

Research Article

Fault Diagnosis and Prediction of Continuous Industrial Processes Based on Hidden Markov Model-Bayesian Network Hybrid Model

Jiarula Yasenjiang ^(b),¹ Chenxing Xu ^(b),¹ Shengpeng Zhang ^(b),² and Xin Zhang ^(b),²

¹Department of Industrial Engineering, Institute of Mechanical Engineering, Xinjiang University, Urumqi 830017, China ²China National Petroleum Corporation Western Drilling and Engineering Research Institute, Urumqi 830002, China

Correspondence should be addressed to Chenxing Xu; xu2644026687@163.com

Received 25 April 2022; Revised 25 August 2022; Accepted 29 September 2022; Published 18 November 2022

Academic Editor: Jitendra Kumar

Copyright © 2022 Jiarula Yasenjiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Hidden Markov models (HMMs) have been recently used for fault detection and prediction in continuous industrial processes; however, the expected maximum (EM) algorithm in the HMM has local optimality problems and cannot accurately find the fault root cause variables in complex industrial processes with high-dimensional data and strong variable coupling. To alleviate this problem, a hidden Markov model-Bayesian network (HMM-BN) hybrid model is proposed to alleviate the local optimum problem in the EM algorithm and diagnose the fault root cause variable. Firstly, the model introduces expert empirical knowledge for constructing BN to accurately diagnose the fault root cause variable. Then, the EM algorithm is improved by sequential and parallel learning to alleviate the initial sensitivity and local optimum problems. Finally, the log-likelihood estimates (LL) calculated by the improved hidden Markov model provide empirical evidence for the BN and give fault detection, prediction, and root cause variable detection results based on information about the similar increasing and decreasing patterns of LL for the training data and the online data. Combining the Tennessee Eastman (TE) process and the continuously stirred tank reactor (CSTR) process, the feasibility and effectiveness of the model are verified. The results show that the model can not only find the fault in time but also find the cause of the fault accurately.

1. Introduction

The rapid development of the social economy and the continuous progress of science and technology, as well as the continuous improvement of people's requirements for product quality, system performance, and economy, have prompted the modern industrial process to show an increasingly complex trend in terms of structure and automation. The increasing complexity of systems leads to reliability and safety issues becoming the most important aspects of system design. It is imperative to understand and optimize the coupling of industrial processes through reliability and safety analysis, thus reducing production costs as much as possible on the premise of ensuring the safety of industrial processes. The failure or unreliability of complex

industrial processes will have a great negative impact on social stability and economic development. For example, in June 2000, the explosion at the A. L Ahmedi refinery in Kuwait caused a loss of more than 100 million US dollars [1]. The reliability and safety analysis of the industrial process is conducive to clearly grasping the production status of the industrial process. Crucially, the information obtained through analysis is beneficial to equipment maintenance and helps prevent failures or safety accidents.

In order to reduce or avoid industrial process failures, improve the safety and reliability of industrial processes, and promote the operation and development of enterprises, integrated fault diagnosis, prediction, and health management technology has gained increasing attention and application [2]. The technology includes detecting faults and giving early warning, detecting faults and finding out fault variables for isolation, identifying faults, and determining the size and type of faults.

The data-driven fault detection method has been proven to be effective [3-7]. For example, principal component analysis (PCA), partial least squares (PLS), canonical correlation analysis (CCA), and independent component analysis (ICA), the basic methods of machine learning, have been deeply promoted and studied [8-11]. Kernel principal component analysis (KPCA) methods are proposed to deal with the nonlinear problems of industrial processes [12]. Despite the promising performance of the KPCA method in nonlinear processes, as the complexity of industrial processes increases, KPCA has emerged with high computational costs and memory storage problems. An improved kernel method is proposed to improve the robustness of the KPCA method [13]. Dynamic principal component analysis (DPCA) and dynamic kernel principal component analysis (DKPCA) are proposed for the dynamic process with autocorrelation of time series [12, 14]. A recursive or window principal component analysis method is proposed for timevarying industrial processes [15]. A distributed principal component analysis model is proposed to solve the problem that industrial processes contain a large number of variables [16, 17]. The above data-driven fault detection methods all have their own advantages, and they are applied by training on and learning a large number of datasets.

However, the abovementioned models or methods have challenges in complex continuous fault diagnosis; for example, traditional models such as PCA, PLS, KPCA, and support vector machines (SVMs) do not deal with timeseries industrial process fault detection, and methods such as DPCA, DKPCA, and dynamic independent component analysis (DICA) can deal with time-series between data by means of augmented matrices or vectors, but these methods do not have an autocorrelation of the feature components. The existence of autocorrelation in feature components and model residuals can also have an impact on the use of monitoring statistics. At the same time, this type of method may not work with complex, large-scale industrial process data. To adapt to the increasing complexity of industrial process fault detection, deep learning (DL) methods such as deep migration networks (DTN) and stacked auto-encoders (SAEs) have recently been proposed. Aiming at the unsupervised self-reconstruction of SAE in the pretraining stage, which cannot ensure the relevance of deep features with fault types, a stacked supervised automatic encoder is proposed to pretrain the deep network [18]. The method gradually learns advanced fault-related features from the original input data by stacking multiple supervised automatic encoders in layers to improve the classification accuracy of the classifier. The DTN has alleviated the negative impact of each batch of samples not reflecting the overall distribution of the entire dataset, and given the physicochemical nature of industrial chemical process variables, the features extracted from these process variables also contribute differently to the domain adaptation in the DTN. A linear discriminant analysis (LDA)-based DTN is proposed for fault classification of chemical processes. The model determines the degree of influence of each variable on the source and target domain samples by introducing the LDA method and then designs a weighted maximum mean discrepancy (MMD) for domain adaptation to improve the generalisation performance and classification accuracy of the model [19]. Deep learningbased models have demonstrated powerful data processing capabilities and have shown high fault detection accuracy in experiments. However, the root cause variables of highly coupled, complex industrial process faults remain difficult to identify.

