to conduct follow-up process monitoring research.

## 2.2. Fault Identification

The purpose of fault identification is to identify the abnormal variables when the system is faulty. Fault identification can be classified into reconstruction methods, contribution plots, and reconstruction-based contributions [22]. Contribution plots are widely used methods and easy to calculate. Miller et al. used contribution plots for product quality control [23]. MacGregor combined contribution plots with PCA and PLS for online monitoring [24]. Dunia et al. proposed a subspace-based fault reconstruction method for fault identification tasks [25]. Qin et al. defined a $T^2$ contribution that eliminates the cross-talks among variables [26]. Yue et al. proposed a combined index used Hotelling's $T^2$ and SPE [27]. Alcala and Qin proposed a new method for a contribution analysis based on the reconstruction of a fault detection index along the direction of a variable [28].

The aforementioned methods are based on multivariant statistical methods. However, DLAN as a deep neural network is essentially a black box model. Different from multivariant statistics, humans do not totally understand the internal structure of neural networks. Therefore, the interpretability of deep neural networks is relatively weak. In this paper, we develop a contribution-based method by using DLAN for fault identification.

## 2.3. Fault Diagnosis

The purpose of fault diagnosis is to diagnose the system failure when the system is faulty. The present fault diagnosis methods include classification-based methods and inference-based methods. Classification-based methods can directly classify the fault data into specific fault types via a classifier. These methods mainly use process data with fault type labels for building and training artificial neural networks (ANNs). Watanabe and Venkatasubramanian used ANNs for fault diagnosis of chemical processes [29,30]. Xie and Zhang used the deep belief network (DBN) for fault classification and achieved satisfactory diagnosis performance [31,32]. Wu et al. used a convolutional neural network (CNN) for fault classification and further improved the fault diagnosis accuracy [33]. Wang et al. proposed a novel method for rotating machinery on the basis of multisensor data fusion and bottleneck layer optimized convolutional neural network [34]. Li et al. proposed a wavelet transform-assisted CNN for the hypertoxic fluorochemical engineering processes [35]. Wang and Zhang proposed a neural network-based process fault diagnosis system with Andrews plot for information preprocessing to enhance the performance of online process fault diagnosis [36]. Classification-based methods require many fault data with fault type labels for supervised training. Therefore, these methods have some limitations in multimodal industrial processes without fault data. In addition, classification neural networks are vulnerable to attack by adversarial examples. Adversarial examples are created by adding a small amount of noise to an original sample in such a way that no problem is perceptible to humans, yet the sample will be incorrectly recognized by a model [37,38].

Inference-based methods formally describe the connections between process variables, and then they perform fault inference for searching for the fault propagation path and determining the root cause of the system failure. Fault tree analysis (FTA) and signed directed graph (SDG) are the main models for representing the internal structure of the system. Compared with the tree structure of FTA, SDG can accurately describe the causal relationship between variable/variable and variable/fault type by means of a graph structure. Iri et al. were the first to present the SDG model [39]. Then, Shiozaki et al. developed a novel SDG with five state nodes and added the fault occurrence time [40]. Batanov and Cheng developed a Fault Diagnosis Expert System (FDES) for locating the root causes of a set of abnormalities in the ethylene distillation process. [41]. Because SDG is only a qualitative method and cannot meet the need of fault diagnosis system, some quantitative methods combined with SDG have been studied. Vedam and Lee proposed the PCA-SDG

and PLS-SDG methods, which use PCA or PLS for fault detection and then use SDG for fault diagnosis [42,43]. Maurya et al. proposed qualitative trend analysis (QTA) combined with SDG, which can diagnose tiny faults [44]. Because original SDG lacks quantitative information, redundant and spurious results may be produced in fault inference. To solve this problem, Yang et al. developed the probabilistic SDG (PSDG) method [45]. The root cause of the system failure can be determined by calculating the conditional probabilities of the directional edges in a graph. Gharahbagheri et al. proposed a fault diagnosis method by using Kernel PCA combined with the Bayesian network (BN), which achieved satisfactory process monitoring performance [46].

In this paper, we present a two-stage qualitative trend analysis (TSQTA) method for extracting the trend states of local moving window data, transforming the continuous data of abnormal variables into trend state information. Then, we present a five-state Bayesian network (FSBN) method for fault diagnosis. Based on the results of fault detection and identification, the method can search the fault propagation path and infer the root cause variable by using the trend state information of abnormal variables.

## 3. Methodology

In this section, we introduce the basic theory of our proposed methods. DLAN was used to detect the faulty status and identify abnormal variables. TSQTA was used to extract the trend state information of abnormal variables. FSBN was used to search the fault propagation path and infer the root cause of the system failure.

### 3.1. Deep Local Adaptive Network

One industrial process generally experiences several operating modes. If the operating condition shifts to a new mode that has never occurred before, the process monitoring model may fail. DLAN was proposed to solve the multimodal adaptation problem [21]. This method has two parts: LAS and VAE-BiLSTM. LAS is used to preprocess local moving window data, and VAE-BiLSTM is trained to detect the unstable deviation in the local moving window.

#### 3.1.1. LAS for Data Preprocessing

For a training dataset $X \in R^{m \times n}$, the sample in a local moving window at time $i$ is defined as

$$w_i = \begin{bmatrix} x_{i-t+1} & \cdots & x_i \end{bmatrix} = \begin{bmatrix} x^{(1)} \\ \cdots \\ x^{(m)} \end{bmatrix} = \begin{bmatrix} x_{i-t+1}^{(1)} & \cdots & x_i^{(1)} \\ \vdots & \ddots & \vdots \\ x_{i-t+1}^{(m)} & \cdots & x_i^{(m)} \end{bmatrix} \tag{1}$$

where $t$ is the length of the local moving window, m is the number of variables, and n is the sample number of the training dataset. In order not to rely on historical data, the mean and the standard deviation are defined as

$$\hat{w}_i = \begin{bmatrix} y_{i-t+1} & \cdots & y_i \end{bmatrix} = \begin{bmatrix} y^{(1)} \\ \cdots \\ y^{(m)} \end{bmatrix} = \begin{bmatrix} y_{i-t+1}^{(1)} & \cdots & y_i^{(1)} \\ \vdots & \ddots & \vdots \\ y_{i-t+1}^{(m)} & \cdots & y_i^{(m)} \end{bmatrix} \tag{2}$$

$$y_t^{(m)} = \frac{x_t^{(m)} - mean(x^{(m)})}{gmstd^{(m)}(X^{(m)})} \tag{3}$$

$$gmstd^{(m)}\left(X^{(m)}\right) = \sqrt{\frac{n_1(std(X_1^{(m)}))^2 + \ldots + n_p(std(X_p^{(m)}))^2}{n_1 + \ldots + n_p}} \tag{4}$$

where $\hat{w}_i$ denotes the standardized sample of $w_i$, $gmstd(\cdot)$ denotes the global mean standard deviation (gmstd), $n_j$ *(j = 1~p)* is the number of samples under mode *j* in the training dataset,

$X_j$ is the normal data under mode $j$ in the training dataset, and $mean(\cdot)$ and $std(\cdot)$ denote the mean and the standard deviation operations.

Here we assume that normal data in the standardized sample $\hat{w}_i$ follow Gaussian distribution approximately. When the system is faulty, one or several variables will tend to rise or fall in the local moving window. The standardized sample $\hat{w}_i$ of the fault data will deviate from the distribution of the standardized sample of the normal data. Figure 2 shows the comparison of one variable of $\hat{w}_i$ for normal data (yellow) and fault data (red).
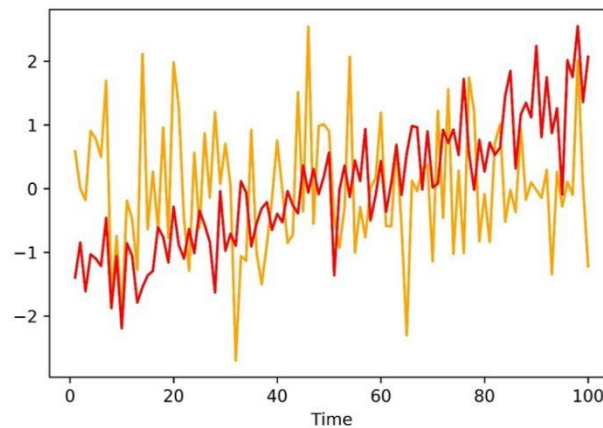


**Figure 2.** The comparison of one variable of $\hat{w}_i$ for normal data (yellow) and fault data (red).

### 3.1.2. VAE-BiLSTM for Fault Detection

After the LAS calculation, VAE-BiLSTM was built and trained by the standardized samples. VAE is composed of an encoder and a decoder [47]. Given an input data $\hat{w}_i \in R^{m \times t}$, an encoder $f_\theta(\cdot)$ calculates a feature vector $h_i$ as a latent representation. Then, a decoder $g_\phi(\cdot)$ reconstructs $\widetilde{w}_i \in R^{m \times t}$ from the latent feature vector $h_i$.

$$h_i = f_\theta(\hat{w}_i) \tag{5}$$

$$\widetilde{w}_i = g_\phi(h_i) \tag{6}$$

Especially, $h_i$ is assumed as a univariate Gaussian distribution, and a "reparameterization trick" is presented as

$$h_{i,l} = \mu_i + \sigma_i \odot \varepsilon_{i,l} \tag{7}$$

where $\varepsilon_{i,l} \sim \mathcal{N}(0, I)$ and $\odot$ denotes an element-wise product. $\mu_i$ and $\sigma_i$ are extracted by an encoder. The loss function of VAE includes the KL-divergence loss and the reconstruction loss as

$$\mathcal{L}(\hat{w}_i) = \lambda \sum_{j=1}^{m} \left( \left( \mu_i^{(j)} \right)^2 + \left( \sigma_i^{(j)} \right)^2 - 1 - log \left( \left( \sigma_i^{(j)} \right)^2 \right) \right) + \left\| \widetilde{w}_i - \hat{w}_i \right\|_2^2 \tag{8}$$

where $\lambda > 0$ is a trade-off parameter for the penalty of the KL-divergence loss.

VAE-BiLSTM is a model that the encoder and the decoder are composed of. BiLSTM is more applicable for time series data. To extract the contextual information efficiently, BiLSTM has a feedforward neural network and a backward neural network. Hence, we used BiLSTM as the layers of VAE to encode and decode the data.

