Fault Diagnosis for TE Process Using RBF Neural Network

XIN LIU AND HAI HE (Member, IEEE)
Chongqing City Management College, Chongqing 401331, China
Corresponding author: Hai He (cswu_haihe@163.com)

This work was supported in part by the Science and Technology Research Program of Chongqing Education Commission of China under Grant KJQN201803310 and Grant KJQN201803308, and in part by the Research Innovation Team of Chongqing City Management College under Grant KYTD202006.

ABSTRACT Fault diagnosis of industrial process has long been a challenging issue owing to the industrial system that exhibits nonlinearity, coupled parameters and time-varying in the production process. This paper presents a novel dynamic fault diagnosis model (AUKF-RBF) based on radial basis function (RBF) neural network for Tennessee Eastman (TE) industrial process. In order to effectively reflect the dynamic features of industrial system, a dynamic fault diagnosis model is established based on UKF and RBF neural network. In particular, UKF is used to optimize the weights, the center, and the width of the hidden layer nodes of RBF. Furthermore, to reduce the effect of the inappropriate initial filter parameters in UKF, an adaptive factor $\delta_k$ is developed to tune the covariance matrix adaptively. Finally, the proposed fault diagnosis algorithm is applied to TE benchmark industrial process. Experimental results show the effectiveness of the proposed fault diagnosis method.

INDEX TERMS Fault diagnosis, radial basis function, unscented Kalman filter, Tennessee Eastman process, dynamic modeling.

I. INTRODUCTION

Modern industrial systems are very complicated due to the structure, production flow, and automation degree [1]. A fault in any part of the system may result in the entire system failure. With the increasing demands on reliability and security of the modern industrial system, it is significantly important to detect potential or occurring faults before such faults lead to serious property damage and casualties. Therefore, how to use the current and past states of the system to detect and diagnose the faults is receiving increasing attention nowadays.

Fault diagnosis methods can be divided into model-based and data-driven [2]–[4]. Model-based methods have been successfully applied to many industrial systems, where physical and mathematical models were used to describe the production process [5], [6]. However, the model-based approaches require extensive priori knowledge of the system, it is difficult to know the dynamics of the system in real applications. On the other hand, data-driven approaches have been widely used in fault diagnosis [7]–[11]. In the data-driven method, the large amount of production process data were used to analyze and establish the fault diagnosis model. The data-driven methods mainly include fuzzy logic, multivariate statistical process monitoring (MSPM), support vector machine (SVM), and artificial neural networks (ANNs). Each method has its own advantage and disadvantage, the usage of these methods depends on the different requirements.

ANNs have been widely used in complex system modeling and fault diagnosis due to the nonlinear approximation ability and the strong self-learning ability [12]–[16]. Back propagation (BP) NN as the widely used ANNs has been successfully applied to fault diagnosis in industrial
process. Compared with back propagation (BP) NN, RBF neural network has the characteristics of better approximation ability, faster learning speed, and simpler network structure. Many researchers have worked on RBF fault diagnosis issue. Hence, many different fault diagnosis approaches and techniques have been proposed. Liu et al. [17] proposed an adaptive RBF neural network to approximate nonlinear fault function. Huang et al. [18] proposed a fault detection and diagnosis method for nonlinear systems with modeling uncertainties. Two RBF neural networks were used to establish the unknown nonlinear dynamics and detect the nonlinear characteristics of the fault function. However, the computational load is large due to the two neural networks. In [19], a combined with particle swarm optimization (PSO) and RBF neural network method was proposed to diagnose the nonlinear prediction model. However, PSO has a slow fine-tuning ability, so it is easy to be trapped into local optima. Ke et al. [20] proposed a self-adaptive RBF neural network method for power transformer fault diagnosis. The proposed RBF model is combined with fuzzy c-means and quantum-inspired particle swarm optimization. Unfortunately, the algorithm has difficulties in achieving high-dimensional data parameter optimization. In [21], a fault tolerant algorithm is proposed to train an RBF network and select the RBF centers simultaneously. In order to establish the RBF neural model, a combined with the hybrid forward and the continuous forward algorithm was presented to optimize network parameters [22]. However, the learning process of the algorithm was complicated. It is difficult to enforce in practice industrial application.