Hidden Markov model (HMM) has received a lot of attention to advance the development of time-series fault detection. HMM is a particularly famous directed graph structure that is mainly used for modeling time-series data. The earliest application of HMM to fault detection was based on the combination of a pattern recognition system and HMM [20]. The HMM has received the attention of researchers since it was used for process monitoring. To improve the fault detection accuracy of the HMM method, a classification system that combines stochastic resonance and HMM is proposed [21]. The advantages of the HMM method of time-series modeling are exploited in combination with goodness-of-fit tests. Online data is used for the detection of industrial process conditions [22]. Targeting the prediction of remaining useful life (RUL), it proposed a prediction method for remaining useful life on the basis of HMM and Prognostics and Health Management (PHM). The prediction process is divided into three parts, including offline modeling, online state estimation, and online prediction. As part of offline modeling, HMM and PHM are established to map the entire degradation path. In the process of operation, the degradation state of the object is estimated in real time. Once the final degradation state is reached, they extract the degradation features and obtain the survival function through the fitted PHM. The proposed methods present higher accuracy than traditional methods [23]. The studies confirm the role of HMM in the field of fault detection and prediction. In the field of complex system diagnosis and prediction, the combination of HMM and Bayesian network (BN) has attracted the attention of some researchers. However, the combination of HMM and BN still faces some challenges. Rebello et al. presented a new method to evaluate the functional reliability of complex industrial systems running in steady state [24]. They used HMM to map continuous data to unobservable state probabilities. DBN found the posterior state probability of the system by considering the dependencies between components in the system. They believed that the degradation of the main components of the system would cause system failure. However, in fact, the cause of failure in complex systems was not just degradation. For the diagnosis and prediction of complex systems, Don and Khan proposed a new detection method by combining HMM and BN [25, 26]. They divided the measured data into three parts: the upper and lower parts are related to the data of dangerous areas, and the data in the middle part represent normal data. They used the average value of logarithmic likelihood (LL) estimation to find the most similar sequence for diagnosis and prediction. However, this method is not very reliable. The reason is that the

detected data may contain a variety of variables, some of which can be partitioned to represent hazards or safety, while special variables such as vibration or chemical reaction temperature cannot be partitioned by this method. The smaller the vibration, the more stable the system is. Some special chemical reactions require extremely high or low temperatures. It is unreasonable to partition the data for diagnosis and prediction using this method.

The BN, also known as the belief network or directed acyclic graph model, is a probability graph model. A directed acyclic graph consists of nodes representing variables and directed edges connecting these nodes. BN can represent the causal relationship between variables or components in a concept or probability model diagram. Although this feature cannot solve all problems, BN is still superior to other classical methods in the applicable range [27]. For example, Liu et al. [28] proposed to use BN and a genetic algorithm (GA) to assemble parts and to reflect the causality between parts through BN so as to obtain relatively high-quality product assembly with low-quality parts. Considering the diagnosis results of complex mechanical systems are influenced by the strong dependence of wear among components, which leads to difficult diagnosis and identification of fault modes, Pang et al. propose a BN model for fault diagnosis of lock mechanisms based on degraded data [29]. The BN probability diagram shows the causal relationship between each variable and each component, and the model shows a good diagnostic effect.

Recently, hybrid modeling technology has been proposed [30, 31]. An important idea of hybrid modeling is to adopt various models or methods to represent various relationships among variables. Therefore, hybrid modeling technology has attracted considerable attention in the field of process modeling and monitoring. Inspired by the idea of mixed modeling and causality, this paper proposes a hybrid model of a hidden Markov-Bayesian network (HMM-BN), upgrading the HMM to ameliorate its robustness. When analyzing the reliability and safety of complex industrial processes, this model represents the causal relationship between variables or components of complex industrial processes in the form of probability diagrams. It solves the shortcomings of the abovementioned methods.

In this study, the hybrid HMM-BN model is proposed based on the deficiencies of the fault diagnosis and prediction model mentioned above. By introducing experts' experience and knowledge, incremental and decremental patterns of log-likelihood estimates, improving the EM algorithm, and combining the BN, it aims to improve the robustness of the HMM-BN model and expand its scope of application in fault detection and prediction of complex industrial processes. The working principle of the HMM-BN hybrid model is to calculate the log-likelihood value through HMM training. It takes advantage of the improved EM algorithm to find globally three adjacent log-likelihood values with similar increasing/decreasing ranges, the average value, and the standard deviation of three adjacent loglikelihood values globally, so as to enhance the accuracy of model fault detection and prediction. The BN is constructed based on experts' experience and knowledge. HMM training

results and log-likelihood estimates are used to update the BN, and the root cause variables of the faults were eventually identified based on the percentage change in probability of each variable. The effectiveness of the HMM-BN hybrid model is illustrated by the Tennessee Eastman (TE) process and continuously stirred tank reactor (CSTR) process examples.

The structure of this paper is as follows: Section 2 introduces the basic knowledge of the HMM and BN and the premise assumption of the HMM-BN hybrid model, as well as the learning of HMM parameters. Section 3 introduces the model's construction in detail. Section 4 introduces the examples of the TE process and the CSTR process, based on which the effectiveness of the model is verified. Section 5 gives the conclusion.

2. Basic Knowledge and Method Description

2.1. Hidden Markov Model. The hidden Markov model (HMM) describes the process of randomly generating observation sequences from hidden Markov chains, which belong to the generation model. Markov process refers to the assumption that in a random process, the conditional distribution of state i_{t+1} at time t+1 is only related to its previous state i_t , that is, $p(i_{\ell+1} | i_1, i_2, ..., i_t) = p(i_{t+1} | i_t)$. The process is shown in Figure 1(a). HMM is a probability model about time series. As shown in Figure 1(b), it describes the process of randomly generating an unobservable state random sequence (state sequence) from a hidden Markov chain and then generating an observation random sequence (observation sequence) from each state. There are a limited number of hidden states in the model, and the observed values are output discretely or continuously. HMM has two kinds of probabilities, one is called transition probability, and the other is called observation probability. The former indicates the probability of changing from a hidden state to other states, while the latter refers to the probability of generating observations from a hidden state. Although the real hidden state sequence is hidden and cannot be observed directly, it can be inferred by measurement. HMM has the following main elements:

(1) Hidden state:

$$Q = \{q_1, q_2, \dots, q_N\},$$
 (1)

where N represents the number of hidden states.

(2) Observation variable:

$$V = \{v_1, v_2, \dots, v_M\},$$
 (2)

where M represents the number of observed variables.

(3) Initial probability distribution:

$$\pi = \{\pi_i\}.\tag{3}$$

(4) State transition probability:

$$A = \left\{a_{ij}\right\}, 1 \le i, j \le N,\tag{4}$$



FIGURE 1: Schematic diagram of Markov process and hidden Markov model process.

where $a_{ij} = P(i_{t+1} = q_j | i_t = q_j)$ represents the probability that the state will change from $i \longrightarrow j$.

(5) Observation probability matrix:

$$B = \{b_j(k)\},\tag{5}$$

where $b_j(k) = P(O_t = v_k | i_t = q_j)$ represents the probability relationship between the hidden variable and the observed variable.