At the modeling stage, the historical data were divided into a training dataset and a validation dataset. After training the VAE-BiLSTM model, we used $\xi_i = \mathcal{L}(\hat{w}_i)$ as the abnormal score to determine whether $w_i$ is abnormal or not. The abnormal scores $\xi_i$ of all the validation data were calculated to determine the upper control limit η based on kernel density estimation. This threshold corresponds to the 99.99% confidence level. For online monitoring, the current process data $w_i$ were standardized by using the LAS method.

Then, the standardized data $\hat{w}_i$ were used to calculate the abnormal score $\xi_i$ by using the VAE-BiLSTM model. If $\xi_i > \eta$, a fault is detected; otherwise, the process is normal.

### 3.1.3. Contributions for Fault Identification

The structure of DLAN is similar to the PCA method. The function of encoder is similar to the dimension reduction from raw data space to principal component space. The function of the decoder is similar to the reconstruction from principal component space to raw data space. Based on the similarity, we applied the contribution calculation to DLAN. Hotelling's $T^2$ and SPE are usually used for PCA monitoring. For the DLAN method, these two indexes can be calculated as

$$T^2 = h_i^T h_i \tag{9}$$

$$SPE = \left\| \widetilde{w}_i - \hat{w}_i \right\|_2^2 \tag{10}$$

$$\widetilde{w}_i = \begin{bmatrix} z_{i-t+1}^{(1)} & \cdots & z_i^{(1)} \\ \vdots & \ddots & \vdots \\ z_{i-t+1}^{(m)} & \cdots & z_i^{(m)} \end{bmatrix} \tag{11}$$

Furthermore, the contribution of each variable was calculated as

$$c_i^{(m)} = \sum_{j=i-t+1}^{i} \left( z_j^{(m)} - y_j^{(m)} \right) \tag{12}$$

$c_i^{(m)}$ denotes the contribution of variable $m$ of $\hat{w}_i$, which is used to measure the contribution of each variable to the system failure.

Only using the contribution plot makes it very difficult to identify abnormal variables. Furthermore, we defined a threshold for each variable to determine which variables are abnormal. Using the contribution thresholds can quantificationally identify the abnormal variables. The contributions $c_i^{(m)}$ of all the validation data were used to determine the upper limit $\psi^{(m)}$ based on kernel density estimation. The threshold corresponds to the 90% confidence level. For online monitoring, if the contribution of variable m of the current data $c_i^{(m)} > \psi^{(m)}$, variable m is abnormal; otherwise, variable m is normal.

### 3.2. Two-Stage Qualitative Trend Analysis

The basic idea of QTA is to extract effective qualitative trend information from quantitative process data by converting time series data into a trend sequence. Trend analysis can qualitatively reflect the operating status of the system and help operators understand the tendency of abnormal variables, so that they can adjust the operating variables to restore the stability of the system as soon as possible [48,49]. Based on the local moving data, we present a novel TSQTA method, which has three parts: trend symbol, trend extraction, and trend recognition.

### 3.2.1. Trend Symbol

For moving window data with a fixed time length, we considered this variable as consisting of two trends in this window. By fitting two linear segments, the trends of this variable can be obtained. This has two advantages: (1) Using two consecutive trends to describe the tendency of variables can clearly represent the dynamic information. (2) Using window data with a fixed time length can align the trend lengths of variables, which can precisely show the correlation between the trends of variables.

In our method, we used 5 symbols to describe the trends of variables, including "normal" (A), "rise" (B), "fall" (C), "step up" (D), and "step down" (E) (see Figure 3). These trends can represent the states of the normal and abnormal variables accurately.
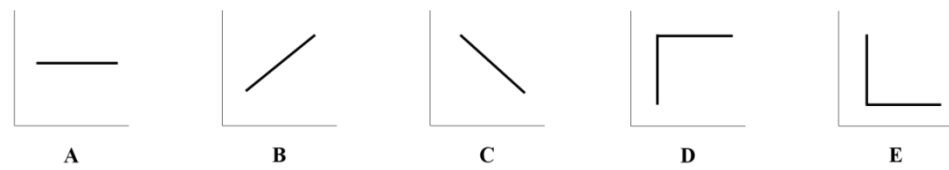
**Figure 3.** Five symbols to describe the trends of variables.

### 3.2.2. Trend Extraction

The purpose of trend extraction is to split the local moving window into two segments and fit them by using linear equation. The linear equation is defined as

$$y(t) = p(t - t_b) + y_b \tag{13}$$

where $t_b$ denotes the beginning of the segment, $y_b$ denotes the measurement of the variable at $t_b$, and $p$ denotes the slope of the trend.

If the length of the segment is $L$, the linear fitting can be implemented as

$$p = \frac{\sum_{i=1}^{L} (t_i - \bar{t})[y(t_i) - \bar{y}]}{\sum_{i=1}^{L} (t_i - \bar{t})^2} \tag{14}$$

Then, we enumerated the split point $t_k = 5, 6, \ldots, t - 5$. $1 \sim t_k$ is the first segment, and $(t_k + 1) \sim t$ is the second segment. Through fitting two segments separately and accumulating the fitting error between the prediction and the measurement, we can find the best split point $t_k^*$ with the minimum fitting error as

$$t_k^* = \min_{t_k} \sum_{j=1}^{t_k} |y(j) - y_1(j)| + \sum_{j=t_k+1}^{t} |y(j) - y_2(j)| \tag{15}$$
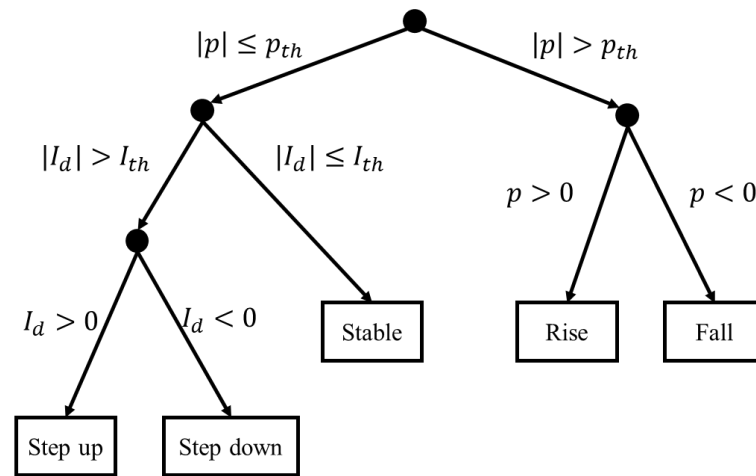
### 3.2.3. Trend Recognition

The purpose of trend recognition is to determine the trend state with two linear segments. Figure 4 shows the linear segments in the local moving window, where $y_e^1$ means the end measurement of the first segment, $y_b^2$ means the beginning measurement of the second segment, and $y_e^2$ means the beginning measurement of the second segment.
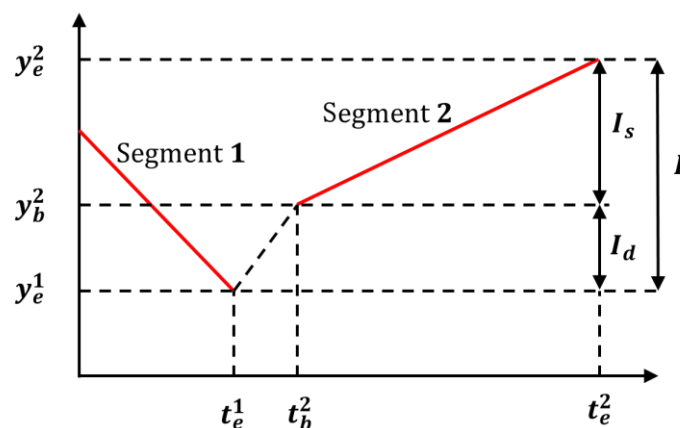


**Figure 4.** The linear segments in the local moving window.

Then, we defined $I_d = y_b^2 - y_e^1$, which measures the continuity of two linear segments. Each variable has two thresholds, $p_{th}^{(m)}$ and $I_{th}^{(m)}$. $p_{th}^{(m)}$ is used to determine whether variable m is stable or not. $I_{th}^{(m)}$ is used to determine whether variable m has step change or not. We can calculate the slopes $p_i$ of the local moving windows and the difference $I_{d,i}$ between two consecutive samples. Then, we used kernel density estimation with 99% confidence

level to calculate $p_{th}^{(m)}$ and used twice the maximum of $I_{d,i}$ to calculate $I_{th}^{(m)}$. The rule of the trend recognition is illustrated in detail in Figure 5.



**Figure 5.** The rule of the trend recognition.

### 3.3. Five-State Bayesian Network

The Bayesian network is a model that regards variables as nodes, uses the causality between variables as edges, and calculates the conditional probabilities to measure the information flow relationship of variables [46,50]. The definition of the Bayesian network is as

$$G = (V, E, \psi) \tag{16}$$

where $V$ denotes the nodes, $E$ denotes the edges between the nodes, and $\psi$ denotes the states of the nodes. The typical Bayesian network used for industrial process monitoring has two states, $\psi \in \{\text{normal}, \text{abnormal}\}$. In practical industrial processes, only using two states cannot describe the trends of variables and the positive/negative effects between variables. For example, with a positive effect A→B, B rises when A rises, and B falls when A falls; with a negative effect C→D, D rises when C falls, and D falls when C rises. Considering the trend states extracted by TSQTA, we developed an FSBN method. Because we focused on the current states of variables, the trend states of the second segments were utilized as the node states of the FSBN model, that is, $\psi \in \{\text{normal}, \text{rise}, \text{fall}, \text{stepup}, \text{stepdown}\}$. Compared with the two-state Bayesian network, the FSBN method can find more accurate root cause variables of the system failure by utilizing the correlations between the trend states of process variables.

#### 3.3.1. Causality Network

The key of FSBN for fault diagnosis is how to build a causality network for an industrial process to express the causal relationship between variables. There are mainly three methods: mathematic method, experiential knowledge method, and data-driven method. The mathematical method requires knowledge of the differential equations of variables. However, complex industrial processes are difficult to be described by mathematical models. The experiential knowledge method relies on the knowledge of experts to determine the causal relationship between variables. It can build a complete and valid graph without any process data, but the modeling work is tough and time-consuming. The data-driven method uses the trend relationship of historical data to automatically calculate the causality of variables. However, it relies on the causal information of historical data, which means if there are no relevant trends of correlated variables, the data-driven method cannot mine the causality even some simple causality.

