IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 8, 2020, accepted July 10, 2020, date of publication July 16, 2020, date of current version July 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3009644

A Deep Learning Based Fault Diagnosis Method With Hyperparameter Optimization by Using Parallel Computing

CHAOZHONG GUO[1], LIN LI[2], YUANYUAN HU[1], AND JIHONG YAN[1]
[1]School of Mechatronics Engineering, Harbin Institute of Technology, Harbin 150001, China
[2]College of Electromechanical Engineering, Qingdao University of Science and Technology, Qingdao 266061, China

Corresponding author: Jihong Yan (jyan@hit.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 51505099, and in part by the China Postdoctoral Science Foundation under Grant 2016T90305.

ABSTRACT Bearing fault diagnosis is of great significance to ensure the safe operation of mechanical equipment. This paper proposes an intelligent fault diagnosis method of rolling bearings based on deep belief network (DBN) with hyperparameter optimization by using parallel computing. Different with traditional diagnosis methods that extract the features manually depending on much prior knowledge about signal processing techniques and diagnostic expertise, DBN extracts fault features automatically by machine learning mechanism. Considering the time consuming problem, parallel computing is adopted to the DBN training process by using a Master/Slave mode to improve the training speed so that the global optimization with Genetic Algorithm and higher diagnosis accuracy can be achieved. Finally, the proposed method is verified with the public datasets provided by Case Western Reserve University (CWRU) with various fault depths in different locations and loads of rolling bearings. The results indicate that the proposed method can identify bearing faults under different conditions correctly which significantly enhances the intelligence of fault classification and reduces the time for parameter selection of deep learning models.

INDEX TERMS Deep belief network, hyperparameter optimization, parallel computing, fault diagnosis.

I. INTRODUCTION

With the proposal of "Industrial Internet" and "Industry 4.0", many countries from all over the world put forward different strategies to explore and promote intelligent manufacturing. Key equipment such as numerical control machines and engines has become more automatic, precise and efficient, while big data are collected constantly from these equipment after long-time operation by sorts of sensors, which pushes the fault diagnosis field into the era of "mechanical big data" [1]. However, traditional fault diagnosis is mostly completed by professional technicians and diagnostic experts, which requires much prior knowledge and users' experience. Moreover, it is obviously unrealistic for professional technicians to analyze massive data of mechanical equipment without using new advanced methods [2]–[4]. As one of the most commonly used parts in rotating machinery, failures of rolling bearings can cause expensive shutdowns, drifts in production and even human casualties. Condition monitoring is an effective method to ensure the process safety and improve product quality, where process safety and production quality are two important performance indicators [5], [6]. Therefore, it is necessary to develop new efficient and reliable intelligent fault diagnosis methods to detect performance-indicator-related faults of rolling bearings.

Recently, deep learning has been widely used in vision, speech and language processing, image recognition [7], [8], as well as fault diagnosis [9], [10]. Deep learning network constructs a deep model to simulate the learning process of human brain and is a powerful tool used in wide fields in the big data era. When it is applied to fault diagnostics, the complex mapping relation between fault features and fault types can be established so that the rich internal information behind the data is extracted, which is effective to improve the accuracy of fault diagnosis [1]. However, the performance of most deep learning models hinges on many internal hyperparameters, e.g., architectures of the deep neural networks, learning rates, momentum terms, and training batches. In the deep learning community, hyperparameter selection is usually accomplished manually by properly fine-tuning deep learning parameters and selecting the most

The associate editor coordinating the review of this manuscript and approving it for publication was Liangxiu Han.

appropriate values. The tuning process requires a human expertise and is known to be computationally expensive. As a result, automatic optimization of deep learning hyperparameters is an effective and highly desirable way to find the appropriate parameter values and their optimal combination [11]. Genetic Algorithm (GA) is a powerful population-based robust search and global optimization method that simulates the natural biological evaluation. It searches from a population of solutions rather than from a single point and thus prevents the convergence to suboptimal solutions [12], which provides a possible way to perform the hyperparameter optimization for deep leaning networks.