Modeling method based on neural networks typically assume that the environmental noise and internal state variables of the system are stable and that the mapping between input and output variables are static. As a result, the static modeling method is limited to relatively certain process or real-time industrial system. With the evolution of the actual system, the accuracy and generalization of the established model based on an early database cannot be guaranteed, which leads to the problem that the static neural networks model cannot effectively reflect the dynamic features of industrial system [23]. In recent years, UKF has received extensive attention among scientists and engineers due to its straightforward implementation, higher filtering accuracy, and favorable convergence [24]. UKF has been used to establish dynamic evolution modeling by optimizing the neural network model in real time. To avoid the improper initial filter value of the UKF, an adaptive factor is design to adaptively optimize the covariance matrix. The excellent performance of the proposed method is demonstrated using TE process date. The key contributions of the proposed algorithm are as following:

- A dynamic fault diagnosis model is established based on UKF and RBF neural network, where UKF is used to train the weights $\omega_{m,l}$, the center $c_m$, and the width $b_m$ of the RBF neural network, where UKF can be used to establish dynamic evolution modeling by optimizing the neural network model in real time. In this paper, a novel fault diagnosis approach based on adaptive UKF and RBF neural network is proposed for fault diagnosis in TE process. First, UKF is used to train the weights $\omega_{m,l}$, the center $c_m$, and the width $b_m$ of the RBF neural network, where UKF can be used to establish dynamic evolution modeling by optimizing the neural network model in real time. To avoid the improper initial filter value of the UKF, an adaptive factor is design to adaptively optimize the covariance matrix. The excellent performance of the proposed method is demonstrated using TE process date. The key contributions of the proposed algorithm are as following:

- To reduce the effect of the inappropriate initial filter parameters, adaptive factor $\delta_k$ is developed to tune the covariance matrix adaptively.

The rest of this paper is organized as follows: Section II briefly reviews RBF and UKF. Section III presents the AUKF-RBF fault diagnosis model, where adaptive factor is constructed to handle the covariance matrix. In Section IV, the proposed algorithm is applied to TE process, and experimental results are discussed in detail. Finally, the conclusion is provided in Section V.

II. OVERVIEW OF RBF AND UKF

A. RBF NEURAL NETWORK

RBF neural network is a feedforward neural network composed of input layer, single hidden layer, and output layer [31], [32]. The connection weights between the input layer and hidden layer are obtained by nonlinear transformation; a linear transformation is applied between the hidden layer and output layer. The output of the network is the linear weighted sum of the weights of the hidden layer. The network topology is shown in Fig. 1.

In the hidden layer, gaussian radial basis function is applied to transform the input vector. The low-dimensional data are mapped to a high-dimensional space. In the RBF, the linearly
inseparable problem in low-dimensional space is transformed to the linearly separable problem in high-dimensional space, effectively overcoming the problems of local minima and slower convergence speed in a BP neural network [33].

B. UNSCENTED KALMAN FILTER

Unscented Kalman filter (UKF) is an efficient nonlinear filter method [34], [35]. In contrast to a traditional filter algorithm, the UKF uses a certainty-sampling strategy to approximate the nonlinear distribution instead of linearizing the nonlinear function and calculating the Jacobian matrix. The specified chosen sample points can represent the true mean and covariance of the Gaussian random variable. The main idea of the UKF is the unscented transformation (UT). The statistical characteristic of the random variable of the nonlinear transformation can be calculated using the UT.

Assume that $x$ and $C$ represent the mean and covariance of the $n$ dimension random variable $X$. The random variable $X$ is propagated through the nonlinear function $y = f(x)$. In order to approximate the statistical characteristic of $y$, a certainty sampling strategy is used to choose $2l + 1$ sigma vectors $\xi_i$ with weights $w_i$. The $2l + 1$ sigma vectors are formulated as follows

$$
\begin{align*}
\tilde{\xi}_{0,k-1} &= \hat{x}_{k-1} \\
\tilde{\xi}_{i,k-1} &= \hat{x}_{k-1} + \left(\sqrt{\lambda + \rho}C_{k-1}\right)_{i} \\
\tilde{\xi}_{i+n,k-1} &= \hat{x}_{k-1} - \left(\sqrt{\lambda + \rho}C_{k-1}\right)_{i} \\
w_{0}^i &= \rho/(l + \rho) \\
w_{0}^i &= \rho/(l + \rho) \\
w_{0}^i &= \rho/(l + \rho)
\end{align*}
$$

where $i = 1, 2, \ldots, l$, $w_0^i$ and $w_0^i$ represent the weights of predicted mean and covariance, respectively, $\rho = \alpha^2(l + \nu) - \lambda$ is a scaling parameter, $\nu$ is a secondary scaling parameter, $\alpha$ reflects the distribution of the sample points, and $\beta$ is used to approximate the prior distribution of $\hat{x}_k$.