For a practical industrial process, we suggest that if there are not enough data, we can utilize the process flow diagrams and the control strategy to build the causality network.

If there are sufficient data, we can combine the expert knowledge and the data analysis results to build the causality network accurately.

After modeling the causality network, we used conditional probabilistic tables (CPT) to describe the relationship between the states of variables. The CPT between the nodes can be calculated by maximum likelihood estimation.

### 3.3.2. Fault Inference

Fault inference can be implemented by Bayesian inference. Firstly, we selected an abnormal variable (usually starting at the variable with the largest contribution) and set the actual abnormal state (rise, fall, step up, and step down) 100%. If this variable is a root node or its cause variables are normal, it is the root cause variable of the system failure; otherwise, we should calculate the abnormal state probabilities of its abnormal cause variables by using Bayesian equation. To ensure the correct positive and negative effects of the variables, we should delete the cause variables whose actual abnormal state probabilities are less than 5%. Among the cause variables, the variable with the maximum actual abnormal state probability is the new cause node of the system failure. Then, we set the actual abnormal state of the new cause node to 100%. Like this procedure, we located the abnormal node by following the causality from downstream to upstream, until the current node does not have cause nodes or all cause nodes of the current node are normal. The ending node is the root cause variable of the system failure.

## 4. Framework

The three-step framework for multimodal industrial process monitoring is shown in Figure 6.

Offline modeling.

(1) Collect the historical normal data in one or several operating modes.
(2) Split the data into training data and validation data.
(3) Reshape training data and validation data into local moving window data and implement LAS procedure.
(4) Design a deep neural network as the DLAN model.
(5) Use training data for training the DLAN model and use validation data for calculating the detection threshold $\eta$.
(6) Use the DLAN model to calculate the contributions of variables for validation data and calculate the contribution thresholds $\psi^{(m)}$ of variables.
(7) Use the local moving window data of the raw historical normal data to calculate the slope thresholds $p_{th}^{(m)}$ and the step thresholds $I_{th}^{(m)}$.
(8) Calculate the trend states of variables of the local moving window data by using the TSQTA method.
(9) Use the process knowledge to build causality network of variables as the FSBN model.
(10) Use the trend states to calculate the conditional probabilities of the FSBN model.

Online monitoring.

(1) Collect the online data and reshape them into local moving window data and implement LAS procedure.
(2) Use the DLAN model to calculate the abnormal score $\xi_i$ of the online data. If $\xi_i > \eta$, the system is faulty; otherwise, the system is normal.
(3) If the system is faulty, use the DLAN model to calculate the contributions $c_i^{(m)}$ of the variables. If $c_i^{(m)} > \psi^{(m)}$, variable $m$ is an abnormal variable; otherwise, variable $m$ is a normal variable.
(4) Rank the abnormal variables with $c_i^{(m)}/\psi^{(m)}$ to measure the degree of anomaly.
(5) Calculate the trend states of abnormal variables of the local moving window data by using the TSQTA method.
(6) Update the trend states of abnormal variables in the FSBN model.

(7)  Through fault inference, search the root cause variable of the system failure and the fault propagation path.
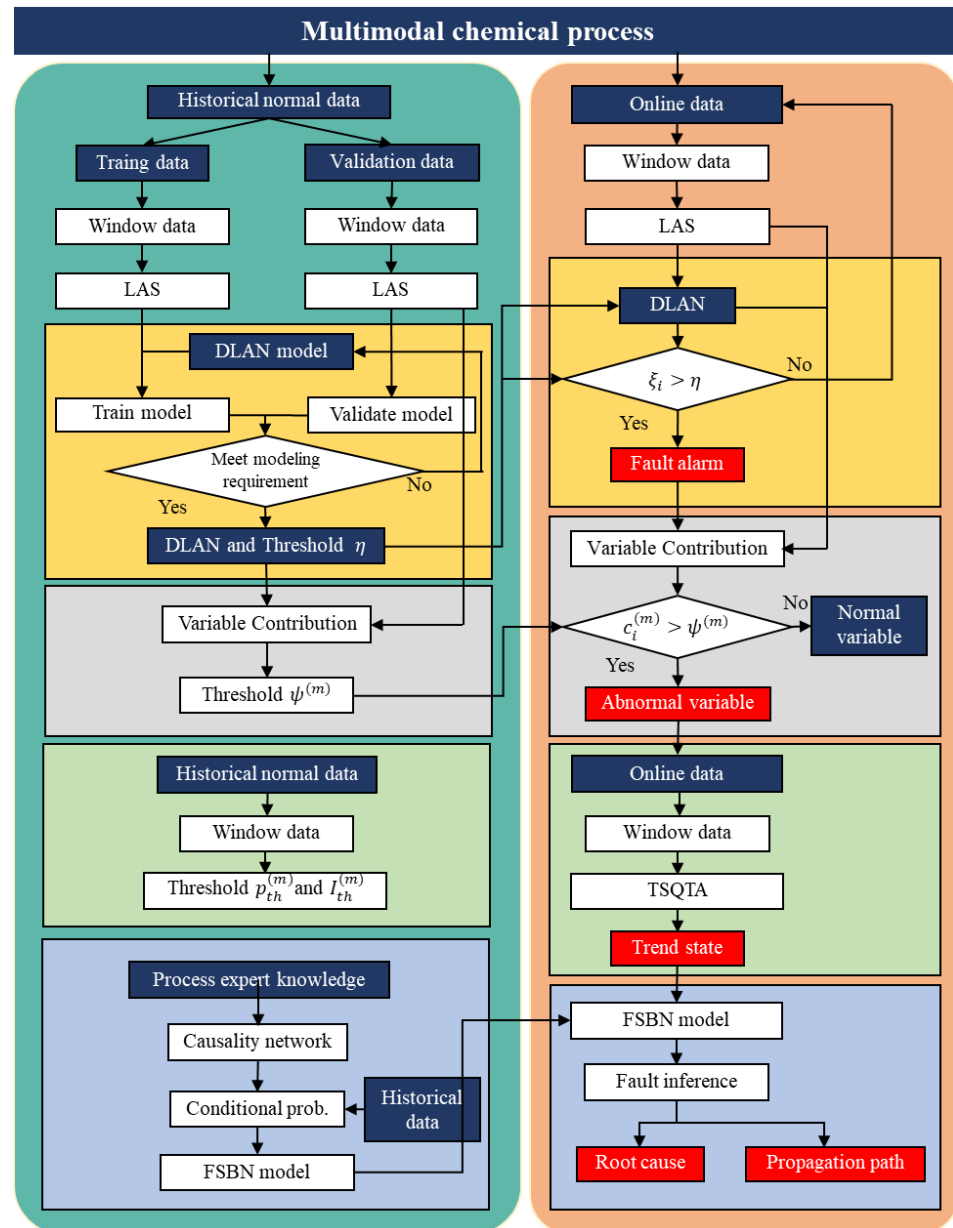


**Figure 6.** The three-step framework for multimodal industrial process monitoring.

## 5. Case Studies

This section shows two case studies of the three-step multimodal industrial process monitoring framework. These two case studies will validate the effectiveness and applicability of our proposed framework.

### 5.1. Tennessee Eastman Process

TEP simulation was developed by Downs and Vogel [51]. The process includes a reactor, a condenser, a compressor, a separator, and a stripper. Then, Ricker designed a control system and presented six operating conditions with different product ratios and different yields for TEP [52,53]. In this case, we utilized the revised version established by Matlab simulink, which was developed by Bathelt [54]. This version includes 30 process measurements, 12 manipulated variables, and 43 component variables. Because

the sampling interval time of component variables is very long in the practical factories, we only used 42 variables here for the experiment. To simulate the multimodal processes, we used the operating conditions of Modes 1, 2, 4, and 5 presented by Ricker. Figures 7 and 8 show the process flow diagram and the control system of TEP simulation.
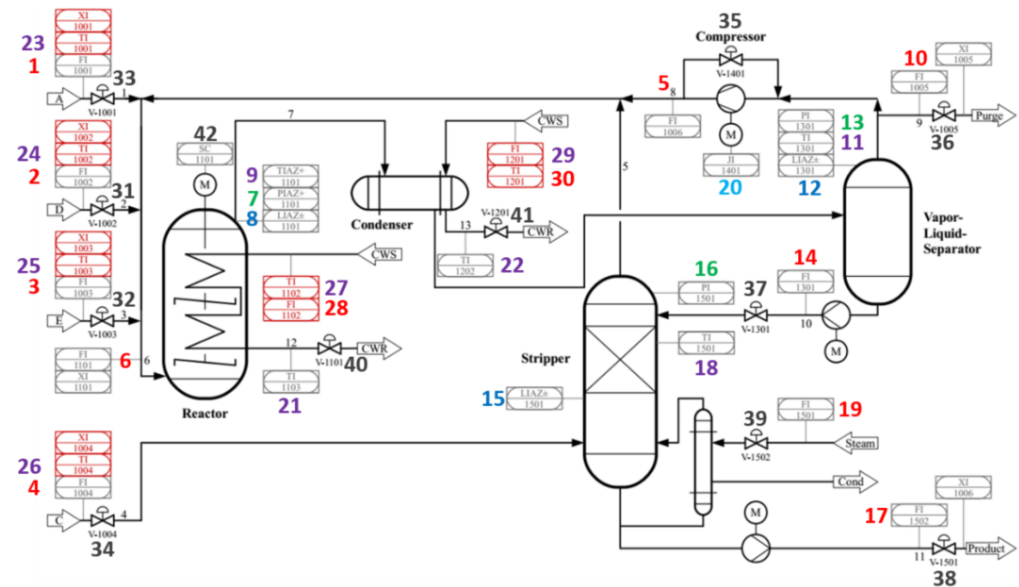


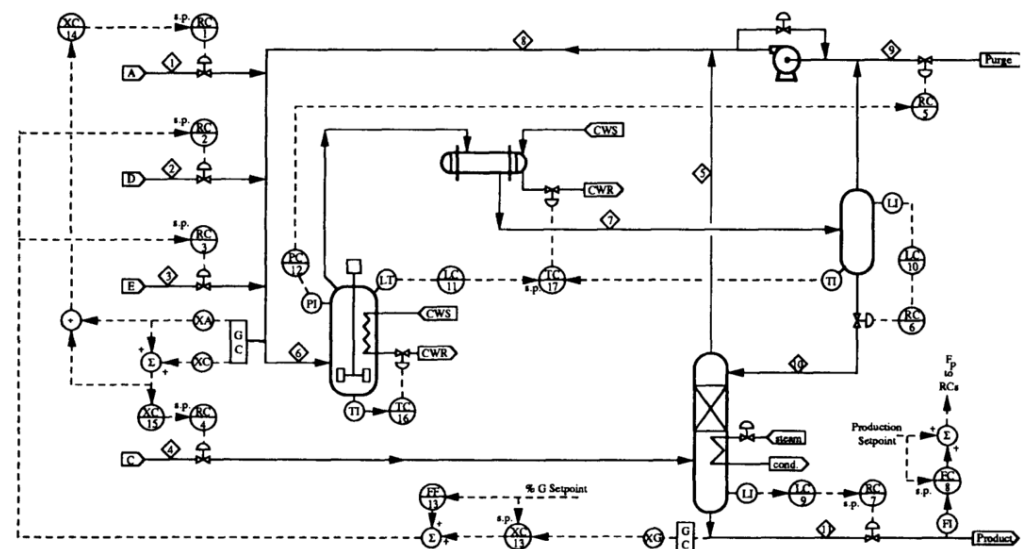**Figure 7.** The process flow diagram of TEP simulation.