On the other hand, with the development of big data, rapid computing speed and strong computational capacity of algorithms are of vital importance. Because of the great numerical calculation and data processing capacity, parallel computing has been used as the theoretical basis and supporting tool of large-scale scientific computing [13], [14]. As known, the training process of DBN is time consuming especially for massive data. Considering the strong computing ability, parallel computing is introduced to the training process of the DBN based fault diagnosis model with parameter optimization so that faster computing speed and higher classification accuracy can be achieved. Therefore, the main contributions of this study can be summarized as follows:

1. In order to obtain the optimal hyperparameters of DBN, GA optimization is used for DBN hyperparameter selection so that the optimal hyperparameters can be found to ensure higher diagnosis accuracy for performance-indicator-related faults. This provides an effective way for parameter selection of DBN and other deep learning models.

2. Parallel computing is employed to improve the computing speed of the training process of a deep belief network based diagnosis model with GA optimization. This GA and parallel computing integration method is a useful tool for performance-indicator-related fault diagnosis based on deep learning model when dealing with big data in the era of intelligent manufacturing.

The rest sections of this paper are organized as follows. In Section II, related work about deep learning based fault diagnosis and parallel computing is discussed. In Section III, theories of DBN and parallel computing are presented. In Section IV, the DBN based diagnosis method with parameter optimization is proposed. Section V provides an application of the proposed method on the bearing fault data provided by Case Western Reserve University (CWRU). Finally, Section VI draws the conclusions.

II. RELATED WORK

Many works have been done for fault diagnosis by using traditional methods which usually include two steps: manual feature extraction and fault classification [15]–[19]. In recent years, more and more scholars have introduced deep learning to mechanical fault diagnosis considering the adaptive feature learning ability, and lots of research achievements have been made [20]–[23]. Kong *et al.* [20] proposed a multi-ensemble method based on deep auto-encoders (DAEs) for fault diagnosis of rolling bearings. The final diagnosis result was obtained by majority voting among the results of several DAEs. Shao *et al.* [21] developed an adaptive deep belief network with dual-tree complex wavelet packet to diagnose the rolling bearing faults of CWRU datasets. Ding *et al.* [22] combined SAE-based raw signal sensing and deep Q-network for fault diagnosis of rotating machinery, and the proposed method was verified by using rolling bearing datasets and hydraulic pump datasets. In [23], a multiscale cascading deep belief network (MCDBN) for automatic fault identification of rotating machinery was presented to learn the broader feature representation and improve the recognition precision. However, in the field of deep learning networks, there is not a systematic method to determine the structure and parameters at present [24]. The trial-and-error method is widely used to find the relatively appropriate value for critical parameters by repeated attempts, according to which, the researchers can choose the network topology and model parameters based on their sufficient experience [25]. Ghasemi *et al.* [26] selected batch size, number of hidden layers and number of neurons in each hidden layer, and learning rate by trial and error experiments. In the selection experiment for each parameter, all the other parameters were predefined so that the effect of the selected parameter can be tested. Ma *et al.* [27] also determined the structure and the learning rate by using the trial and error method. In [16], the authors claimed that it is difficult to determine the best network architecture due to the lack of scientific guidance. Several different DBN models were taken for performance evaluation, and the structures were designed entirely by experience from which the most outstanding one was selected. Whereas in some other literature, heuristic optimization methods are utilized to determine the optimal structure and parameters of deep learning network, such as particle swarm optimization [28], [29], simulated annealing [30]. These researches provide good references for this work.

In terms of parallel computing, Randall and Lewis [31] accomplished the ant colony optimization algorithm by using parallel computing. Bekas *et al.* [32] used the Message Passing Interface (MPI) cluster platform in Matlab to implement the large-scale matrix pseudo-spectral calculation. Żurek [33] introduced the parallel calculation technology to GA on a small computer cluster to improve the optimization efficiency. Recently, parallel computing has been applied in the field of deep learning. Zhao *et al.* [34] proposed a parallel computing method of deep belief networks for traffic flow prediction. The data features were learnt by multiple computing nodes by a master-slave parallel computing structure. Cybenko [35] discussed the future of parallel computing in deep learning for social network analysis. Chuang *et al.* [36] proposed a method to enable parallel deep neural network training on the IBM Blue Gene/Q computer system to solve the time consuming problem by using the data-parallel Hessian-free 2nd order optimization algorithm. The literature shows that parallel computing is an effective tool to reduce training time of deep learning networks.