We can obtain the transformed $y_i$ by a nonlinear transformation for sigma vectors as

$$y_i = f(\xi_i).$$

Mean and covariance of $y_k$ can be approximated as

$$
\tilde{y}_k \approx \sum_{i=0}^{2n} w_i^{(m)} y_i \\
C_y \approx \sum_{i=0}^{2n} w_i^{(c)} (y_i - \tilde{y}_k)(y_i - \tilde{y}_k)^T.
$$

III. AUKF-RBF ALGORITHM

For RBF neural network, the connection weights of hidden layer to output layer, the center nodes, and the width of hidden layer have great influence on the prediction performance of network model. However, it is difficult to determine the appropriate value for these parameters. UKF has a strong non-linear tracking ability, in this paper, we use the UKF to optimize the weights, the center, and the width of the hidden layer nodes of RBF neural network. The AUKF-RBF neural network is a three layers feedforward neural network, and the inputs of the neural network are the state of the system, and the output is the type of the faults. The structure of AUKF-RBF is shown in Fig. 2.

Then the state vector $w_k$ of UKF can be formulated as follows:

$$
\begin{align*}
w_k &= [\omega_{11}, \ldots, \omega_{ml}, \zeta_1, \ldots, \zeta_m, b_1, \ldots, b_m].
\end{align*}
$$

where $\omega_m$, $c_m$, and $b_m$ are the connection weights, the center nodes, and the width of hidden layer, respectively. $m$ and $l$ represent the number of the hidden layer nodes and the number of the input layer nodes, respectively.

The system state vector and observation vector can be expressed as follows

$$
\begin{align*}
w_k &= w_k - 1 + q_k \\
y_k &= h_k(w_k, x_k) + r_k \\
&= F^t(w_k^n, F^{n-1}(w_k^{n-1} \ldots F^2(w_k^1, x(k)))) + r_k
\end{align*}
$$

where $q_k$ and $r_k$ are the system process noise and observation noise, respectively. $F^t$ is transfer function of the $n$th neural network layer, $x(k)$ is the input of the neural network. According to the nonlinear transformation principle of (1)-(5) and the system function of (7), the learning procedures of UKF-RBF neural network are calculated as follows:

Step 1: Initialization

$$
\begin{align*}
\bar{w}_0 &= E[w_0] \\
C_0 &= E[(w_0 - \bar{w}_0)(w_0 - \bar{w}_0)^T]
\end{align*}
$$

where $w_0$, $\bar{w}_0$ and $C_0$ represent the initial state vector, mean values and covariance, respectively.

Step 2: Calculate the Sigma points

$$
\xi_i, k-1 = \left[ \begin{array}{c} \hat{w}_{k-1} \\ \hat{y}_{k-1} \end{array} \right] \pm \left( \sqrt{(l + \rho)C_{k-1}} \right)_{i}.
$$

Step 3: Predict update

$$
\begin{align*}
\xi_i, k-1 &= \xi_i, k-1 \\
\hat{w}_{k|k-1} &= \sum_{i=0}^{2n} w_i^{(m)} \xi_i, k|k-1 \\
C_{k|k-1} &= \sum_{i=0}^{2n} w_i^{(c)} (\xi_i, k|k-1 - \hat{w}_{k|k-1}) \\
&= \sum_{i=0}^{2n} w_i^{(m)} (\xi_i, k|k-1 - \hat{w}_{k|k-1}) \\
\hat{y}_{k|k-1} &= h_k(\xi_i, k|k-1, x(k)) \\
C_{k|k-1} &= \sum_{i=0}^{2n} w_i^{(c)} (\xi_i, k|k-1 - \hat{w}_{k|k-1})
\end{align*}
$$

Step 4: Measurement update

$$
\begin{align*}
C_{k|k-1} &= \sum_{i=0}^{2n} w_i^{(c)} (\xi_i, k|k-1 - \hat{w}_{k|k-1}) \\
&= \sum_{i=0}^{2n} w_i^{(c)} (\xi_i, k|k-1 - \hat{w}_{k|k-1})
\end{align*}
$$
The structure of AUKF-RBF.

\[
C_{\tilde{y}_k} = \sum_{i=0}^{2n} w_i^{(c)} (y_{i,k|k-1} - \tilde{y}_{i,k|k-1}) \\
\times (y_{i,k|k-1} - \tilde{y}_{i,k|k-1})^T \tag{17}
\]

\[
\hat{y}_k = \hat{y}_{k|k-1} + G (y_k - \tilde{y}_{k|k-1}) \tag{18}
\]

\[
C_k = C_{k|k-1} - GC_{\tilde{y}_k} G^T \tag{19}
\]

where \( G = C_{\tilde{y}_k} C_{\tilde{y}_k}^{-1} \) is the Kalman gain.