**Figure 8.** The control system of TEP simulation.

### 5.1.1. Offline Modeling

For these four modes, the sampling interval time is 1 min with 42 variables. Each mode collects normal data and fault data. Normal data for each mode include 80,000 samples (55.5 days). Fault data for each mode include 28 disturbances as fault types. Each fault type simulates for 10 h. Fault data for each mode include 16,800 samples (11.7 days). We split the 80,000 samples into training/validation/testing datasets with 60%/20%/20%.

For offline modeling, we only used the training and validation normal datasets in Modes 1 and 4. We reshaped the samples into local moving window data with 42 variables for 60 min and implemented the LAS procedure. Then, we designed a DLAN model listed in Table 3.

**Table 3.** The structure of the DLAN model for TEP.

| Layer | Input Channel | Output Channel | INPUT DIM | OUTPUT DIM |
|---|---|---|---|---|
| BiLSTM | 42 | 100 | $42 \times 60$ | $200 \times 60$ |
| Linear/ReLU | 200 | 100 | $200 \times 60$ | $100 \times 60$ |
| BiLSTM | 100 | 50 | $100 \times 60$ | $100 \times 60$ |
| Linear/ReLU | 100 | 50 | $100 \times 60$ | $50 \times 60$ |
| Reparameterization | 50/50 | 50/50 | $50 \times 1$ | $50 \times 1$ |
| BiLSTM | 50 | 100 | $50 \times 60$ | $200 \times 60$ |
| Linear/ReLU | 200 | 100 | $200 \times 60$ | $100 \times 60$ |
| BiLSTM | 100 | 42 | $100 \times 60$ | $84 \times 60$ |
| Linear/ReLU | 84 | 42 | $84 \times 60$ | $42 \times 60$ |

We used the training dataset in Modes 1 and 4 to train the DLAN model with 20 epochs and 0.0001 learning rate. After training, we used the validation dataset in Modes 1 and 4 to calculate the abnormal scores of the samples. We calculated the upper control limit $\eta$ based on kernel density estimation. Then, we used the validation dataset in Models 1 and 4 to calculate the contributions of variables of samples. We calculated the contribution thresholds $\psi^{(m)}$ based on kernel density estimation. Next, we used the training and validation datasets in Modes 1 and 4 to calculate the slope thresholds $p_{th}^{(m)}$ and the step thresholds $I_{th}^{(m)}$ for each variable. Then, we implemented the TSQTA method to calculate the trend states of the training, validation, and fault datasets in Modes 1 and 4.

According to Figures 7 and 8, we established a causality network of TEP through process knowledge. Finally, we utilized the trend states of the training, validation, and fault datasets in Modes 1 and 4 to calculate the conditional probabilities of the causality network for building the FSBN model (see Figure 9).
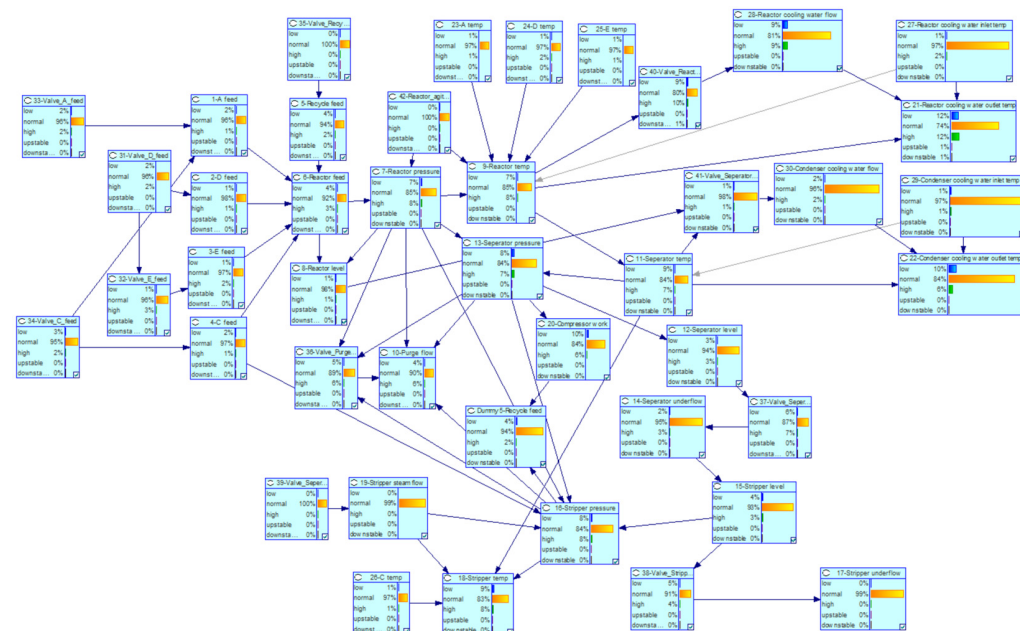


**Figure 9.** The structure of FSBN model.

5.1.2. Mode 2 Fault 3: Step Change of Feed D Temperature

In this case, the temperature of feed D has a step change. Because of the flow controller of the cooling water of the reactor, the temperature of the reactor is stable. For this disturbance, only variable 24 has a step change, and the other variables are normal. Figure 10 shows the detection result of the DLAN model. The red line means the detection threshold $\eta$, and the green line means the time of the disturbance introduction. Before the green

line, the abnormal score is lower than the detection threshold, which means the system is normal. When the disturbance is introduced, the abnormal score immediately exceeds the red line, which means the system is changing. Because the system is still stable due to the flow controller of the cooling water of the reactor, the abnormal score will return to normal.
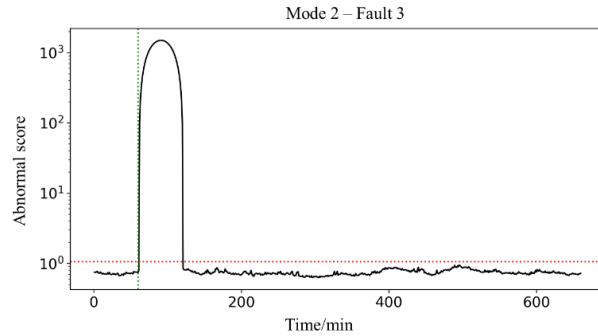


**Figure 10.** The detection result of the DLAN model for Mode 2 fault 3.

When the abnormal score exceeds the red line in 1 min after the disturbance introduction, DLAN will calculate the contributions of variables to identify the abnormal variables (see Table 4). Only variable 24 is identified so that this is the root cause variable. Figure 11a illustrates the linear fitting of variable 24 in 1 min, which shows that the TSQTA method can describe the actual trend states of the process variables.

**Table 4.** The contributions of abnormal variables in 1 min for Mode 2 fault 3.

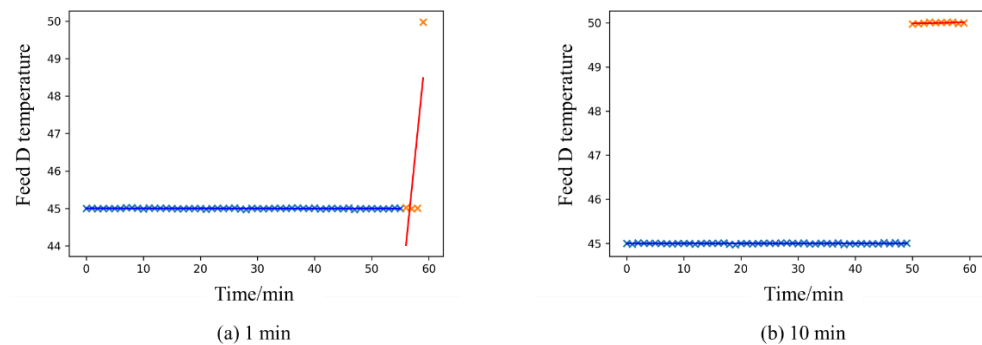| Variable Index | Description | Contribution | First Trend | Second Trend |
|---|---|---|---|---|
| 24 | Feed D Temperature | 2537.8 | Normal | Rise |



**Figure 11.** The linear fitting of variable 24 for Mode 2 fault 3: (**a**) 1 min; (**b**) 10 min.

Table 5 lists the contributions of the abnormal variables in 10 min after the disturbance introduction. The second trend changes to "Step up", which corresponds to the disturbance of step change. Figure 11b illustrates the linear fitting of variable 24 in 10 min, which shows the trend states of variable 24 clearly.