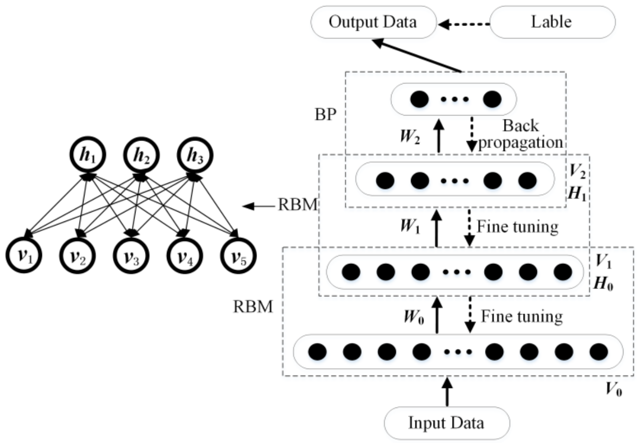

FIGURE 1. Architecture of deep belief network.

Therefore, in this paper the parallel computing is adopted to speed the training process of DBN with GA optimization of the hyperparameters for bearing fault diagnosis.

III. DEEP BELIEF NETWORK AND PARALLEL COMPUTING

A. DEEP BELIEF NETWORK

Deep belief network was first proposed by Hinton and Salakhutdinov [37] in 2006 which has the powerful ability of unsupervised feature learning. This is of great help when introducing DBN to fault diagnosis for large systems with multiple sensors. In fact, a DBN is composed of multiple unsupervised restricted Boltzmann machines (RBMs) and a supervised back-propagation (BP) network [38], as shown in Fig. 1, where a RBM includes one visible layer to represent data and one hidden layer to increase learning capacity. The unsupervised learning is performed in RBM with energy based propagation. Assume v_i represents the state of the i-th visible unit and h_j represents the state of the j-th hidden unit. The energy function of the standard RBM is shown as (1).

$$F_E(v, h|\theta) = -\sum_{i=1}^{n} b_i v_i - \sum_{j=1}^{m} c_j h_j - \sum_{i=1}^{n}\sum_{j=1}^{m} v_i w_{ij} h_j \quad (1)$$

where θ represents the energy function set $\theta = \{w, b, c\}$, w_{ij} indicates the connection weights between visible units and hidden units; b_i indicates the biases of visible units, and c_j is the biases of hidden units. The energy based joint and probability distribution are calculated as (2) and (3).

$$P(v, h|\theta) = \frac{e^{-F_E(v,h|\theta)}}{Z(\theta)} \quad (2)$$

$$Z(\theta) = \sum_{v,h} e^{-F_E(v,h|\theta)} \quad (3)$$

where Z is the partition function. The activation probabilities of the j-th hidden unit and the i-th visible unit are given in (4) and (5).

$$P(h_j|v) = \sigma(c_j + \sum_i w_{ij} v_i) \quad (4)$$

$$P(v_i|h) = \sigma(b_i + \sum_j w_{ij} h_j) \quad (5)$$

where $\sigma(x) = 1/(1 + \exp(-x))$ is the sigmoid function.

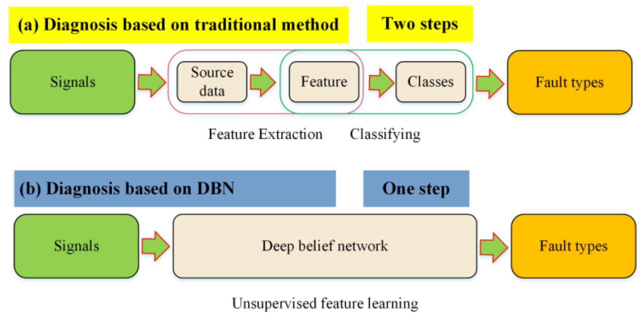

FIGURE 2. Different diagnosis frameworks with the traditional method and DBN model.

The contrastive divergence (CD) algorithm is an approximation of the log-likelihood gradient and is effective in RBM training. The contrastive divergence based training procedure can be descripted as bellow [24].

1. Given the initial bias values of the visible and hidden units, and learning rates.

2. Compute the output of the hidden layer h_j^0 according to (4) by using the forward propagation to variables v_i^0 of the visible layer.

3. Compute the output of the visible layer v_i^0 according to (5) by applying the back propagation to variables h_j^0 of the hidden layer.