Step 5: Calculate the updated output of the hidden layer
\[
C_m (\|\hat{y}_k - c_m\|) = \exp \left(-\frac{\|\hat{y}_k - c_m\|^2}{2b^2_m}\right), \quad m = 1, \ldots, j \tag{20}
\]

where \( P_m \) is the \( m \)th output of the hidden layer, \( \|\cdot\| \) is the Euclidean distance between the input layer and the hidden layer center.

Step 6: Calculate the output of the RBF
\[
y = \sum_{j=1}^{n} \omega_{m,j} C_m (\|\hat{y}_k - c_m\|). \tag{21}
\]

Step 7: Construct the adaptive factor \( \delta_k \).

For UKF filter, the initial filter parameters have a great influence on the convergence performance. Proper initial parameters can converge quickly. However, there are some deviations between the initial filter parameters and the ideal filter parameters. The deviation will affect the state matrix \( \hat{y}_k \) and covariance matrix \( C_{\tilde{y}_k} \). The errors will accumulate with constant updating of the algorithm. To reduce the effect of the errors produced by the improper initial filter parameters, we use the adaptive factor \( \delta_k \) to tune the covariance matrix adaptively.

The AUKF-RBF algorithm is shown in detail in Algorithm 1.

After introducing the adaptive factor, the theoretic covariance matrix is denoted as \( \hat{C}_{\tilde{y}_k} \), then formula (17) is rewritten as

\[
\bar{C}_{\tilde{y}_k} = \frac{1}{\delta_k} \sum_{i=0}^{2n} w_i^{(c)} (y_{i,k|k-1} - \tilde{y}_{i,k|k-1})(y_{i,k|k-1} - \tilde{y}_{i,k|k-1})^T. \tag{22}
\]

Assume that the estimate of the covariance matrix of the predicted residual vector is obtained and denoted as \( \bar{\hat{C}}_{e_k} \).

Algorithm 1 AUKF-RBF Algorithm

1: Initialize the filtered value \( \hat{y}_0 \) and covariance matrices \( C_0 \).
2: while \( w > T_c \) do
3: Calculate the Sigma point state vector.
4: One-step system state prediction and covariance matrix prediction.
5: Calculate the system observations and covariance matrix.
6: Adaptive adjustment of the covariance matrix.
7: if theoretic covariance matrix \( \geq \) estimate covariance matrix then
8: The adaptive factor \( \delta_k = 1 \).
9: else
10: The adaptive factor \( \delta_k = tr(C_{\tilde{y}_k})/tr(\bar{\hat{C}}_{e_k}) \).
11: end if
12: Calculate the Kalman gain \( G \).
13: Update the system state estimation and covariance matrix.
14: Calculate the updated RBF hidden layer output.
15: end while

According to Sage’s windowing, the estimated covariance matrix of \( \bar{\hat{C}}_{e_k} \) can be designed as follows
\[
\hat{C}_{e_k} = \frac{1}{N} \sum_{i=0}^{N-1} e_{k-i} e_{k-i}^T \tag{23}
\]

where \( N \) is the size of the window, the predicted residual vector \( e_k = y_k - h_k [x_{k|k-1}, x(k)] \).

According to the UKF optimality principle, the theoretic covariance matrix and the estimate covariance matrix should satisfy the following condition
\[
\bar{\hat{C}}_{e_k} = \hat{C}_{\tilde{y}_k}. \tag{24}
\]

From (22), we obtain
\[
\bar{\hat{C}}_{e_k} = \frac{1}{\delta_k} \sum_{i=0}^{2n} w_i^{(c)} (y_{i,k|k-1} - \tilde{y}_{i,k|k-1})(y_{i,k|k-1} - \tilde{y}_{i,k|k-1})^T. \tag{25}
\]
By combining (17), it follows that

\[ \delta_k = \frac{\text{tr}(C_y)}{\text{tr}(\hat{C}_e)}. \]  

(26)

In real industrial application, the adaptive factor should satisfy \(0 < \delta_k \leq 1\), then the adaptive factor can be designed as

\[ \delta_k = \begin{cases} 1, & \text{tr}(C_y) \geq \text{tr}(\hat{C}_e) \\ \frac{\text{tr}(C_y)}{\text{tr}(\hat{C}_e)}, & \text{tr}(C_y) < \text{tr}(\hat{C}_e). \end{cases} \]  

(27)