**Table 5.** The contributions of abnormal variables in 10 min for Mode 2 fault 3.

| Variable Index | Description | Contribution | First Trend | Second Trend |
|---|---|---|---|---|
| 24 | Feed D Temperature | 21,721.9 | Normal | Step up |

### 5.1.3. Mode 5 Fault 6: Flow Loss of Feed A

In this case, the flow of feed A is lost and changes to 0. This disturbance cannot be eliminated by the controller system so that it is a dangerous fault. Figure 12 shows the detection result of the DLAN model. The red line means the detection threshold $\eta$, and the green line means the time of the disturbance introduction. When the disturbance is introduced, the abnormal score immediately exceeds the red line, which means the system is changing. Because the system cannot be controlled by the controller system, the detection result will be always abnormal.
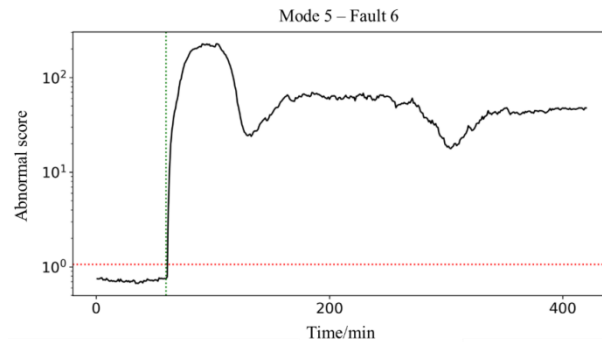


**Figure 12.** The detection result of the DLAN model for Mode 5 fault 6.

When the abnormal score exceeds the red line in 1 min after the disturbance introduction, DLAN will calculate the contributions of variables to identify the abnormal variables (see Table 6). Only variable 1 is identified so that this is the root cause variable.

**Table 6.** The contributions of abnormal variables in 1 min for Mode 5 fault 6.

| Variable Index | Description | Contribution | First Trend | Second Trend |
|---|---|---|---|---|
| 1 | Feed A flow | 13.8 | Normal | Fall |

Table 7 lists the contributions of abnormal variables in 2 min after the disturbance introduction. If we do not use the contribution thresholds in our method, we cannot identify which variables are abnormal (see Figure 13). Through our method, we can identify that variable 33 is abnormal. There are two abnormal variables identified by the DLAN model. Therefore, we should implement the FSBN procedure.

**Table 7.** The contributions of abnormal variables in 2 min after disturbance introduction.

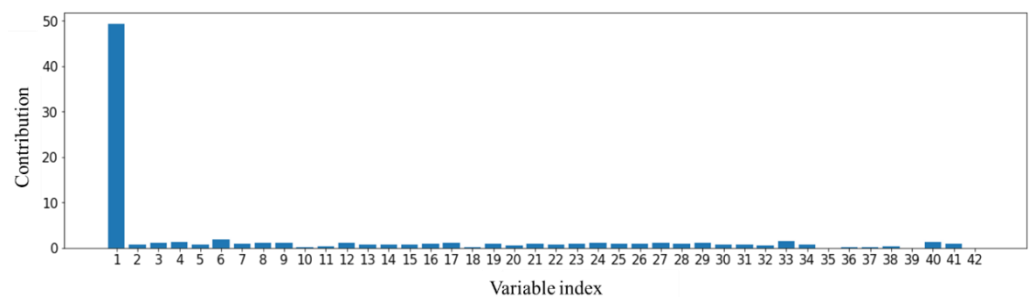| Variable Index | Description | Contribution | First Trend | Second Trend |
|---|---|---|---|---|
| 1 | Feed A flow | 42.7 | Normal | Fall |
| 33 | Feed A valve | 1.3 | Normal | Rise |



**Figure 13.** Contributions of variables in 2 min for Mode 5 fault 6.

Firstly, we started at variable 1, which has the largest contribution. We set the trend state "Fall" of "Feed A flow" as 100% and updated the FSBN model. The trend state "Fall" of "Feed A valve" is 96%. However, the actual trend state of "Feed A valve" is "Rise", the probability of which is 2% < 5%. Therefore, the root cause of this fault is "Feed A flow falls". If we use the two-state Bayesian network, we will find that the root cause is "Feed A valve rises", which is not the real root cause. Therefore, our proposed FSBN method can provide more accurate and more detailed information.

Table 8 lists the contributions of abnormal variables in 10 min after the disturbance introduction. Figure 14 illustrates the linear fitting of abnormal variables, which shows the trend states clearly.

**Table 8.** The contributions of abnormal variables in 10 min after disturbance introduction.

| Variable Index | Description | Contribution | First Trend | Second Trend |
| --- | --- | --- | --- | --- |
| 1 | Feed A flow | 181.8 | Normal | Step down |
| 33 | Feed A valve | 72.6 | Normal | Rise |
| 16 | Stripper pressure | 7.0 | Normal | Fall |
| 7 | Reactor pressure | 6.8 | Normal | Fall |
| 13 | Separator pressure | 6.6 | Normal | Fall |
| 21 | Outlet temperature of cooling water of reactor | 5.3 | Normal | Rise |
| 20 | Compressor power | 2.7 | Normal | Fall |
| 28 | Flow of cooling water of reactor | 2.5 | Normal | Fall |
| 40 | Valve of cooling water of reactor | 2.1 | Normal | Fall |
| 9 | Reactor temperature | 1.6 | Normal | Rise |
| 6 | Reactor feed | 1.6 | Normal | Fall |
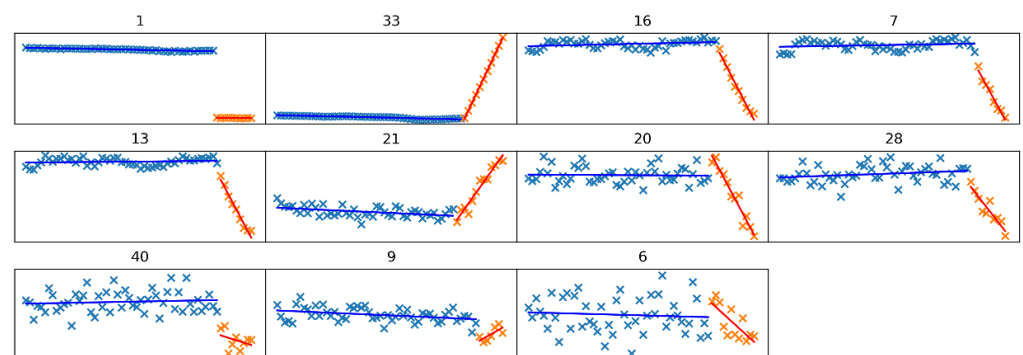


**Figure 14.** The linear fitting of abnormal variables in 10 min for Mode 5 fault 6.

We should implement the FSBN procedure in 10 min. Firstly, we started at variable 1, which has the largest contribution. We set the trend state "Step down" of "Feed A flow" as 100% and updated the FSBN model. The final root cause variable is still "Feed A flow" with "Step down", which corresponds to the actual fault disturbance. Figure 15 illustrates the comparison of the fault inference results of the two-state Bayesian network and our proposed FSBN. It proves that the FSBN method can search the correct root cause variable, which causes the system failure, and describe the detailed fault propagation path.
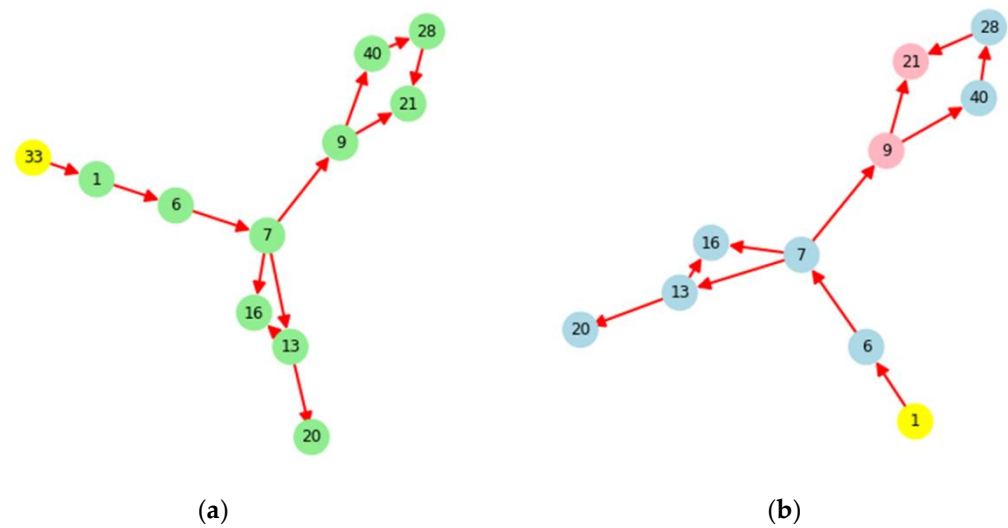
**Figure 15.** The fault inference results in 10 min for Mode 5 fault 6 (yellow: root cause; green: abnormal; red: rise or step up; blue: fall or step down): (**a**) Two-state Bayesian network; (**b**) Five-state Bayesian network.

### 5.2. Chlorobenzene Production Process

Finally, we applied the three-step multimodal industrial process monitoring framework to a practical chlorobenzene production process. This can help operators learn the abnormal changes of the process and the fault propagation of variables. The process includes three parts:

(1) chlorination reaction: Benzene ($C_6H_6$) and chlorine ($Cl_2$) react in 10 chlorinators A~J. The chlorinators include FeS and produce chlorinated benzene ($C_6H_5Cl$) and HCl. $C_6H_5Cl$ enters five intermediate tanks A~E. HCl enters the gas absorption unit.

(2) dehydrochlorination: $C_6H_5Cl$ enters two filters A/B from five intermediate tanks, then enters two dehydrochlorination towers A/B. The bottom liquid of the towers enters the distillation unit. The top gas of the towers enters the gas absorption unit through condensers and recoolers.

(3) gas absorption: The gas from the chlorination reaction and dehydrochlorination units enters two absorption towers from below. The absorption liquid from the circulating tank enters two absorption towers from above. The bottom liquid of the towers re-enters the circulating tank, and the top gas of the towers enters the hydrochloric acid production unit. The liquid of the circulating tank enters the five intermediate tanks A~E.

### 5.2.1. Offline Modeling

We collected the historical data including 136 variables in 1 month from DCS. The sampling time is 1 min, and the data include 43,205 samples. Figure 16 illustrates the flows of $Cl_2$ and $C_6H_6$ for 10 chlorinators. In the process of continuous operation, there are several times to switch the operating conditions, so that this is a multimodal process. There are four modes, which are shown in Figure 16. In the first case, we used the data in Modes 1 and 3 as training and validation data for offline modeling, and then we used the data in Mode 4 as testing data for application. In the second case, we used the data in Mode 2 as training and validation data for offline modeling, and then we used the data in Mode 3 as testing data for application.