4. Apply the forward propagation to variables v_i^1 to obtain new variables h_j^1.

5. Update the critical parameters of RBM using (6) to (8).

$$\Delta w_{ij} = \lambda_w(E[v_i^0 h_j^0] - E[v_i^1 h_j^1]) \quad (6)$$

$$\Delta b_i = \lambda_b(E[v_i^0] - E[v_i^1]) \quad (7)$$

$$\Delta c_j = \lambda_c(E[h_j^0] - E[h_j^1]) \quad (8)$$

where E[.] represents the mathematical expectation.

6. Repeat the above steps for a given number of epochs or until the reconstruction error converges to a certain value.

In general, DBN trains each RBM layer by layer, where the input of visible layer of upper RBM is the output of hidden layer of lower RBM. It leads to the fact that these features obtained from high layer are more representational than that obtained from lower layer. In terms of the last layer, a supervised method is used to train the BP network, and the errors between the actual output and the expected output is propagated back step by step to realize fine tuning of DBN weights.

The frameworks for fault diagnosis with the traditional method and the DBN based method are given in Fig. 2 according to [1], [39]. In the traditional procedure, feature extraction is used to extract the fault related features from data collected by multi-sensors based on signal processing techniques. Then fault diagnosis is performed to identify fault classes based on the selected features, which is a two-step process. This process highly depends on the experts' experience which affects the accuracy of the diagnosis. Whereas by using the unsupervised feature learning based DBN model,

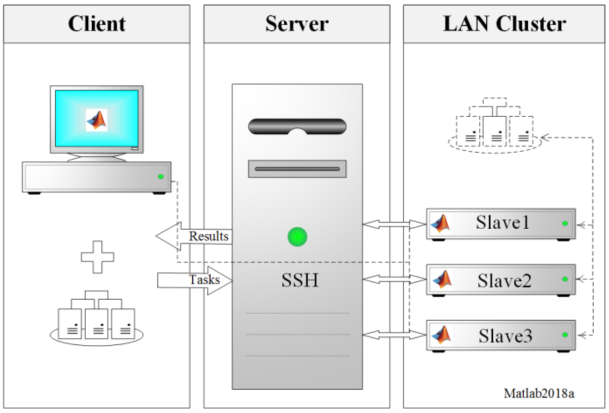

FIGURE 3. Distributed parallel computing platform architecture.

features can be adaptively learned from mechanical raw data with a general-purpose learning procedure instead of being extracted by diagnosticians, which is more intelligent than traditional diagnosis methods [1]. Therefore, the DBN based diagnosis method will be adopted for bearing fault diagnosis in this paper.

B. PARALLEL COMPUTING BASED ON MatlabMPI

Time consuming is still a problem that restricts the training process of DBN models when dealing with massive data. Parallel computing is an efficient method to improve the processing speed by using multiple processors and the training time of DBNs is much reduced. Therefore, the hyperparameter optimization can be achieved for DBN based fault diagnosis by using a parallel computing technique.

Parallel computing refers to solving computing problems by taking advantage of multiple computing resources at the same time to improve the computing speed and processing capacity of computer systems effectively. The basic idea of parallel computing is to use multiple processors to solve the same problem in a cooperative way, which means a given calculation task is divided into several sub-tasks and each sub-task will be assigned to an independent processing unit for computing. One of the most widely used parallel programming techniques is MPI. It is based on information transfer application interface which can achieve multi-host networking collaboration for parallel computing. Due to the coming of big data era, MPI has been applied in many fields, especially in distributed storage parallel machine due to universality, point-to-point communication, diversity of implementation mode, efficiency and so on. In MATLAB programming environment, parallel computing is implemented with the message passing interface, named as MatlabMPI, which is utilized in the DBN based fault diagnosis in this paper to perform parallel computing for hyperparameter optimization of the DBN model.

In details, this paper makes full use of the existing computer resources in our laboratory including four 2U multi-core servers in the same local area network (LAN), to realize the establishment of a distributed parallel computing platform of computer cluster, as shown in Fig. 3.