As we all know, the parameter that the HMM model needs to learn and train is λ . λ contains three main elements, namely the initial probability distribution π , state transition matrix *A*, and emission matrix *B*, which are formulated as follows:

$$\lambda = \{\pi, A, B\}. \tag{6}$$

The HMM model needs to solve the following three problems:

(1) Evaluation problem, that is, using a forward-backward algorithm to calculate $P(o|\lambda)$

The specific calculation steps of forward algorithm are as follows:

(i) Definition and assumption: Use forward-backward algorithm to calculate the following data sequence:

$$o(t \times M) = [o_1, o_2, \dots, o_t]^T,$$
 (7)

where t represents the measured quantity and M represents the variable quantity.

For the convenience of calculation, set forward variables as follows:

$$\alpha_t(i) = P(o_1, o_2, \dots, o_t, i_t = q_i \mid \lambda), \tag{8}$$

where *t* refers to time *t* and *i* refers to state *i*. (ii) Initialization:

$$a_1(i) = \pi_i b_i(o_1), (9)$$

where $1 \le i \le N$. (iii) Inductive recurrence:

0

$$\alpha_{t+1}(j) = \sum_{i=1}^{N} b_j(o_{t+1}) a_{ij} \alpha_t(i), \qquad (10)$$

where
$$1 \le t \le T - 1, \ 1 \le j \le N$$
.
 $P(o \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i).$ (11)

- (iv) Specific calculation steps of backward algorithm:
- (a) Definition and assumption:

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \dots, o_t | i_t = q_i, \lambda).$$
(12)

(b) Initialization:

$$\beta_t(i) = 1. \tag{13}$$

(c) Inductive recurrence:

$$\beta_t(i) = \sum_{j=1}^N b_j(o_{t+1}) \alpha_{ij} \beta_{t+1}(j),$$
(14)

where $t = T - 1, T - 2, ..., 1, 1 \le i \le N$.

(d) I remination:
And
$$\beta_t(i) \longrightarrow P(o | \lambda)$$
.

(2) Parameter learning problem, that is, using EM algorithm to learn the best parameter λ

Step E, solving the Q function:

$$Q(\lambda, \tilde{\lambda}) = \sum_{I} \log P(o, I | \lambda) p(I | \tilde{\lambda}),$$

$$= \sum_{I} \log P(o, I | \lambda) \frac{p(o, I | \tilde{\lambda})}{p(o | \tilde{\lambda})},$$
(15)
$$\propto \sum_{I} \log P(o, I | \lambda) p(o, I | \tilde{\lambda}),$$

where $\tilde{\lambda}$ is the current estimated value of hidden Markov model parameter, and λ is the model parameter to be maximized.

Joint probability calculation formula:

$$P(o, I | \lambda) = \pi_i \alpha_{i_1 i_2} \alpha_{i_2 i_3} \dots \alpha_{i_{T-1} i_T} b_{i_1 o_1} b_{i_2 o_2} \dots b_{i_T o_T}.$$
 (16)

Logarithmic function of joint probability calculation formula:

$$\log P(o, I | \lambda) = \log \pi_i \alpha_{i_1 i_2} \alpha_{i_2 i_3} \dots \alpha_{i_{T-1} i_T} b_{i_1 o_1} b_{i_2 o_2} \dots b_{i_T o_T},$$

= $\log \pi_i + \sum_{t=1}^{T-1} \log \alpha_{i_t i_{t+1}} + \sum_{t=1}^T \log b_{i_t o_t}.$
(17)

Substitute equation (16) into the Q function:

$$\sum_{I} \log P(o, I | \lambda) p(I | \tilde{\lambda})$$

$$= \sum_{I} \log \left(\pi_{i} \alpha_{i_{1}i_{2}} \alpha_{i_{2}i_{3}} \dots \alpha_{i_{T-1}i_{T}} b_{i_{1}o_{1}} b_{i_{2}o_{2}} \dots b_{i_{T}o_{T}} \right) P(o, I | \tilde{\lambda}),$$

$$= \sum_{I} \left(\log \pi_{i} + \sum_{t=1}^{T-1} \log \alpha_{i_{t}i_{t+1}} + \sum_{t=1}^{T} \log b_{i_{t}o_{t}} \right) P(o, I | \tilde{\lambda}).$$
(18)

Step M, solving the parameters of the model:

$$\begin{cases} \tilde{\pi} = \operatorname{argmax} \sum_{I} \left[\log \pi_{i} P(o, I | \tilde{\lambda}) \right], \\ \tilde{A} = \operatorname{argmax} \sum_{I} \left[\log a_{ij} P(o, I | \tilde{\lambda}) \right], \\ \tilde{B} = \operatorname{argmax} \sum_{I} \left[\log b_{j}(k) P(o, I | \tilde{\lambda}) \right]. \end{cases}$$

$$s.t. \quad \begin{cases} \sum_{i=1}^{N} \pi_{i} = 1 \\ \sum_{i=1}^{N} a_{ij} = 1 \\ \sum_{i=1}^{N} b_{j} = 1 \end{cases}$$

$$(19)$$

The parameters of the model can be obtained by solving equation (19) with the Lagrange multiplier method.

(3) Decoding problem, that is, using Viterbi algorithm to find a state sequence I, which makes P(I|o) maximum

Define the maximum probability in all single paths $(i_1^*, i_2^*, \dots, i_t^*)$ with the state of *i* at time *t*:

$$\delta_t(i) = \max P(o_1, o_2, \dots, o_t, i_1, i_2, \dots, i_{t-1}, i_t = q_i | \lambda).$$
(20)

From the definition and hidden Markov hypothesis, the recursive formula of variable δ can be obtained, and the following formula can be deduced:

$$\delta_{t+1}(j) = \max P(i_{t+1} = i, i_t, \dots, o_{t+1}, o_t, o_{t-1}, \dots, o_1 | \lambda),$$

= $\max \left[\delta_t(i) a_{ij} \right] b_j(o_{t+1}).$ (21)

The t - 1 th node defining the path with the highest probability among all the single paths $(i_1, i_2, ..., i_t)$ with the state i at time t is

$$\varphi_t(i) = \operatorname{argmax} \left[\delta_{t-1}(j) a_{ij} \right], i = 1, 2, \dots, N.$$
(22)

2.2. Bayesian Network. Bayesian network (BN) is a directed acyclic graph, which consists of nodes representing variables and directed edges connecting these nodes. Nodes represent random variables, and the directed edges between nodes

represent the relationships among nodes. Conditional probability is used to express the dependencies among variables, and prior probability is used to express information without parents. It is an uncertainty processing model that simulates causality in the process of human reasoning.

The nodes in the directed acyclic graph of a BN represent random variables (X_1, X_2, \ldots, X_n) , which can be observable variables, hidden variables, unknown parameters, etc. In order to determine the root cause variables of a fault, this study uses the causal inference capability of BN to find the root cause variables of a fault.