IV. EXPERIMENTS AND DISCUSSION

The Tennessee Eastman (TE) process is an industrial benchmark for evaluating process monitoring and process control method. The structure of the TE process is shown in Fig. 3. There are five major parts, including reactor, condenser, compressor, separator, and stripper. The process contains eight components: A, B, C, D, E, F, G, and H. The gaseous reactants A, C, D, E and the inert B are fed to the reactor to produce liquid products G and H [36], [37]. There is 21 kinds of faults exist in the TE process, of which 16 kinds are known faults (faults IDV(1) to IDV(15), IDV(21)) and 5 kinds are unknown faults (faults IDV(16) to IDV(20)). TE process contains two blocks of variables, i.e., 12 manipulated variables and 41 measured variables. The measured variables include 22 continuous process measurements and 19 sampled process measurements. All experiments are made using the software MATLAB R2016a on a desktop computer (2.40GHz Intel Core i5 with 8 GB of RAM).

Remark 1: In TE process, the type of faults IDV(16) to IDV(20) belong to unknown faults. Faults IDV(1) - IDV(3) and IDV(6) - IDV(10) are caused by abnormal temperature, pressure or component of reactants A, B, C and D. Faults IDV(4), IDV(11), IDV(13) and IDV(14) are caused by abnormal cooling water of reactors. Faults IDV(5), IDV(12) and IDV(15) are caused by abnormal cooling water of condensers. In this paper, faults IDV(1) to IDV(15) are used to test the performance of the proposed fault diagnosis model.

A. PERFORMANCE OF FAULT DIAGNOSIS

The training data of each fault are sampled once every 3 minutes. The first 500 samples are normal samples and the last 480 samples are fault samples. For each fault, 60 samples are used to test the performance of the proposed algorithm, where the first 30 samples are normal and the last 30 samples are faulted. The RBF neural network is a three layers feedforward neural network, and the inputs of the neural network are the 52 variables of TE. The output of normal sample is 2 and the output of fault sample is 1. For the tested normal sample, if the output is between 1.5 and 2.5 (error is less than 0.5), then the diagnostic result is correct; otherwise, the diagnostic result is incorrect. For the tested fault sample, if the output is between 0.5 and 1.5 (error is less than 0.5), then the diagnostic result is correct; otherwise, the diagnostic result is incorrect. Back propagation (BP) neural network, RBF neural network, and UKF-RBF are used for comparison to demonstrate the effectiveness of the proposed model.

Fig. 4, Fig. 5 and Fig. 6 show the fault diagnosis results of faults IDV(5), IDV(7) and IDV(12), respectively. From these results, it can been seen that the proposed AUKF-RBF algorithm performs the best results as its outputs are more closer 2 and 1. Although UKF-RBF performs not as
well as AUKF-RBF, it also can detect almost all the faults. BP and RBF cannot diagnose the fault accurately as their fault diagnosis models are static. The static modeling method is limited to relatively certain processes of the industrial systems. On the other hand, UKF-RBF and AUKF-RBF methods can effectively reflect the dynamic features of the system in real time.

Taking fault IDV(7) an example, the detailed experimental error of each algorithm are shown in Fig. 7. It can be observed that the prediction error of BP is between $-0.7$ and $0.6$. The false alarm appears at the beginning of the diagnosis. The prediction error of RBF is between $-0.9$ and $0.6$. We can see that the fault diagnosis model is not able to detect the occurrence of fault from the 40th sample. It is obvious that the accuracy of UKF-RBF and AUKF-RBF with dynamic molding is significantly higher than BP and RBF, especially all the output errors of AUKF-RBF are less than 0.5. From these results, we could also conclude that the false alarm rate can be decreased more efficiently with the UKF optimization approach than that without optimization.

To demonstrate the performance of the proposed fault diagnosis for the faults IDV(1) to IDV(15) of TE, the statistical results are displayed in Table. 1 by using fault detection rate (FDR) and false alarm rate (FAR). FDR is the ratio of the number of effective alarms to the total number of faulty samples. FAR is the ratio of the number of false alarms to the total number of normal samples. It can be observed that AUKF-RBF has almost the highest FDR in all the simulated faults. Only in one fault, i.e., IDV(9), UKF-RBF has a higher FDR than AUKF-RBF. It is also obvious that AUKF-RBF has almost the lowest FAR in all the simulated faults, except for IDV(13). Some faults cannot be detected using BP and RBF, i.e., IDV(9) and IDV(14). Compared with BP and RBF, UKF-RBF and AUKF-RBF significantly enhance the FDR and reduce the FAR.