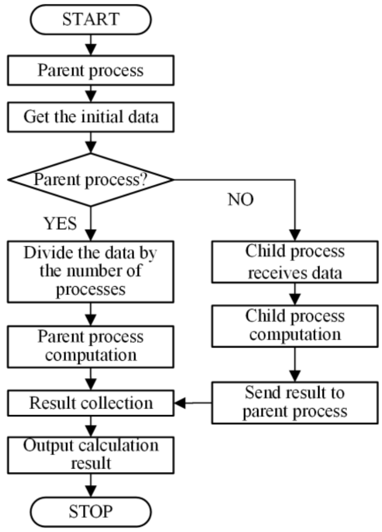

FIGURE 4. Flowchart of parallel computing.

The parallel computing runs in a Master/Slave mode. One server is chosen as the master node to assign computing tasks to other servers as slave nodes, coordinate scheduling parallel computing between processes of slave nodes, and aggregate the calculation results of slave nodes. The flow chart is available in Fig. 4.

IV. AN INTELLIGENT DIAGNOSIS MODEL BASED ON PARAMETER OPTIMIZATION BY USING PARALLEL COMPUTING

During the DBN training process, hyperparameters such as number of hidden layers, number of nodes of each hidden layer, learning rate, momentum, and batch size are the key factors to the DBN performance. Among these parameters, the learning rate and momentum affect the update speed and gradient descent direction of RBMs in the fine tuning phase which are very critical to guarantee the faster convergence of DBNs. An inappropriate learning rate can lead to poor local optimum or make the training algorithm diverge [40], [41]. Momentum helps to accelerate or decelerate the base learning rate with respect to the changing gradient of the network. Traditionally these two parameters are determined by trial and error method separately, or given directly based on experiences, which limits the performance of DBNs. Therefore, in this section, GA is used to find the optimal combination of the learning rate and momentum so that accurate diagnosis can be achieved. GA is a popular meta-heuristic optimization algorithm based on Darwinian evolution and survival of the fittest. It maintains diversity of population to enhance the ability to converge on the global optimum. Moreover, because of the inherent data parallel nature, GAs can be parallelized efficiently which further results in faster solutions [42]. In this paper, GA is introduced to the DBN training process to find the optimal learning rate a and momentum b by using the established distributed parallel computing platform of computer clusters. The flowchart of the GA combined DBN training process is shown in Fig. 5. Generally, the procedure

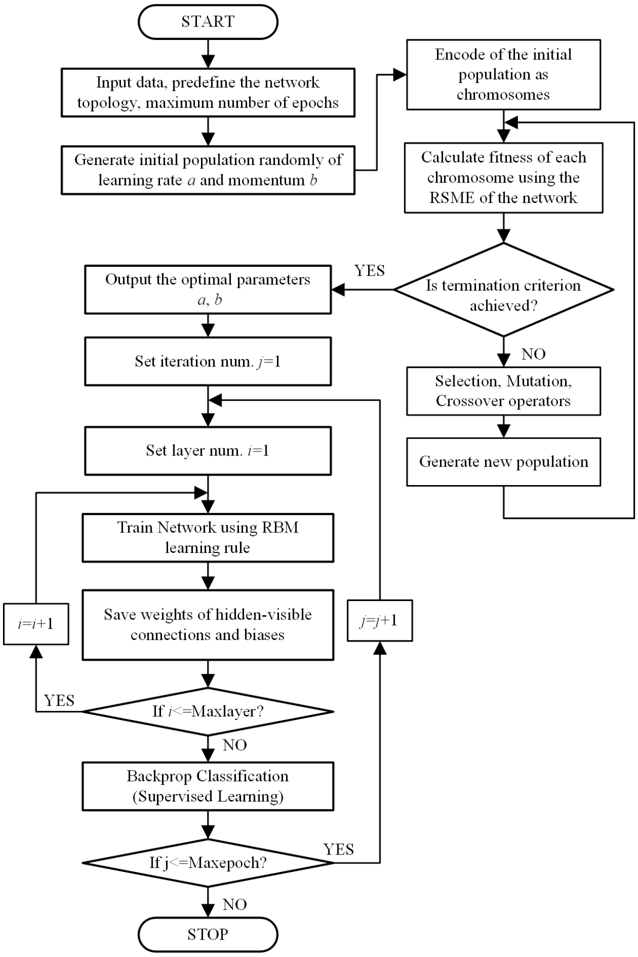

FIGURE 5. Flowchart of GA-DBN training process.