As shown in Figure 2, assuming that the industrial process variable of node *A* directly affects the industrial process variable of node *B*, i.e., $A \longrightarrow B$, the directed arc (A, B) from the variable of node *A* to the variable of node *B* is established by the arrow that points from *A* to *B*, and the weight (i.e., connection strength) is expressed by the conditional probability P(B|A). In short, a BN is formed by drawing the variables involved in a research system independently in a directed graph according to the conditions. It is mainly used to describe the conditional dependence between random variables, with circles representing variables and arrows representing conditional dependence.

The joint probability distribution of nodes x can be expressed by the following formula:

$$p(x) = \prod_{i \in I} p\left(x_i \,\middle|\, x_{pa(i)}\right),\tag{23}$$

where pa(i) represents the parent node of node *i*.

For any random variable, its joint probability can be obtained by multiplying their local conditional probability distributions

$$p(x_1, x_2, \dots, x_k) = p(x_k | x_1, x_2, \dots, x_{k-1}) \dots p(x_2 | x_1) p(x_1).$$
(24)

The learning relationship diagrams are considered important in BN. The knowledge of causality and influence among variables provides the fundamental reason for the whole system. BN is capable of calculating the posterior probability according to the evidence for variables and knowing the states of several other variables. Domain expert knowledge and statistical data are two main information sources for building BN. In the field of fault detection in complex industrial processes, the conditional probability table (CPT) of BN is provided through a historical database. Unfortunately, when generating CPT from historical data, there may be problems such as insufficient data, missing values, fuzzy confidence limits of variables, etc. To solve these problems, experts' experience and knowledge are introduced into the BN to improve its accuracy.

The Bayesian network relies mainly on graph structure and CPT to update the BN through likelihood evidence. After providing CPT and likelihood evidence to complete the BN update, the iterative update of the fault can be obtained by calculating and analyzing the posterior probability. In fault isolation, the parent node with the largest percentage change is the root cause variable of the fault.



FIGURE 2: Simple form of Bayesian connection and conditional probability table.

2.3. Expected Maximum Algorithm Amelioration. As is known to all, the hidden Markov model has a challenging problem when learning parameters, which means the expected maximum (EM) algorithm will cause accuracy problems due to initial sensitivity and local optimization. However, this disadvantage will not be greatly affected in a small number of datasets. Consequently, this problem has not attracted attention. Considering the arrival of the era of big data, the data-driven fault detection and prediction model must be an indispensable tool for the state monitoring of complex industrial process systems. In this paper, the EM algorithm is upgraded to ameliorate its robustness and expand its applicable scope. The rest of this section will introduce the advancement of the EM algorithm.

Since the data collected for industrial process variables are usually continuous values, the state emission probability *B* can be described by a Gaussian mixture model (GMM) (μ , σ , w), where μ and σ are the mean and standard deviation of the sequence X, respectively, $w = (w_1, w_2, \dots, w_k)$ is the Gaussian subdistribution weight and K is the number of Gaussian subdistributions. From the perspective of a learning strategy, EM algorithms can be called parallel learning methods when learning the state emission probability parameters. The algorithm chooses the initial parameters of each set of Gaussian functions and simultaneously, or in parallel, estimates the probability that each point will be generated in each Gaussian function. The process is competitive, and if the probability of a data point belonging to a member is high, the probability of it belonging to other members decreases, and the parameters are updated at the same time. Solving the two problems of the EM algorithm, requires all Gaussian functions to "find" each Gaussian scatter "cluster" from the scatter set through sequential learning before competing in parallel. The mean lies in the larger region of this "cluster," and the variance reflects the structure (or shape) of the scatter "cluster."

The steps for learning the state emission probability parameters using the EM method are as follows:

Step 1. Set the initial parameters μ^0, σ^0, w^0 ;

Step 2. From equation (25)calculate μ_h^{t+1} , Σ_h^{t+1} , w_k^i (*i* = 1,..., N; *h* = 1,..., K);

Step 3. Iterate to $\sum_{h=1}^{K} \|\mu^{t+1} - \mu^t\| < \delta$ convergence using step 2;

where ${\cal G}$ indicates a high-dimensional Gaussian function

 $G(x^{i}, \mu_{h}, \sum_{h}) = (|\sum_{h}|^{-(1/2)}/(2\pi)^{d/2})e^{-1/2}(x^{i} - \mu_{j})^{T}$ $\sum_{h}^{-1}(x^{i} - \mu_{h}), \sum = \sigma^{2}I, i \text{ indicates the } i\text{th sample, } h$ indicates the *h*th Gaussian member, \sum indicates the covariance, and *d* indicates the marginal distance.

$$\begin{cases} \mu_{k}^{t+1} = \frac{\sum_{i=1}^{N} w_{k}^{i} x^{i}}{\sum_{i=1}^{N} w_{k}^{i}}, \\ \sum_{k}^{t+1} = \frac{\sum_{i=1}^{N} w_{k}^{i} (x^{i} - \mu_{k}^{t}) (x^{i} - \mu_{k}^{t})^{T}}{\sum_{i=1}^{N} w_{k}^{i}}, \\ w_{k}^{i} = \frac{G(x^{i}, \mu_{k}^{t}, \Sigma_{k}^{t}) \sum_{i=1}^{N} w_{k}^{i}}{G(x^{i}, \mu_{k}^{t}, \Sigma_{k}^{t}) \sum_{j=1}^{N} w_{k}^{j}}. \end{cases}$$
(25)

The biggest problem with the EM algorithm is that it cannot guarantee the global optimal solution or that the obtained solution will change with different initial values, that is, the so-called initial value sensitivity. As shown in Figure 3. The horizontal axis of the graph represents the unknown parameter (assuming there is only one parameter here), and the vertical axis represents the likelihood function L. If the initial point is selected as point A, then the point M1 that finally converges is naturally the global maximum point of the likelihood function, and the corresponding parameter is naturally the global optimal solution. However, if the initial point is point B, then the final convergence point is point M2. This is just the maximum point, and the corresponding parameter is the local optimal solution. In order to address the initial sensitivity and local optimum problems, this study uses two parts of optimal parameter learning, sequential learning and parallel learning. Sequential learning: there is no direct competition for data resources between the Gaussian functions, and each Gaussian function learns its own parameters (mean and covariance) in different time periods. The learning process for each Gaussian function is essentially the same as for the EM algorithm: first, the class of the data point set is divided using the current parameters, then the parameters of the current Gaussian member are re-estimated based on the division, and then the original dataset is redivided using the updated parameters. And so on, until there is no significant change in the estimated parameters between the first and second times. Parallel learning: there is a direct conflict between the individual Gaussian functions regarding the allocation of data resources. All Gaussian functions compete with each other to update the parameters simultaneously (and the weight coefficients are updated accordingly).