(**a**)



(**b**)
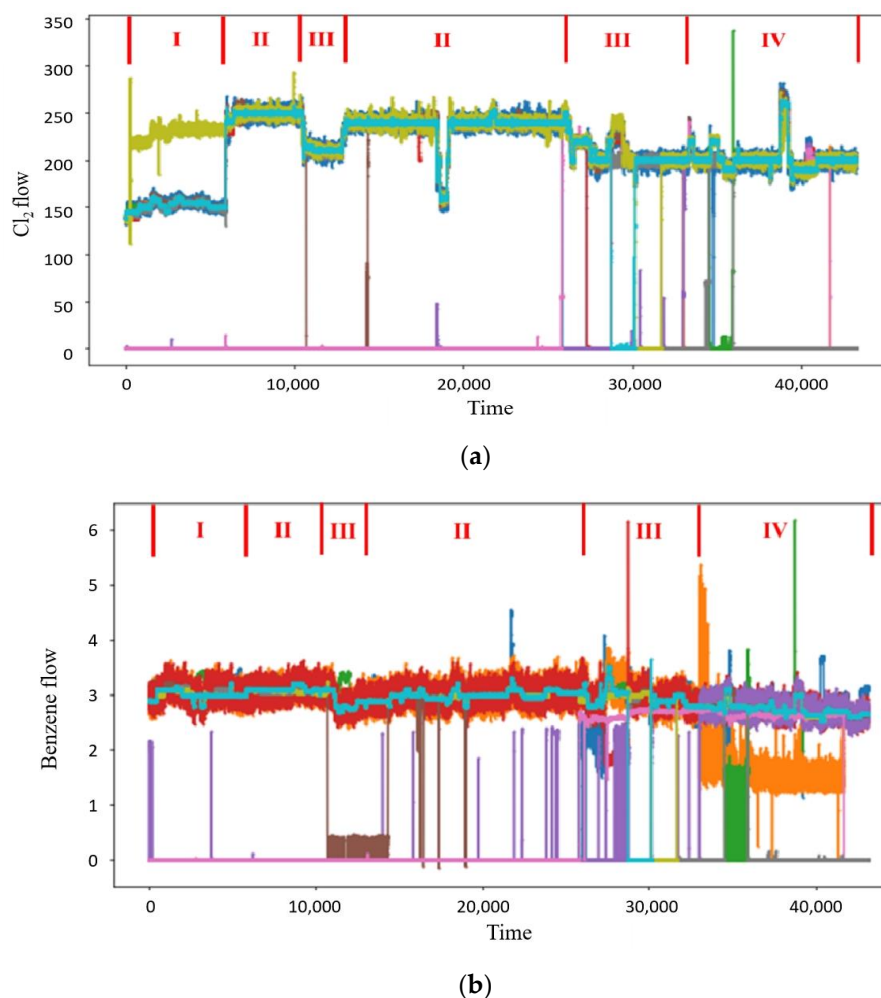
**Figure 16.** The flows of $Cl_2$ (**a**) and benzene (**b**) for 10 chlorinators.

We reshaped the samples into local moving window data with 136 variables for 60 min and implemented the LAS procedure. Then, we designed a DLAN model listed in Table 9.

**Table 9.** The structure of the DLAN model for chlorobenzene production process.

| Layer | Input Channel | Output Channel | Input Dim | Output Dim |
|---|---|---|---|---|
| BiLSTM | 136 | 300 | $136 \times 60$ | $600 \times 60$ |
| Linear/ReLU | 600 | 300 | $600 \times 60$ | $300 \times 60$ |
| BiLSTM | 300 | 150 | $300 \times 60$ | $300 \times 60$ |
| Linear/ReLU | 300 | 150 | $300 \times 60$ | $150 \times 60$ |
| Reparameterization | 50/50 | 150/150 | $150 \times 1$ | $150 \times 1$ |
| BiLSTM | 150 | 300 | $150 \times 60$ | $600 \times 60$ |
| Linear/ReLU | 600 | 300 | $600 \times 60$ | $300 \times 60$ |
| BiLSTM | 300 | 136 | $300 \times 60$ | $272 \times 60$ |
| Linear/ReLU | 272 | 136 | $272 \times 60$ | $136 \times 60$ |

We used the training dataset to train the DLAN model with 50 epochs and 0.0001 learning rate. After training, we used the validation dataset to calculate the abnormal scores of samples. We calculated the upper control limit $\eta$ based on kernel density estimation. Then, we used the validation dataset to calculate the contributions of each variable of samples. We calculated the contribution thresholds $\psi^{(m)}$ based on kernel density estimation. Next, we used the training and validation datasets to calculate the slope thresholds $p_{th}^{(m)}$

and the step thresholds $I_{th}^{(m)}$. Then, we implemented the TSQTA method to calculate the trend states of the training and validation datasets.

We established a causality network of chlorobenzene production process through process knowledge (see Figure 17). We can find that these variables cluster into two parts. The variables of the bottom right part mainly belong to the chlorination reaction unit. The variables of the top left part mainly belong to the dehydrochlorination and gas absorption units. Finally, we utilized the trend states of the training and validation datasets to calculate the conditional probabilities of the causality network for building the FSBN model.
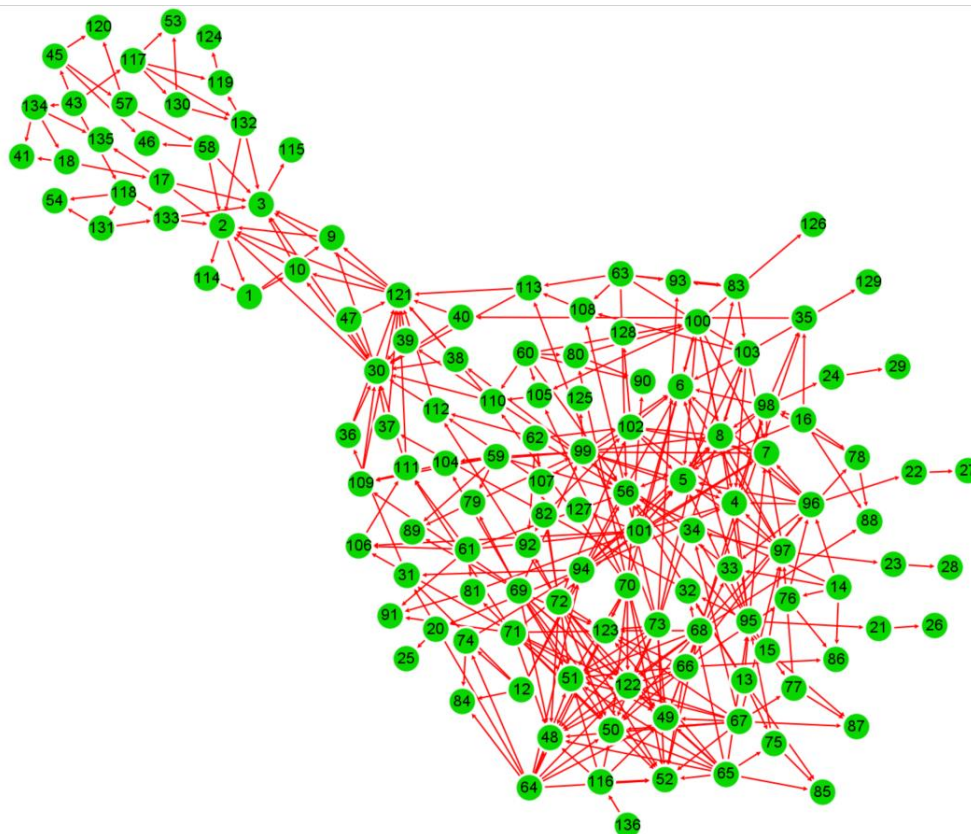


**Figure 17.** Causality network of the chlorobenzene production process.

5.2.2. Modes 1 and 3 for Modeling and Mode 4 for Application

For the data in Mode 4, when t = 710 min, the abnormal score exceeds the detection threshold, which means system failure (see Figure 18).
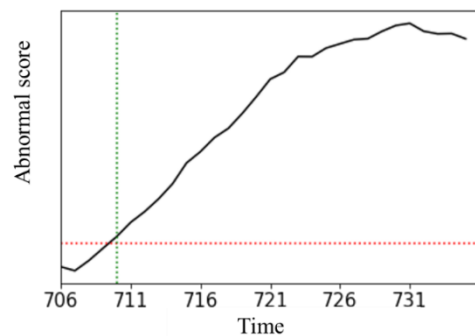


**Figure 18.** Fault detection when t = 710 min in Mode 4.

Then, we calculated the contributions of variables to find abnormal variables and used the TSQTA method to identify their trend states. Through the calculation, we found

39 abnormal variables. Figure 19 illustrates the linear fitting of 16 abnormal variables, which have larger abnormal contributions. We found that these variables actually have abnormal trends. This can prove that our methods can detect the system anomaly and abnormal variables correctly and effectively.
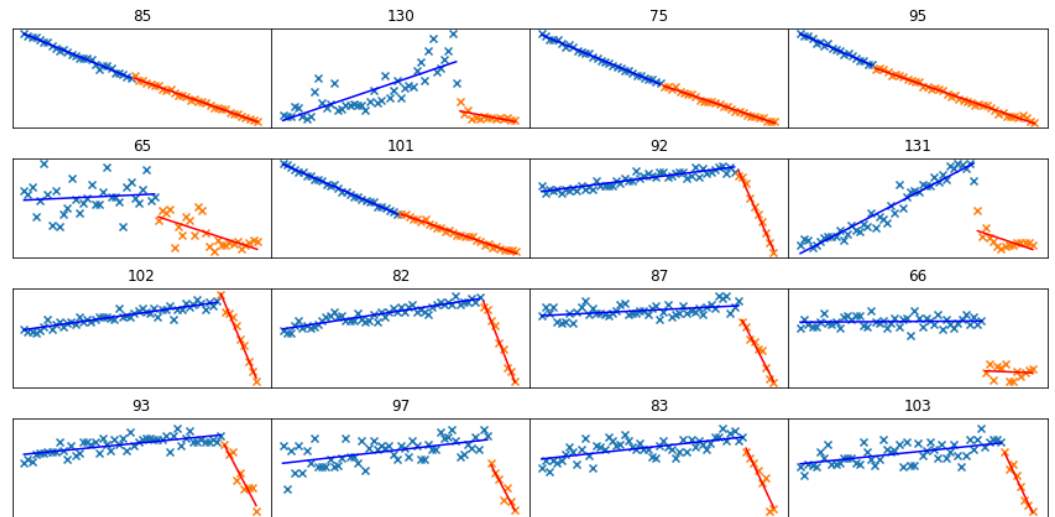