### B. CLASSIFICATION OF MULTIPLE FAULTS

To evaluate the performance of the proposed fault diagnosis model in classifying multiple faults, faults IDV(1) to IDV(15) are used to summarize the classification results. In the multiple faults classification, the output of [0 0 0 0] represents without fault, [0 0 0 1] represents fault IDV(1), [0 0 1 0] represents fault IDV(2), and so on, [1 1 1 1] represents fault IDV(15). 1600 samples are used to train the classification model, where the number of normal samples and fault samples of each fault are 50, respectively. Another 320 samples

| Faults | FDR(%) | FAR(%) |
|--------|--------|--------|
| BP | RBF | UKF-RBF | AUKF-RBF | BP | RBF | UKF-RBF | AUKF-RBF |
| IDV(1) | 83.7 | 86.4 | 96.7 | 98.6 | 12.1 | 11.4 | 13.1 | 13.6 | 0.6 |
| IDV(2) | 75.3 | 82.3 | 91.3 | 94.4 | 6.7 | 2.1 | 1.5 | 0 |
| IDV(3) | - | 40.0 | 70.4 | 71.5 | - | 5.7 | 8.3 | 4.9 |
| IDV(4) | 89.7 | 91.7 | 90.1 | 92.2 | 1.2 | 1.7 | 0 | 0 |
| IDV(5) | 67.7 | 66.1 | 70.5 | 70.5 | 7.7 | 12.4 | 1.6 | 0.8 |
| IDV(6) | 73.3 | 76.7 | 99.88 | 99.98 | 22.1 | 8.4 | 1.3 | 0.2 |
| IDV(7) | 64.7 | 69.5 | 67.7 | 89.2 | 5.8 | 4.7 | 6.7 | 5.8 |
| IDV(9) | - | 68.4 | 68.2 | - | 6.6 | 12.3 | 9.6 |
| IDV(10) | 67.4 | 42.7 | 76.5 | 88.4 | 10.4 | 16.8 | 7.6 | 4.6 |
| IDV(11) | 24.4 | 33.2 | 87.4 | 92.2 | 7.2 | 4.2 | 1.5 | 0.6 |
| IDV(12) | - | 84.2 | 97.5 | 99.88 | - | 13.8 | 10.2 | 0 |
| IDV(13) | 53.3 | 61.7 | 85.2 | 92.3 | 14.6 | 17.7 | 2.6 | 3.4 |
| IDV(14) | - | 83.4 | 93.4 | - | - | 13.8 | 2.6 |
| IDV(15) | 64.4 | 68.6 | 84.2 | 86.8 | 2.1 | 1.9 | 1.6 | 0.9 |
are used to test, where the number of each fault samples are 20. Table 2 shows randomly ten classification results of the multiple faults. It can be observed that almost faults can be classified correctly. Only IDV(5) is classified into IDV(12).

Table 3 shows the classification accuracy and computation time. It is observed that the classification accuracy of the proposed AUKF-RBF is highest. Compared with BP and RBF, UKF training method takes more computation time to diagnose fault. However, classification accuracy of UKF-RBF and AUKF-RBF algorithms increase more than 12 percent. UKF-RBF and AUKF-RBF has almost the same computation time while AUKF-RBF has obvious advantage in classification accuracy.

To sum up, the proposed AUKF-RBF provides a better performance in comparison with the other algorithms in classifying single fault and multiple faults. We can conclude that the proposed AUKF-RBF algorithm can effectively reflect the dynamic features of the system in real time.

V. CONCLUSION

In this paper, a novel AUKF-RBF method based on adaptive UKF and RBF neural network is proposed for fault diagnosis in TE process. UKF is employed to optimize the weights $\omega_m$, center $c_m$ and width $b_m$ of RBF. In order to avoid the inappropriate initial value of the UKF filter, an adaptive factor is proposed to tune the covariance matrix of UKF. Experimental results demonstrate the effectiveness and performance of the AUKF-RBF algorithm for fault detection in the TE process.

It is worth noting that noises are unavoidable in practical industrial systems and the statistical characteristic of noise is unknown or time varying owing to the uncertain internal and external influence factors. Recently, many existing methods have improved the adaptive ability of noise change [23]. Hence, in our future work, we will work on developing robust fault diagnosis method for TE.
REFERENCES

[1] S. Yin, S. X. Ding, X. Xie, and H. Luo, “A review on basic data-driven approaches for industrial process monitoring,” IEEE Trans. Ind. Electron., vol. 61, no. 11, pp. 6418–6428, Nov. 2014.