Through the abovementioned methods, the appropriate parameters of the Gaussian function are found and used as initialization parameters of the EM algorithm. In this case, the EM algorithm is used to fine-tune the parameters from a global perspective. The algorithm structure block diagram is shown in Figure 4.

The improved EM method learns the emission probability parameters in the following steps:



FIGURE 3: Local optimal problem.



FIGURE 4: Improved structure block diagram of EM algorithm.

2.3.1. Sequential Learning

Step 1. Set the initial values of σ and \in ;

Step 2. Set h = 1, $X_{I}^{h} = X$;

Step 3. While $XL \neq do$

Randomly selecting $\mu_h(0) \in X_L^h$ as the initial mean; Set initial covariance $\sum_{h} (0) = \sigma^2 I$;

Get initial allocation $X_{s}^{h}(0) = \left\{ x \mid x \in X_{L}^{h}, d_{\sum_{m}}(x, \mu_{h}) \leq \varepsilon \right\}, \quad X_{s}^{h}(0) \text{ is the}$ point that falls within $V_{\epsilon}(\mu_m)$;

Finding the high-density region and structure of the hth Gaussian scatter cluster by iteration; t = 0;

While
$$\|\mu_h(t+1) - \mu_m(t)\|_2 < 10^{-6}$$
;

Update the mean value of the *h*th Gaussian member μ_h , $\mu_{h}(t+1) = \sum_{x^{i} \in X_{s}(t)} x^{i} / N(X_{s}(t));$

Update the covariance of the *h*th Gaussian member \sum_{h} , $\sum_{m} (t+1) = \sum_{x^i \in X_s(t)} (x^i -$

$$\mu_m(t))(x^i - \mu_m(t))^T / N(X_s(t)) - 1;$$

Add the same very small positive real number δ to each eigenvalue of the covariance $\sum_{m} (t+1)$, $\sum_{m} (t+1) = \sum_{m} (t+1) + \delta \cdot I$, preventing the occurrence of eigenvalues of zero and hence of singular matrices;

Update
$$X_s^h(t+1) = \left\{ x \mid x \in X_L, d_{\sum_m(t)}(x, \mu_m(t) \le \varepsilon) \right\};$$

 $t = t+1;$

End {While}

Calculate the weight factor of the hth Gaussian member, $w^h = N(X_c)/N(X);$

When learning the next Gaussian membership parameter, the last data point assigned during the previous sequential learning is ignored, $X_L = X_L - X_s^h(t)$;

$$m = m + 1;$$

End {While}

Step 4. Remove noncompliant data points by reference to the number X_s^h of data points contained in all members.

2.3.2. Parallel Learning. The parameters $(\mu, \sigma, w)^0$ obtained by sequential learning are used as the initial parameters of the EM method, and then all parameters are continued to be fine-tuned using EM.

Where X denotes the original dataset (x_1, x_2, \ldots, x_N) , X_s^h denotes the data points contained in the *h*th member, X_L denotes the set of data points left after ignoring the elements learned from the previous members, $V_{\varepsilon}(x)$ denotes the high-dimensional Gaussian ellipsoid, ε denotes the radius, x denotes the centroid, N(X) denotes the number of data points in the dataset X, and $d_{\sum}(x,\mu)$ denotes the Marxist distance.

The goal of the EM algorithm is to ensure the maximum log-likelihood estimation value. The larger the value, the better the algorithm. In this section, it is verified that the improved EM algorithm is superior to the original EM algorithm by comparing the log-likelihood estimation values. The specific results are shown in Figure 5. It can be seen from the figure that the test results of log-likelihood estimation and iteration times are obviously lower than those of the original EM algorithm. It shows that our methods can get better parameter estimates. Although the improved algorithm in this section cannot guarantee that the global optimal result can be found every time, the superiority of the algorithm is verified by comparison. It lays the foundation for improving the robustness and expanding the application range of the HMM-BN hybrid model.

3. Fault Diagnosis and Prediction of HMM-BN Hybrid Model

This section describes the HMM-BN hybrid model's construction, fault detection, and fault prediction in detail, as shown in Figure 6. The following section explains the structure of the model step by step.

3.1. Dataset Establishment. Normal operating condition data is not the type of data detected under the optimal working conditions of a complex industrial system. Although some defects may occur with the operation of complex industrial systems, the system can still provide basic production functions. For improving the robustness of the HMM-BN hybrid model, normal operating conditions data are defined using continuous industrial process knowledge and experts' experiences of process operation mode, sequence, state, and variable coupling. The introduction of expert empirical knowledge can usually determine the key characteristic parameters required for continuous industrial process operating conditions, fault monitoring, and building continuous industrial process operating datasets and BN. In this study, the Tennessee Eastman Process (TE) and Continuously Stirred Tank Reactor (CSTR) process knowledge and operating experience are used to select continuous operating variables, extract key characteristic parameters, determine coupling relationships between variables, and build process datasets and BN. However, how to systematically introduce experts' empirical knowledge into other continuous industrial processes, define data from normal operating conditions, and collect and build the required datasets for fault detection remains a challenge.

There are various different methods of data preprocessing [32, 33]. Data preprocessing is the process of cleaning up and converting the original dataset, which usually includes deleting outliers, supplementing missing values, normalizing, feature extraction, and the definition of a suitable data format. In this study, the data preprocessing approach is to read operational data from the original dataset, convert it into a data format suitable for further analysis, identify and extract key feature parameters based on engineering knowledge and expert experience, remove outliers, and delete or add missing values. Ultimately, a dataset suitable for process inspection is created.

3.2. Data Analysis. In line with the basic knowledge of the improved EM algorithm, HMM, and BN introduced in Section 2, it carries out model training and parameter learning. The specific training and learning steps are as follows:

3.2.1. Off-Line Modeling

Step 1. Training the HMM and BN models with the normal operating conditions dataset constructed by Section 3.1;

Step 2. Firstly, the parameters of equation (6) are initialized, and then the most probable sequence is calculated according to the evaluation problem in Section 2.1;

Step 3. Learning of model parameters and maximum likelihood values by the improved EM algorithm in Section 2.3;

Step 4. Calculating log-likelihood estimates and storing the results in a historical database to facilitate later process predictions;

Step 5. Based on the decoding problem in Section 2.1, fault detection and prediction using the range of increase or decrease, mean, and standard deviation of three adjacent log-likelihood estimates as evidence;

Step 6. Build the initial BN causality diagram in the way described in Section 2.2, calculating the conditional probability values, and transmit them as evidence to the initial BN to build the BN under normal operating conditions.