can be divided into two stages. In the first stage, GA is used to obtain the optimal learning rate a and momentum b with the search ranges for both parameters setting as [0, 1]. It is noted that regarding this optimization problem, we only focus on the effect of learning rate and momentum on the DBN performance and treat them as variables. Other parameters such as the number of hidden layers (Maxlayer), the number of neural nodes of each layer, the maximum number of DBN training epochs (Maxepoch) and the maximum number of GA iterations, etc. are predefined and treated as constants [43]–[45]. Herein, the objective of GA optimization is to find the minimum of the root mean square error (RMSE) between the original data and the reconstructed data of validation samples. During the GA-DBN training process, the parallel computing technique is utilized to reduce the computing time by dividing the GA population (chromosomes) into several groups and distributing them to sub-servers for traversal calculation, where each sub-server starts the multi-core computing pool to implement the "parallel" calculation. The fitness values corresponding to the distributed chromosomes are calculated, and the operations of selection, mutation and crossover are performed until the optimal parameters are found. Then the DBN model with optimal parameters (a, b) are trained as described in Section III. The scheme of the proposed

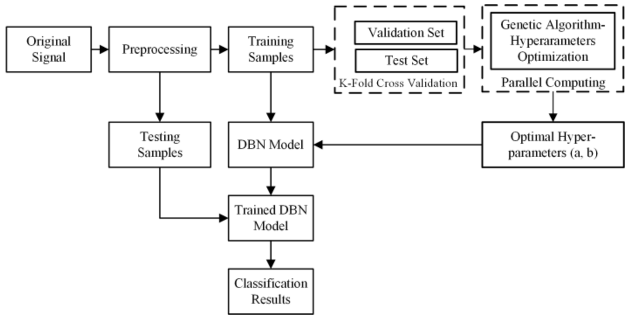

FIGURE 6. Scheme of the proposed intelligent diagnosis method.

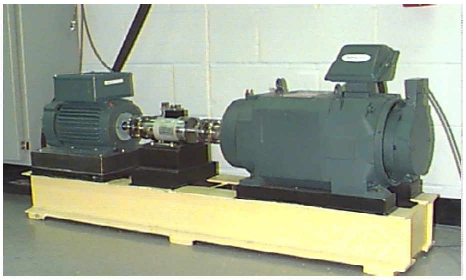

FIGURE 7. The testbed of the bearing system.

intelligent fault diagnosis based on DBN model with parallel computing can be shown in Fig. 6. After the well trained DBN model is obtained, the fault identification of the testing samples is performed.

The proposed method integrates single-machine multi-core computing and computer cluster distributed computing which provides a feasible solution for the fault diagnosis system to process massive data in the era of intelligent manufacturing.

V. FAULT DIAGNOSIS OF ROLLING BEARING

A. INTRODUCTION OF FAULT DATASETS

The datasets used to verify the proposed method are the public bearing data provided by the electronic engineering laboratory of CWRU. The structure of the rolling bearing testbed is shown in Fig. 7. The left part is a 3 horsepower (hp) motor, the right part is a dynamometer. A toque transducer is placed in the middle of the testbed. There are three types of faults located in the inner ring, outer ring and ball. For each type of fault, three broken depths of 0.007, 0.014, and 0.021 inch are introduced by using electric spark. The vibration signals of the rotor system under four different loads of 0, 1, 2, and 3 hp were collected while the sampling frequency is 12 kHz. Table 1 lists four datasets A, B, C, and D, corresponding to data under four types of loads, where each dataset is composed of totally 9 failure types of inner ring failure, ball failure, outer ring failure in three different fault depths and a normal state. Each dataset corresponding to one fault condition contains 50 samples and each sample includes 2,048 sampling data points. Therefore, 500 samples with 9 fault types and one normal state are obtained. A dataset E was created by synthesizing dataset A–D to simulate the multiple working conditions in practical situations.

TABLE 1. Introduction of bearing fault datasets.

Fault type	Fault diameter (inch)	No. of Sample of Each Dataset A(0hp)/ B(1hp) / C(2hp) / D(3hp)	Dataset E(0-3hp)	Classification label
IF1	0.007	50	200	1
IF2	0.014	50	200	2
IF3	0.021	50	200	3
RF1	0.007	50	200	4
RF2	0.014	50	200	5
RF3	0.021	50	200	6
OF1	0.007	50	200	7
OF2	0.014	50	200	8
OF3	0.021	50	200	9
Normal	0	50	200	10

In Table 1, 'Normal' represents the normal state of the bearing, 'RF', 'IF', and 'OF' represent the failure of the ball, inner ring and outer ring of the rolling bearings, respectively.