**Figure 19.** The linear fitting of 16 abnormal variables when t = 710 min in Mode 4.

Finally, we used the FSBN model to implement the fault diagnosis. Firstly, we started at variable 85, which has the largest contribution. Through fault inference, we found that variable 65 "Chlorine feed flow of chlorinator B falls" is the root cause. From Figure 19, we found that variable 65 actually has the "Fall" abnormal trend state. Figure 20 shows the fault propagation path when t = 710 min in Mode 4.
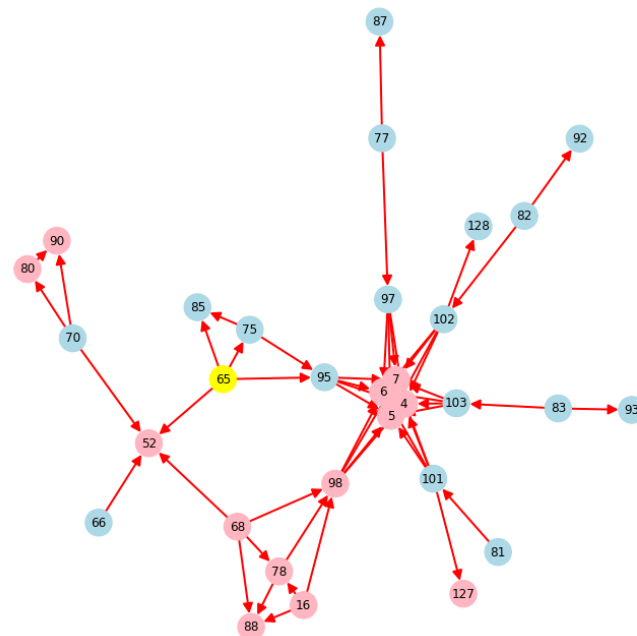


**Figure 20.** The fault propagation path when t = 710 min in Mode 4 (yellow: root cause; red: rise or step up; blue: fall or step down).

### 5.2.3. Mode 2 for Modeling and Mode 3 for Application

For the data in Mode 3, when t = 1236 min, the abnormal score exceeds the detection threshold, which means system failure (see Figure 21).
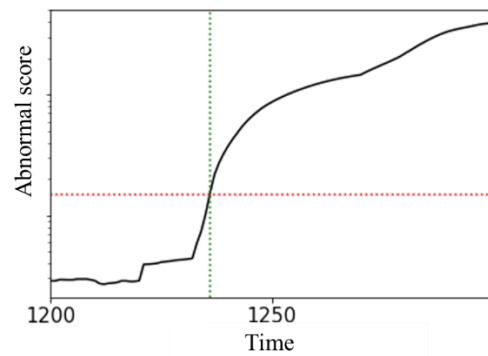
**Figure 21.** Fault detection when t = 1236 min in Mode 3.

Then, we calculated the contributions of the variables to find the abnormal variables and used the TSQTA method to identify their trend states. Through the calculation, we found 23 abnormal variables. Figure 22 illustrates the linear fitting of 16 abnormal variables, which have larger abnormal contributions. We found that these variables actually have abnormal trends.
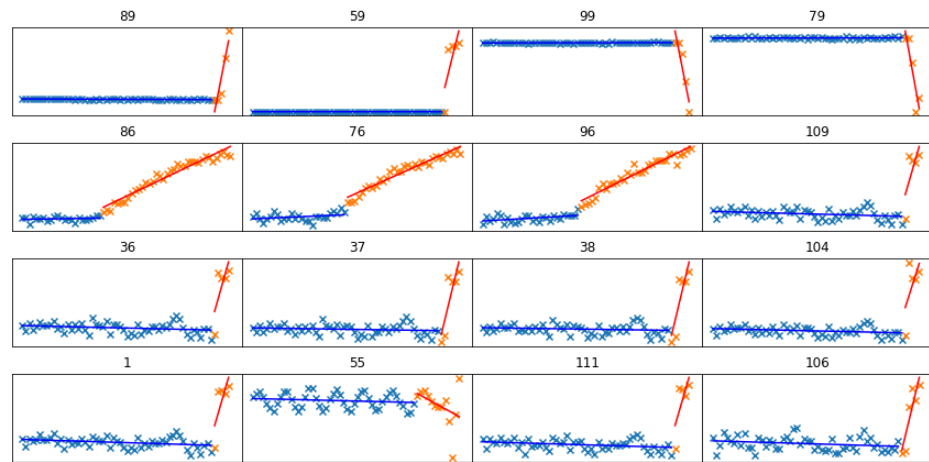


**Figure 22.** The linear fitting of 16 abnormal variables when t = 1236 min in Mode 3.

Finally, we used the FSBN model to implement the fault diagnosis. Firstly, we started at variable 89, which has the largest contribution. Through fault inference, we found that variable 59 "Chlorine feed flow of chlorinator F rises" is the root cause. From Figure 22, we found that that variable 59 actually has the "Rise" abnormal trend state. Figure 23 shows the fault propagation path when t = 1236 min in Mode 3.
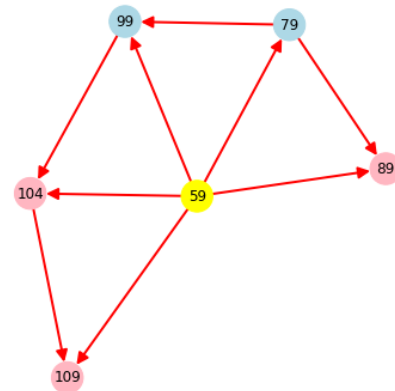


**Figure 23.** The fault propagation path when t = 1236 min in Mode 3 (yellow: root cause; red: rise or step up; blue: fall or step down).

## 6. Conclusions

In this paper, a three-step framework based on DLAN, TSQTA, and FSBN is proposed for multimodal industrial process monitoring. The method uses the DLAN method to detect the system anomaly and calculate the contributions for identifying abnormal variables. Especially, we defined a threshold for each variable to determine which variables are abnormal. Then, the TSQTA method was developed to identify the trend states of abnormal variables. We describe the trends of variables with 5 states, including "normal", "rise", "fall", "step up", and "step down". Finally, based on the trend states of variables, the FSBN method is presented to search the root cause variable of the system anomaly and find the fault propagation path. The proposed framework can adapt to multimodal industrial processes, which can help operators learn about the processes especially when the system is abnormal.

The results of case studies are as follows:

(1)     TEP simulation

In the case of Mode 2, the fault root cause is "Feed D Temperature steps up", which corresponds to the actual failure.

In the case of Mode 5, only using the contribution plot is very difficult to identify the abnormal variable 33. In contrast, using the contribution thresholds can quantificationally identify the abnormal variable 33. Furthermore, we used the TSQTA method to classify the variable trends into five states. By using the typical two-state Bayesian network, the fault root cause is "Feed A valve rises". This is because the flow controller will open the valve to reach the flow setpoint when the flow is reduced or even lost. In contrast, using our FSBN method, we can find the correct fault root cause "Feed A flow steps down".

(2)     Chlorobenzene production process

In the case of Mode 4, we can correctly identify the abnormal variables and find that the root cause is "Chlorine feed flow of chlorinator B falls". In the case of Mode 3, we can correctly identify the abnormal variables and find that the root cause is "Chlorine feed flow of chlorinator F rises". The fault identification and diagnosis results are consistent with the actual trends of the data.

Through the case studies of the TEP simulation and a practical chlorobenzene production process, our proposed framework is well applied to multimodal processes and can detect, identify, and diagnose system failure clearly. It is proven that the framework has wide and effective application prospects in practical multimodal industrial process monitoring.

Further research can be conducted from the following perspectives:

(1)     The deep neural network model used in the modeling needs to be designed for each process. To improve and obtain the optimal model, the Neural Architecture Search (NAS) method can be utilized to automatically search and build the network.

(2)     In the FSBN method, the causality network of the industrial process is established by process knowledge. We can study how to obtain the variable causality automatically and quickly. This will make a fully automatic and intelligent process monitoring approach from modeling to application.

(3)     In the fault inference, FSBN only considers the conditional probabilities between process variables. Next, we can consider to add the time delay and the abnormal amplitude information to the Bayesian network. This will find stronger correlations between process variables and infer more accurate fault root causes.

**Author Contributions:** Conceptualization, W.F. and X.R.; methodology, H.W. (Hao Wu); software, H.W. (Hua Wang); validation, E.W. and X.R.; data curation, E.W.; writing—original draft preparation, H.W. (Hao Wu); writing—review and editing, H.W. (Hao Wu); visualization, H.W. (Hua Wang); supervision, X.R.; project administration, W.F. All authors have read and agreed to the published version of the manuscript.