3.2.2. Online Monitoring

Step 1. Extract key feature parameters and construct test sets based on engineering knowledge and expert experience;

Step 2. Learning parameters, calculating log-likelihood estimates, and computing likelihood evidence using the trained HMM model in offline modeling;

Step 3. Gives detection and prediction results. Based on the parameter learning, log-likelihood estimates of new data samples of industrial processes, an industrial system is considered to be anomalous if it appears to be significantly different, or a system failure if the anomaly persists for a period of time. Based on the log-likelihood estimates and the pattern of increase and decrease for the new data sample, the log-likelihood estimate history library is queried to find three consecutive log-likelihood estimates with similar patterns of increase and decrease, mean, and standard deviation, which are



FIGURE 5: Comparison between the improved EM algorithm and the original EM algorithm.



FIGURE 6: HMM-BN modeling schematic.



FIGURE 7: Reduction method based on LL sequence.

output as predicted values. The sequence of log-likelihood values is examined to determine whether the system will fail in the future. The exact process is shown in Figure 7;

Step 4. The BN is updated by using log-likelihood evidence to determine the root cause of the fault and to isolate the fault. To determine the root cause variables that generate the fault, it evaluates the values of different variables over a period of time before the fault identification point and determines the possibility of each variable causing the fault. It is also transmitted to BN as valid evidence of causal deduction;

Step 5. Engineers handle faults and make fault records.

3.3. *Result Output.* Fault detection and fault prediction results are derived from data analysis, and the results are presented to facilitate engineers' understanding of the operational status of industrial processes. In the event of a fault alarm, the root cause variables are output via BN to help field service personnel deal with system faults in a timely and accurate manner. Finally, the results are documented to enhance the robustness of the model.

4. Application Example

4.1. TE Process. The TE process includes five main units: the reactor, the condenser, the compressor, the separator, and the stripper. There are four reactions that generate two products. Three gaseous reactants are supplied to the reactor, where liquid products are formed through catalytic chemical reactions. The product enters the condenser in the form of steam for liquefaction. Then, the product passes through a gas-liquid separator, and the condensed product and the noncondensed product are separated. A centrifugal compressor recycles the uncondensed product into the reactor, and the condensed product into the reactor, and the condensed product stream flows from the bottom of the stripper and is pumped downstream for further refining.

The flow chart of the TE chemical industry is shown in Figure 8. The chemical process of TE includes 12 operating variables and 41 measured variables (22 continuous measured variables and 19 component measured values). In this paper, 22 continuous variables from the output variables are selected as observation variables, and their descriptions are shown in Table 1. The whole TE simulation process, which includes 15 types of known faults and 5 types of unknown faults [34–36], has simulated and detected 10 types of fault scenarios related to monitored variables. This paper selects 4 kinds of fault scenarios to verify the effectiveness of the model, and the specific fault types and causes are shown in Table 2.

E Feed Loss is a simulation case study, and other test results are given in Section 5. All normal operating condition data were extracted from these 10 fault datasets as the training set, and the test set was composed of 1500 samples (1130 normal samples, 120 samples with smaller fault ranges, and 250 fault samples). First, the HMM was trained, and the BN was built according to the offline modeling method in Section 3.2. The training curve is shown in Figure 9 and the BN is shown in Figure10. Then, a database of log-likelihood estimates is built to lay the foundation for introducing fault data detection and prediction. Finally, a test set is used for validation, and the validation results are shown in Figure 11.

It can be seen from Figure 9 that the log-likelihood estimation value is about constant after 11 iterations. HMM parameters are constantly improved in the iterative updating process, and finally a stable training model is obtained. From Figure 11, it can be seen from the change in the actual log-likelihood estimation curve that the HMM model detected weak anomalies at time node 1126 and strong changes at sample 1265. From the detection results, the HMM model is not only more sensitive to strong fault changes but also possesses the ability to detect weak fault data; in short, the HMM model possesses a pleasing fault detection capability. For the purpose of prediction, the HMM model is used to calculate the mean, deviation, increase, and decrease patterns of the new data set, and the log-likelihood estimator



FIGURE 8: TE model process flow chart.

Variable no	Description	Unit
XMEAS1	Feed (stream 1)	Kscmh
XMEAS2	D feed (stream 2)	kg/hr
XMEAS3	E feed (stream 3)	kg/hr
XMEAS4	A and C feed (stream 4)	Kscmh
XMEAS5	Recycle flow (stream 8)	Kscmh
XMEAS6	Reactor feed rate (stream 6)	Kscmh
XMEAS7	Reactor pressure	Kpa gauge
XMEAS8	Reactor level	%
XMEAS9	Reactor temperature	°C
XMEAS10	Purge rate (stream 9)	Kscmh
XMEAS11	Product separator temperature	°C
XMEAS12	Product separator level	%
XMEAS13	Product separator pressure	Kpa/
XMEAS14	Product separator underflow (stream 10)	m ³ /hr
XMEAS15	Stripper level	%
XMEAS16	Stripper pressure	Kpa gauge
XMEAS17	Stripper underflow (stream 11)	m ³ /hr
XMEAS18	Stripper temperature	°C
XMEAS19	Stripper steam flow	kg/hr
XMEAS20	Compressor work	kw
XMEAS21	Reactor cooling water outlet temperature	°C
XMEAS22	Separator cooling water outlet temperature	°C

TABLE 1: Description of continuous pro-	ocess variables of TE process.
---	--------------------------------

TABLE 2: True root causes for tested fault conditions [35].

Fault ID	True root variable
E Feed Loss	XMEAS3
IDV 4	XMEAS9
IDV 5	XMEAS11
IDV 6	XMEAS1

history library is retrieved to find a similar sequence of loglikelihood estimates, which is the prediction of the industrial process operation. As can be seen from the change in the predicted log-likelihood estimate curve in Figure 11, the predicted results deviate from the actual results to some extent, but the overall change curve shows that the predicted result curve follows the actual test result change curve, which also indicates the predictive validity of the hybrid model.

TE chemical process has a cyclic variable, XMEAS (5). Given that BN is an acyclic network, it establishes the repeated virtual node XMEAS (5) of cyclic flow. In order to ameliorate the robustness of the HMM-BN hybrid model in order to improve the robustness of the hybrid HMM-BN model, engineering knowledge is introduced into the BN, and expert experience of continuous industrial process operation modes, sequences, and variable correlations is used to build a Tennessee Eastman process BN with causality and expert empirical knowledge. First, it uses normal operation data to train HMM. Then, it analyzes the first half of the test data by HMM to obtain the detected data strings and



FIGURE 9: HMM training curve of TE process.