B. RESULTS AND DISCUSSIONS

Following the procedure shown in Fig. 6, the original vibration data is processed by using the fast Fourier transform (FFT) to obtain the frequency spectrum. Then the normalized spectrum values are regarded as the input of the DBN model. During the modelling process, 40% of the samples are randomly selected as the training data and 20% of the samples are regarded as the validation data while the remaining ones are treated as the test data. Considering the FFT spectrum length is 1,024 points and the output is 10 fault types, the dimensions of DBN input and output are set as 1,024 and 10. The numbers of neural nodes of three hidden layers are predefined as 100, 50, and 20 based on experience and trials. Therefore, the DBN structure for fault diagnosis is 1,024-100-50-20-10. The initial weights are randomly set according to the normal distribution, the biases of which are set as 0. The maximum training epoch is set as 100. After predefining of the DBN topology, GA optimization method is used to obtain the optimal DBN parameters with the evolutionary algebra as 100, the population number as 20, the crossover probability as 0.7 and the mutation rate as 0.1. The root mean square error (RMSE) between the original data and the reconstructed data of validation samples is used as the evaluation indicator to obtain the optimal learning rate a and momentum b within the search ranges of [0, 1]. Then the DBN model is trained with the training and validation samples. The computing time, the optimization parameters and the corresponding RMSE of each dataset (A, B, C, D, and E) are listed in Table 2. Three computing modes including: (1) Single PC, single-core processor (SPSP), (2) Single PC, multi-core processors (SPMP), (3) Multi-PC, multi-core processors (MPMP), are investigated to test the computing ability of parallel computing. Herein, for each sub-server the number of processors is 12, and three sub-servers are used to implement the MPMP calculation as slave nodes. It is noted that the number of samples in dataset A, B, C and D is 500, separately, and the number of samples in dataset E is 2000, as known from Table 1.

TABLE 2. Comparison of time consumed with and without parallel computing.

Datasets	SPSP(s)	SPMP(s)	MPMP(s)	$[a, b]$	RMSE(10^{-3})
A	9316.47	1484.68	621.23	[0.028,0.050]	4.952
B	9327.81	1483.92	619.58	[0.006,0.062]	3.244
C	9341.52	1459.26	603.49	[0.005,0.045]	3.235
D	9298.79	1494.35	652.10	[0.005,0.052]	5.217
E	37309.62	5675.08	2296.47	[0.001,0.078]	9.008

Based on Table 2, it can be calculated that the ratios of computing time of SPSP to SPMP for all the datasets (A–E) are 6.28, 6.29, 6.40, 6.40, 6.22, and 6.57, and that of SPSP to MPMP is 15.00, 15.06, 15.48, 14.26, and 16.25. It indicates that by introducing 12-parallel processors in the SPMP mode, the computing speed has been improved over 6 times than using the SPSP mode. And the computing speed can be improved up to 16.25 times by using the MPMP mode. Moreover, the ratio of computing time of SPSP to SPMP for multi-condition dataset E (6.57) is greater than the ratios for single condition datasets A–D (6.28, 6.29, 6.40, 6.40, 6.22), and the same phenomenon can be seen for that of SPSP to MPMP, which indicates that the computing ability is enhanced with the increase of sample size. The reason is that the communication time between multiple cores remains almost the same, whereas the proportion of communication time to the total running time decreases with larger sample size. These results show that using parallel computing can reduce the time greatly on parameter optimization and the computing performance improves as the task size increases, which proves the effectiveness of parallel computation in the parameter optimization of deep learning model for massive data.

After the DBN model is well trained, the remaining samples are used to test the classification performance of the proposed method with different conditions and loads. In order to eliminate the random error of the DBN test process, the final diagnosis accuracy is calculated as the average value of 10 repeated test results for each dataset. The final results are listed in Table 3. It can be seen that the diagnosis accuracies of the proposed GA-DBN method for the single load datasets (A, B, C, and D) are all 100%. The accuracy can reach to 98.44% even when identifying 10 states under 4 different loads (dataset E). The results indicate that the proposed model has high accuracy in bearing fault diagnosis under various conditions with different fault locations, fault degrees and loads.