[2] Z. Gao, C. Cecati, and S. X. Ding, “A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches,” IEEE Trans. Ind. Electron., vol. 62, no. 6, pp. 3757–3767, Jun. 2015.

[3] F. Pacheco, J. V. de Oliveira, R.-V. Sánchez, M. Cerrada, D. Cabrera, C. Li, G. Zurita, and M. Artés, “A statistical comparison of neural classifiers and feature selection methods for gearbox fault diagnosis under realistic conditions,” Neurocomputing, vol. 194, pp. 192–206, Jun. 2016.

[4] N. Bayar, S. Darmoul, S. Hajri-Gabouj, and H. Pirereeval, “Fault detection, diagnosis and recovery using artificial immune systems: A review,” Eng. Appl. Artif. Intell., vol. 46, pp. 43–57, Jun. 2015.

[5] J. Seshadrinathan, B. Singh, and B. K. Panigrahi, “Vibration analysis based interturn fault diagnosis in induction machines,” IEEE Trans. Ind. Informat., vol. 10, no. 1, pp. 340–350, Jul. 2014.

[6] Y. Giritli, L. Zarrì, C. Rossi, F. Filippetti, G. A. Capolino, and D. Casadei, “Advanced diagnosis of electrical faults in wound-rotor induction machines,” IEEE Trans. Ind. Electron., vol. 60, no. 9, pp. 4012–4024, Sep. 2013.

[7] Y. Wang, Q. Jiang, and J. Fu, “Data-driven optimized distributed dynamic PCA for efficient monitoring of large-scale dynamic processes,” IEEE Access, vol. 5, pp. 18325–18333, 2017.

[8] G. Wang and S. Yin, “Quality-related fault detection approach based on orthogonal signal correction and modified PLS,” IEEE Trans. Ind. Informat., vol. 11, no. 2, pp. 398–405, Apr. 2015.

[9] Y. Zhang, H. Zhou, S. J. Qin, and T. Chai, “Decentralized fault diagnosis of large-scale processes using multiblock kernel partial least squares,” IEEE Trans. Ind. Informat., vol. 6, no. 1, pp. 3–10, Feb. 2010.

[10] J. Yi, D. Huang, S. Fu, H. He, and T. Li, “Optimized relative transformation matrix using bacterial foraging algorithm for process fault detection,” IEEE Trans. Ind. Electron., vol. 63, no. 4, pp. 2595–2605, Apr. 2016.

[11] J. Yi, D. Huang, H. He, W. Zhou, Q. Han, and T. Li, “A novel framework for fault diagnosis using kernel partial least squares based on an optimal preference matrix,” IEEE Trans. Ind. Electron., vol. 64, no. 5, pp. 4315–4324, May 2017.

[12] A. Bellini, F. Filippetti, C. Tassoni, and G.-A. Capolino, “Advances in diagnostic techniques for induction machines,” IEEE Trans. Ind. Electron., vol. 55, no. 12, pp. 4109–4126, Dec. 2008.

[13] F. Filippetti, G. Franceschini, C. Tassoni, and P. Vas, “Recent development of induction motor drives fault diagnosis using AI techniques,” IEEE Trans. Ind. Electron., vol. 47, no. 5, pp. 994–1004, Oct. 2000.

[14] H. Su and K. T. Chong, “Induction machine condition monitoring using neural network modeling,” IEEE Trans. Ind. Electron., vol. 54, no. 1, pp. 241–249, Feb. 2007.

[15] R. Zhang, H. Tao, L. Wu, and Y. Guan, “Transfer learning with neural networks for bearing fault diagnosis in changing working conditions,” IEEE Access, vol. 5, pp. 14347–14357, 2017.

[16] Z. Chen, F. Huang, W. Sun, J. Gu, and B. Yao, “RBF-neural-network-based adaptive robust control for nonlinear bilateral teleoperation manipulators with uncertainty and time delay,” IEEE/ASME Trans. Mechatronics, vol. 25, no. 2, pp. 906–918, Apr. 2020.

[17] C. S. Liu, S. J. Zhang, and S. S. Hu, “Adaptive neural-networks-based fault detection and diagnosis using unmeasured states,” IET Control Theory Appl., vol. 2, no. 12, pp. 1066–1076, Dec. 2008.

[18] S. Huang and K. K. Tan, “Fault detection and diagnosis based on modeling and estimation methods,” IEEE Trans. Neural Netw., vol. 20, no. 5, pp. 872–881, May 2009.

[19] M.-F. Feng and J.-C. Lu, “Study on short time traffic flow prediction based on RBF neural network optimized by PSO,” Computer Simul., vol. 27, no. 12, pp. 323–326, 2010.