FIGURE 10: Bayesian network of TE chemical process.

their log-likelihood values. From this generated database, it is able to infer various combinations of states and their respective probabilities. The logarithmic likelihood estimation value can be loaded for each of the data strings detected by HMM. The load value must be normalized to be used as the conditional probability of the CPT table. By establishing CPT and prior probabilities, BN has the ability to detect system anomalies. Figure 10 shows BN in normal operation.

After the HMM model detects the failure, it quickly retrieves the string sequence of recent time, calculates the failure probability of each variable, and finally updates the BN with probability and likelihood evidence. When BN is updated with likelihood evidence, the rate of change is evaluated at each node. When the root node has the highest rate of change, it is considered the cause of a specific problem. If not, the maximum change rate on the previous consecutive parent nodes is considered the root cause. Figure 12 shows the BN of the E feed loss failure. It can be seen from Figure 12 that the root cause variable of the E Feed Lose fault is XMEAS3. Compared to Table 2, it can be noticed that the model detection result is completely correct, which is finally used for fault isolation.

4.2. CSTR Process. The CSTR process contains nine process variables. In this study, 4400 normal operating state samples



FIGURE 11: Detection and prediction result of E feed lose fault.



FIGURE 12: Bayesian network of E feed lose.

were collected as the training set, and another 2850 samples were collected as the test set (1850 normal operating samples and 1000 fault samples) to verify the efficiency of the HMM-BN hybrid model.

Firstly, the HMM was trained with the offline model and training set, and the CSTR-BN process was built based on the training results and expert experience. The HMM training curve is shown in Figure 13, and the BN for the normal operation state is shown in Figure 14. The trained HMM was used to detect the CSTR process test set data, and the test set detection and prediction results are shown in Figure 15. To help engineers find the root cause variables leading to the fault and repair the fault, finally, the conditional probability of the BN is updated based on the loglikelihood evidence, and the fault diagnosis results for the BN are shown in Figure 16. The sequence of log-likelihood estimates in Figure 15 shows that the HMM-BN hybrid model not only detects fault data in a timely manner at the location but also accurately predicts the occurrence of faults. Although there is some difference between the actual and



FIGURE 14: Bayesian network of CSTR process.

predicted curves at the peaks, the overall trend between the actual and predicted curves is consistent, which confirms the fault detection and prediction potential of the hybrid HMM-BN model. When comparing Figures 14 and 16, the most significant change in probability can be found for variables 6, 7, and 8, indicating a CSTR process failure due to a fault in variables 6, 7, and 8.

5. Results and Discussion

In this paper, the purpose is to present a method for fault diagnosis and prediction of complex continuous industrial processes. To achieve this objective, a hybrid hidden Markov model-Bayesian network (HMM-BN) model is proposed, and improvements are made to deal with the initial sensitivity and local optimality problems of the expected maximum (EM) algorithm in the HMM approach, while continuous industrial process operation datasets and the BN are built using continuous industrial process knowledge and expert experience of process operation modes, sequences, states, and variable coupling. The hybrid model is finally validated in the Tennessee Eastman (TE) chemistry process and the continuously stirred tank reactor (CSTR) process. The validity of the model is finally verified in the TE chemical process and the CSTR process, demonstrating the possibility of diagnosing and predicting unknown faults in continuous, complex industrial processes.

In this study, the EM algorithm is improved to reduce the impact of the initial sensitivity and local optimum problems by introducing standard deviation, mean, increasing and decreasing mode ranges when retrieving loglikelihood values during data analysis and introducing expert empirical knowledge when building the dataset and Bayesian network to improve the robustness of the model.

As shown in Table 3, it is proved that the detection, prediction, and isolation results of the hybrid model for the



FIGURE 15: Detection and prediction result of CSTR process fault.



FIGURE 16: Bayesian network of CSTR process fault.

TABLE 3: Other variables fault detection result

Fault ID	True root cause	Diagnosed root cause
E Feed Loss	XMEAS3	XMEAS3
IDV 4	XMEAS9	XMEAS9
IDV 5	XMEAS11	XMEAS11
IDV 6	XMEAS1	XMEAS1

remaining three types of faults are completely correct by comparing the root cause variable and the detection result variable of the fault data [35]. In this paper, four fault scenarios are used to verify the effectiveness of the HMM-BN hybrid model. In accordance with the four detection results, it can be shown that the hybrid model can be used for continuous, complex industrial process state monitoring. However, it has not been verified whether there are other possibilities for this hybrid model. Readers can extend it to other aspects on the basis of the research ideas in this paper in order to fulfill the original intention of this paper by expanding the applicable scope of this model.

To further optimize the model. There are some aspects of the HMM-BN hybrid model that still need to be developed and studied in depth.

 By looking at the detection results in Figure 11, it can be seen that the hybrid model has the ability to detect weak fault signals to a certain extent; whether it can be used for early fault detection is yet to be studied. In addition, the hybrid model requires a large amount of data training; however, there are challenges in collecting fault data samples in normal industrial processes. [37] How to use a small number of samples for fault diagnosis and prediction may be a further research direction.

- (2) Resolving the two hypothetical premises of the HMM and improving the robustness of the model.
- (3) Learning complex continuous industrial process knowledge through artificial intelligence algorithms.
- (4) How to introduce experts' experience and knowledge with a more formal approach.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant no. 52065065) and the Xinjiang University Doctoral Research Start-Up Fund Project BS190216.

References

- V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis," *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 293–311, 2003.
- [2] T. Aggab, P. Vrignat, M. Avila, and F. Kratz, "Remaining useful life estimation based on the joint use of an observer and a hidden Markov model," *Proceedings-Institution of Mechanical Engineers Part O Journal of Risk and Reliability*, vol. 236, no. 5, pp. 676–695, 2021.
- [3] D. Wang, K. L. Tsui, and Q. Miao, "Prognostics and health management: a review of vibration based bearing and gear health indicators," *IEEE Access*, vol. 6, 2017.
- [4] K. Zhong, M. Han, and B. Han, "Data-driven based fault prognosis for industrial systems: a concise overview," *IEEE/ CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 330–345, 2020.
- [5] J. Antoni and R. B. Randall, "The spectral kurtosis: application to the vibratory surveillance and diagnostics of rotating machines," *Mechanical Systems and Signal Processing*, vol. 20, no. 2, pp. 308–331, 2006.
- [6] R. B. Randall and J. Antoni, Differential Diagnostics of Gear and Bearing Faults Using Spectral Correlation, Applied Mechanics-Progress and Applications-The Third Australasian Congress on Applied Mechanics, World Scientific, Singapore, 2002.
- [7] Z. Ge and Z. Song, "Distributed PCA model for plant-wide process monitoring," *Industrial & Engineering Chemistry Research*, vol. 52, no. 5, pp. 1947–1957, 2013.
- [8] B. Gu and W. Xiong, "Multi-mode process monitoring based on multi-block information extraction PCA method with local neighbourhood standardisation," *CIE Journal*, vol. 70, no. 2, pp. 736–749, 2019.