## References

1. Department of Safety Supervision and Administration of Hazardous Chemicals, Analysis Report on Chemical Accidents in China in 2017. 2018, Beijing, China. Available online: https://www.docin.com/p-2495488982.html (accessed on 1 February 2018).
2. Department of Safety Supervision and Administration of Hazardous Chemicals, Analysis Report on Chemical Accidents in China in 2018. 2019, Beijing, China. Available online: https://www.docin.com/p-2176616404.html (accessed on 1 February 2019).
3. Department of Safety Supervision and Administration of Hazardous Chemicals, Analysis Report on Chemical Accidents in China in 2019. 2020, Beijing, China. Available online: https://www.sohu.com/a/415897039_100022601 (accessed on 1 April 2020).
4. Batanov, D.N.; Cheng, Z. An object-oriented expert system for fault diagnosis in the ethylene distillation process. *Comput. Ind.* **1995**, *27*, 237–249. [CrossRef]
5. Venkatasubramanian, V.; Rengaswamy, R.; Yin, K.; Kavuri, S.N. A review of process fault detection and diagnosis Part I: Quantitative model based methods. *Comput. Chem. Eng.* **2003**, *27*, 293–311. [CrossRef]
6. Venkatasubramanian, V.; Rengaswamy, R.; Kavuri, S.N. A review of process fault detection and diagnosis Part II: Qualitative models and search strategies. *Comput. Chem. Eng.* **2003**, *27*, 313–326. [CrossRef]
7. Venkatasubramanian, V.; Rengaswamy, R.; Kavuri, S.N.; Yin, K. A review of process fault detection and diagnosis part III: Process history based methods. *Comput. Chem. Eng.* **2003**, *27*, 327–346. [CrossRef]
8. Zhao, L.; Huang, X.; Yu, H. Quality-analysis-based process monitoring for multi-phase multi-mode batch processes. *Processes* **2021**, *9*, 1321. [CrossRef]
9. Lu, C.; Zeng, J.; Luo, S.; Cai, J. Detection and isolation of incipiently developing fault using Wasserstein distance. *Processes* **2022**, *10*, 1081. [CrossRef]
10. Severson, K.; Molaro, M.; Braatz, R. Principal component analysis of process datasets with missing values. *Processes* **2017**, *5*, 38. [CrossRef]
11. Ying, Y.; Li, Z.; Yang, M.; Du, W. Multimode operating performance visualization and nonoptimal cause identification. *Processes* **2020**, *8*, 123. [CrossRef]
12. Shao, J.; Zhang, X.; Chen, W.; Shen, X. Improved statistical pattern analysis monitoring for complex multivariate processes using empirical likelihood. *Processes* **2020**, *8*, 1619. [CrossRef]
13. Fang, H.; Tao, W.; Lu, S.; Luo, Z.; Wang, Y.; Xue, Y. Nonlinear dynamic process monitoring based on two-step dynamic local kernel principal component analysis. *Processes* **2022**, *10*, 925. [CrossRef]
14. Yin, S.; Ding, S.X.; Haghani, A.; Hao, H.; Zhang, P. A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process. *J. Process Control.* **2012**, *22*, 1567–1581. [CrossRef]
15. Sakurada, M.; Yairi, T. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, MLSDA, Gold Coast, Australia, 2 December 2014; pp. 4–11.
16. Qu, Y.; He, M.; Deutsch, J.; He, D. Detection of pitting in gears using a deep sparse autoencoder. *Appl. Sci.* **2017**, *7*, 515. [CrossRef]
17. Cheng, F.; He, Q.P.; Zhao, J. A novel process monitoring approach based on variational recurrent autoencoder. *Comput. Chem. Eng.* **2019**, *129*, 106515. [CrossRef]
18. Heo, S.; Lee, J. Statistical process monitoring of the tennessee eastman process using parallel autoassociative neural networks and a large dataset. *Processes* **2019**, *7*, 411. [CrossRef]
19. Li, N.; Shi, H.; Song, B.; Tao, Y. Temporal-spatial neighborhood enhanced sparse autoencoder for nonlinear dynamic process monitoring. *Processes* **2020**, *8*, 1079. [CrossRef]
20. Reis, M.; Gins, G. Industrial process monitoring in the big data/industry 4.0 era: From detection, to diagnosis, to prognosis. *Processes* **2017**, *5*, 35. [CrossRef]
21. Wu, H.; Zhao, J. Self-adaptive deep learning for multimode process monitoring. *Comput. Chem. Eng.* **2020**, *141*, 107024. [CrossRef]
22. Qin, S.J. Survey on data-driven industrial process monitoring and diagnosis. *Annu. Rev. Control.* **2012**, *36*, 220–234. [CrossRef]
23. Miller, P.; Swanson, R.E.; Heckler, C.E. Contribution plots: A missing link in multivariate quality control. *Appl. Math. Comput. Sci.* **1998**, *8*, 775–792.
24. MacGregor, J.; Kourti, T. Statistical process control of multivariate processes. *Control Eng. Pract.* **1995**, *3*, 403–414. [CrossRef]
25. Dunia, R.; Qin, S. Subspace approach to multidimensional fault identification and reconstruction. *AIChE J.* **1998**, *44*, 1813–1831. [CrossRef]
26. Qin, S.; Valle, S.; Piovoso, M. On unifying multiblock analysis with application to decentralized process monitoring. *J. Chemometrics.* **2001**, *15*, 715–742. [CrossRef]

27.  Yue, H.; Qin, S. Reconstruction-based fault identification using a combined index. *Ind. Eng. Chem. Res.* **2001**, *40*, 4403–4414. [CrossRef]
28.  Alcala, C.; Qin, S. Reconstruction-based contribution for process monitoring. *Automatica* **2009**, *45*, 1593–1600. [CrossRef]
29.  Watanabe, K.; Hirota, S.; Hou, L.; Himmelblau, D.M. Diagnosis of multiple simultaneous fault via hierarchical artificial neural networks. *AIChE J.* **1994**, *40*, 839–848. [CrossRef]
30.  Venkatasubramanian, V.; Chan, K. A neural network methodology for process fault diagnosis. *AIChE J.* **1989**, *35*, 1993–2002. [CrossRef]
31.  Xie, D.; Bai, L. A hierarchical deep neural network for fault diagnosis on Tennessee-Eastman process. In Proceedings of the 14th IEEE International Conference on Machine Learning and Applications, ICMLA, Miami, FL, USA, 9–11 December 2015; pp. 745–748.
32.  Zhang, Z.; Zhao, J. A deep belief network based fault diagnosis model for complex chemical processes. *Comput. Chem. Eng.* **2017**, *107*, 395–407. [CrossRef]
33.  Wu, H.; Zhao, J. Deep convolutional neural network model based chemical process fault diagnosis. *Comput. Chem. Eng.* **2018**, *115*, 185–197. [CrossRef]
34.  Souza, M.; da Costa, C.; Ramos, G.; Righi, R. A feature identification method to explain anomalies in condition monitoring. *Comput. Ind.* **2021**, *133*, 103528. [CrossRef]
35.  Li, X.; Zhou, K.; Xue, F.; Chen, Z.; Ge, Z.; Chen, X.; Song, K. A wavelet transform-assisted convolutional neural network multi-model framework for monitoring large-scale fluorochemical engineering processes. *Processes* **2020**, *8*, 1480. [CrossRef]
36.  Wang, S.; Zhang, J. An intelligent process fault diagnosis system based on neural networks and Andrews plot. *Processes* **2021**, *9*, 1659. [CrossRef]
37.  Kwon, H.; Lee, S. Ensemble transfer attack targeting text classification systems. *Comput. Secur.* **2022**, *117*, 102695. [CrossRef]
38.  Kwon, H.; Lee, S. Textual Adversarial training of machine learning model for resistance to adversarial examples. *Secur. Commun. Netw.* **2022**, *2022*, 4511510. [CrossRef]
39.  Iri, M.; Aoki, K.; Oshima, E.; Matsuyama, H. An algorithm for diagnosis of system failures in the chemical process. *Comput. Chem. Eng.* **1979**, *3*, 489–493. [CrossRef]
40.  Shiozaki, J.; Matsuyama, H.; Tano, K.; Oshima, E. Fault diagnosis of chemical processes by the use of signed directed graphs: Extension to five-range patterns of abnormality. *Int. J. Chem. Eng.* **1985**, *25*, 651–659.
41.  Chen, K.Y.; Chen, L.S.; Chen, M.C.; Lee, C.L. Using SVM based method for equipment fault detection in a thermal power plant. *Comput. Ind.* **2011**, *62*, 42–50. [CrossRef]
42.  Vedam, H.; Venkatasubramanian, V. Signed digraph based multiple fault diagnosis. *Comput. Chem. Eng.* **1997**, *21*, 655–660. [CrossRef]
43.  Lee, G.; Tosukhowong, T.; Lee, J.; Han, C. Fault diagnosis using the hybrid method of signed digraph and partial least squares with time delay: The pulp mill process. *Ind. Eng. Chem. Res.* **2006**, *45*, 9061–9074. [CrossRef]
44.  Maurya, M.; Rengaswamy, R.; Venkatasubramanian, V. A signed directed graph and qualitative trend analysis-based framework for incipient fault diagnosis. *Chem. Eng. Res. Des.* **2007**, *85*, 1407–1422. [CrossRef]
45.  Yang, F.; Xiao, D. Probabilistic signed directed graph and its application in hazard assessment. In Proceedings of the 5th International Symposium on Safety Science and Technology, Shanghai, China, 24–27 October 2006; Volume 6, pp. 111–115.
46.  Gharahbagheri, H.; Imtiaz, S.; Khan, F. Root cause diagnosis of process fault using kpca and Bayesian network. *Ind. Eng. Chem. Res.* **2017**, *56*, 2054–2070. [CrossRef]
47.  Kingma, D.P.; Welling, M. Auto-encoding variational Bayes. In Proceedings of the International Conference on Learning Representations ICLR 2014, Banff, Canada, 14–16 April 2014.
48.  Janusz, M.; Venkatasubramanian, V. Automatic generation of qualitative descriptions of process trends for fault detection and diagnosis. *Eng. Appl. Artif. Intell.* **1991**, *4*, 329–339. [CrossRef]
49.  Charbonnier, S.; Garcia-Beltan, C.; Cadet, C.; Gentil, S. Trends extraction and analysis for complex system monitoring and decision support. *Eng. Appl. Artif. Intell.* **2005**, *18*, 21–36. [CrossRef]
50.  Amin, M.T.; Khan, F.; Imtiaz, S. Fault detection and pathway analysis using a dynamic bayesian network. *Chem. Eng. Sci.* **2019**, *195*, 777–790. [CrossRef]
51.  Downs, J.; Vogel, E. A plant-wide industrial process control problem. *Comput. Chem. Eng.* **1993**, *17*, 245–255. [CrossRef]
52.  Ricker, N. Optimal steady state operation of the tennessee eastman challenge process. *Comput. Chem. Eng.* **1995**, *19*, 949–959. [CrossRef]
53.  Ricker, N. Decentralized control of the tennessee eastman challenge process. *J. Process Control* **1996**, *6*, 205–221. [CrossRef]
54.  Bathelt, A.; Ricker, N.; Jelali, M. Revision of the tennessee eastman process model. *IFAC-PapersOnLine* **2015**, *48*, 309–314. [CrossRef]