[20] K. Meng, Z. Y. Dong, D. H. Wang, and K. P. Wong, “A self-adaptive RBF neural network classifier for transformer fault diagnosis,” IEEE Trans. Power Syst., vol. 25, no. 3, pp. 1350–1360, Aug. 2010.

[21] H. Wang, R. Feng, Z.-F. Han, and C.-S. Leung, “ADMM-based algorithm for training fault tolerant RBF networks and selecting centers,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 8, pp. 3870–3878, Aug. 2018.

[22] L. Zhang, K. Li, and E.-W. Bai, “A new extension of Newton algorithm for nonlinear system modelling using RBF neural networks,” IEEE Trans. Autom. Control, vol. 58, no. 11, pp. 2929–2933, Nov. 2013.

[23] W. Zhou, X. Li, J. Yi, and H. He, “A novel UKF-RBF method based on adaptive noise factor for fault diagnosis in pumping unit,” IEEE Trans. Ind. Informat., vol. 15, no. 3, pp. 1415–1424, Mar. 2019.

[24] K. Li, L. Chang, and B. Hu, “A variational Bayesian-based unscented Kalman filter with both adaptivity and robustness,” IEEE Sensors J., vol. 16, no. 18, pp. 6966–6976, Sep. 2016.

[25] Z. Yu, J. Wei, and H. Liu, “A new adaptive maneuvering target tracking algorithm using artificial neural networks,” in Proc. IEEE Int. Joint Conf. Neural Netw. (IEEE World Congr. Comput. Intell.), Jun. 2008, pp. 901–905.

[26] L. Xiaoqiang, L. Aijun, Y. Changjun, and S. Fulin, “Widely linear quaternion unscented Kalman filter for quaternion-valued feedforward neural network,” IEEE Signal Process. Lett., vol. 24, no. 9, pp. 1418–1422, Sep. 2017.

[27] R. Zhan and J. Wan, “Neural network-aided adaptive unscented Kalman filter for nonlinear state estimation,” IEEE Signal Process. Lett., vol. 13, no. 7, pp. 445–448, Jul. 2006.

[28] L. Vidyaratne, M. Alam, J. K. Anderson, and K. M. Iftekharuddin, “Improved training of cellular SRN using unscented Kalman filtering for ADP,” inProc. Int. Joint Conf. Neural Netw. (IJCNN), Jul. 2014, pp. 993–1000.

[29] F. Zha, J. Xu, J. Li, and H. He, “IUKF neural network modeling for FOG temperature drift,” J. Syst. Eng. Electron., vol. 24, no. 5, pp. 838–844, Oct. 2013.

[30] X. Wu and Y. Wang, “Extended and unscented Kalman filtering based feedforward neural networks for time series prediction,” Appl. Math. Model., vol. 36, no. 3, pp. 1123–1131, 2012.

[31] J. Park and J. W. Sandberg, “Universal approximation using radial-basis-function networks,” Neural Comput., vol. 3, no. 2, pp. 246–257, Mar. 1991.

[32] Y. Li, S. Qiang, X. Zhuang, and O. Kaynak, “Robust and adaptive backstepping control for nonlinear systems using RBF neural networks,” IEEE Trans. Neural Netw., vol. 15, no. 3, pp. 693–701, May 2004.

[33] C. Cecati, J. Kolbuz, P. Rozyczki, P. Siano, and B. W. Wielomowski, “A novel RBF training algorithm for short-term electric load forecasting and comparative studies,” IEEE Trans. Ind. Electron., vol. 62, no. 10, pp. 6519–6529, Oct. 2015.

[34] S. J. Julier and J. K. Uhlmann, “New extension of the Kalman filter to nonlinear systems,” Proc. SPIE, vol. 3068, pp. 182–194, Jul. 1997.

[35] E. A. Wan and R. van der Merwe, “The unscented Kalman filter for nonlinear estimation,” in Proc. Adapt. Syst. Signal Process., Commun., Control Symp. (AS-SPCC), Oct. 2000, pp. 153–158.

[36] G. Li, S. J. Qin, and D. Zhou, “Geometric properties of partial least squares for process monitoring,” Automatika, vol. 46, no. 1, pp. 204–210, 2010.

[37] G. Lee, C. H. Han, and E. S. Yoon, “Multiple-fault diagnosis of the Tennessee Eastman process based on system decomposition and dynamic PLS,” Ind. Eng. Chem. Res., vol. 43, no. 25, pp. 8037–8048, 2004.