- [9] R. Muradore and P. Fiorini, "A PLS-based statistical approach for fault detection and isolation of robotic manipulators," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 8, pp. 3167–3175, 2012.
- [10] W. Xi, Z. Li, Z. Tian, and Z. Duan, "A feature extraction and visualization method for fault detection of marine diesel engines," *Measurement*, vol. 116, pp. 429–437, 2018.
- [11] Z. Chen, S. X. Ding, K. Zhang, Z. Li, and Z. Hu, "Canonical correlation analysis-based fault detection methods with application to alumina evaporation process," *Control Engineering Practice*, vol. 46, pp. 51–58, 2016.
- [12] Q. Zhang, P. Li, X. Lang, and A. Miao, "Improved dynamic kernel principal component analysis for fault detection," *Measurement*, vol. 158, Article ID 107738, 2020.
- [13] X. Yuan, Y. Wang, C. Yang, Z. Ge, Z. Song, and W. Gui, "Weighted linear dynamic system for feature representation and soft sensor application in nonlinear dynamic industrial processes," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1508–1517, 2018.
- [14] W. Ku, R. H. Storer, and C. Georgakis, "Disturbance detection and isolation by dynamic principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 30, no. 1, pp. 179–196, 1995.
- [15] X. Wang, U. Kruger, and G. W. Irwin, "Process monitoring approach using fast moving window PCA," *Industrial & Engineering Chemistry Research*, vol. 44, no. 15, pp. 5691– 5702, 2005.
- [16] Q. Jiang and B. Huang, "Distributed monitoring for largescale processes based on multivariate statistical analysis and Bayesian method," *Journal of Process Control*, vol. 46, pp. 75–83, 2016.
- [17] Q. Jiang, X. Yan, and B. Huang, "Performance-driven distributed PCA process monitoring based on fault-relevant variable selection and bayesian inference," *IEEE Transactions* on *Industrial Electronics*, vol. 63, no. 1, pp. 377–386, 2016.
- [18] Y. Wang, H. Yang, X. Yuan, Y. A. Shardt, C. Yang, and W. Gui, "Deep learning for fault-relevant feature extraction and fault classification with stacked supervised auto-encoder," *Journal of Process Control*, vol. 92, pp. 79–89, 2020.
- [19] Y. Wang, D. Wu, and X. Yuan, "LDA-based deep transfer learning for fault diagnosis in industrial chemical processes," *Computers & Chemical Engineering*, vol. 140, Article ID 106964, 2020.
- [20] P. Smyth, "Hidden Markov models for fault detection in dynamic systems," *Pattern Recognition*, vol. 27, no. 1, pp. 149–164, 1994.
- [21] C. U. Mba, V. Makis, S. Marchesiello, A. Fasana, and L. Garibaldi, "Condition monitoring and state classification of gearboxes using stochastic resonance and hidden Markov models," *Measurement*, vol. 126, pp. 76–95, 2018.
- [22] A. Pa, U. Cesaro, M. Chadli et al., "Fault detection on fluid machinery using hidden Markov models," *Measurement*, vol. 151, Article ID 107126, 2020.
- [23] G. Zhiyong, L. Jiwu, and W. Rongxi, "Prognostics uncertainty reduction by right-time prediction of remaining useful life based on hidden Markov model and proportional hazard model," *Maintenance and Reliability*, vol. 23, pp. 154–165, 2021.
- [24] S. Rebello, H. Yu, and L. Ma, "An integrated approach for system functional reliability assessment using dynamic bayesian network and hidden Markov model, reliability engineering," *System Safety*, vol. 180, pp. 124–135, 2018.
- [25] M. G. Don and F. Khan, "Dynamic process fault detection and diagnosis based on a combined approach of hidden Markov

and Bayesian network model," *Chemical Engineering Science*, vol. 201, 2019.

- [26] M. G. Don and F. Khan, "Process fault prognosis using hidden Markov model-Bayesian networks (HMM-BN) hybrid model," *Industrial & Engineering Chemistry Research*, vol. 58, p. 27, 2019.
- [27] M. T. Amin, S. Imtiaz, and F. Khan, "Process system fault detection and diagnosis using a hybrid technique," *Chemical Engineering Science*, vol. 189, pp. 191–211, 2018.
- [28] T. Liu and G. Duan, "A selective assembly strategy to improve mechanical product quality based on Bayesian network and genetic algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 116, no. 11-12, pp. 3619–3634, 2021.
- [29] T. Pang, T. Yu, and B. Song, "A Bayesian network model for fault diagnosis of a lock mechanism based on degradation data," *Engineering Failure Analysis*, vol. 122, no. 2021, Article ID 105225, 2021.
- [30] P. J. Smyth, "Hidden Markov models for fault detection in dynamic systems," *Pattern Recognition*, vol. 27, 1995.
- [31] Y. Wang and C. Zhao, "Probabilistic fault diagnosis method based on the combination of nest-loop fisher discriminant analysis and analysis of relative changes," *Control Engineering Practice*, vol. 68, 2017.
- [32] Q. Jiang and X. Yan, "Parallel PCA-KPCA for nonlinear process monitoring," *Control Engineering Practice*, vol. 80, pp. 17–25, 2018.
- [33] S. Yoon and J. F. Macgregor, "Fault diagnosis with multivariate statistical models part I: using steady state fault signatures," *Journal of Process Control*, vol. 11, no. 4, pp. 387–400, 2001.
- [34] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Computers & Chemical Engineering*, vol. 17, 1993.
- [35] H. Gharahbagheri, S. Imtiaz, and F. Khan, "Combination of KPCA and causality analysis for root cause diagnosis of industrial process fault," *Canadian Journal of Chemical Engineering*, vol. 95, no. 8, pp. 1497–1509, 2017.
- [36] H. Yu, F. Khan, and V. Garaniya, "Modified independent component analysis and Bayesian network-based two-stage fault diagnosis of process operations," *Industrial & Engineering Chemistry Research*, vol. 54, no. 10, pp. 2724–2742, 2015.
- [37] Y. Zhuo and Z. Ge, "Auxiliary information-guided industrial data augmentation for any-shot fault learning and diagnosis," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7535–7545, 2021.