The proposed method is compared with other traditional methods and deep learning methods using the CWRU bearing datasets as shown in Table 3. In terms of the traditional methods, the average accuracy of a proposed fuzzy expert method was 96.08% for the CWRU dataset with the defect sizes of 0.007, and 0.021 inch under 2 hp load [15]. In [16], the average diagnosis accuracy was 96.97% for the dataset including totally 11 types of fault labels under 0 hp load by using a grey relation algorithm. In [17], the diagnosis accuracy of a k-nearest neighbor model for CWRU dataset with three fault types under four loads was 92.94%. Similarly, in [18], feature vectors calculated from the Intrinsic Mode Functions of Empirical Mode Decomposition of raw vibration signals were used as the inputs of a typical classifier, and the testing accuracy was 93.82%. In [19], an optimized SVM model was used to identify the fault type and fault severity of the CWRU datasets. The average accuracies for the single load datasets under the loads of 0, 1, 2, 3 hp and the mixed load of 0–3 hp were 97.64%, 99.12%, 99.64%, 97.56%, and 97.91%.

In terms of deep leaning methods, in [20] a multi-ensemble method based on deep auto-encoders was proposed for fault diagnosis of rolling bearings. The diagnosis result of the CWRU dataset under 0 load with 12 health conditions is 96.44%. In [21], an adaptive DBN with dual-tree complex wavelet packet was developed to identify the fault types of CWRU dataset. The testing accuracy for samples with 1 hp load was 98.75%. Whereas in [22], a deep Q-network based fault diagnosis method was proposed for fault diagnosis of CWRU datasets with 10 health conditions under three loads of 1, 2, and 3 hp and one mixed dataset that contains 10 health conditions under loads of 1–3 hp. The average accuracies for these single load datasets were 93.58%, 90.43% and 91.87. Considering the mixed dataset with three loads, the average accuracy was 94.08%. Similarly, in [10] a deep neural network model was used to diagnose the bearing fault of the CWRU dataset under loads of 1–3 hp. The average classification accuracies for these single load datasets were 99.95%, 99.61%, and 99.74%, and the accuracy for the mixed dataset was 99.68%. It can be seen that for single load datasets, the proposed GA-DBN method has higher accuracy up to 100%. But for the mixed load dataset, our classification accuracy is a little lower. The reason is that in our work dataset E contains four loads which make it more complicated in fault diagnosis than the case of three loads. In [23], a multiscale cascading deep belief network (MCDBN) was used to identify the bearing faults under four health conditions of 0–3 hp load. The accuracies were 99.57%, 99.32%, 99.54%, and 99.43% for each single load dataset.

From Table 3, it can be seen that the diagnosis accuracies of the popular benchmark dataset are all over 90% for both the traditional and deep learning methods. Especially, the average accuracies in our paper are 100% for the four single load datasets, and 98.44% for the mixed load dataset, which indicates that the proposed method is relatively more accurate and robust compared to previous works listed in Table 3.

In order to visualize feature extraction capability of DBN, the PCA method is employed to obtain the principal features of the bearing fault with different types. The scatter plots of the original data and the first three main components of the feature matrix extracted from different hidden layers are shown in Figs. 8 to 11, taking dataset C for example.

Fig. 8 shows that the principal components of original data are interwoven, making it difficult to identify various faults.

TABLE 3. Comparisons of diagnosis results.

Datasets	A	B	C	D	E
Fuzzy expert [15]	/	/	96.08%	/	/
Grey relation algorithm [16]	96.97%	/	/	/	/
KNN [17]	/	/	/	/	92.94
EMD+PART Classifiers [18]	/	/	/	/	93.82%
Optimal SVM [19]	97.64%	99.12%	99.64%	97.56%	97.91%
Deep auto-encoders [20]	96.44%		/	/	/
DTCWPT-DBN [21]	/	98.75%	/	/	/
Deep Q-network [22]	/	93.58%	90.43%	91.87%	/
DNN [10]	/	99.95%	99.61%	99.74%	/
Multiscale cascading DBN [23]	99.57%	99.32%	99.54%	99.43%	/
Proposed method	100%	100%	100%	100%	98.44%

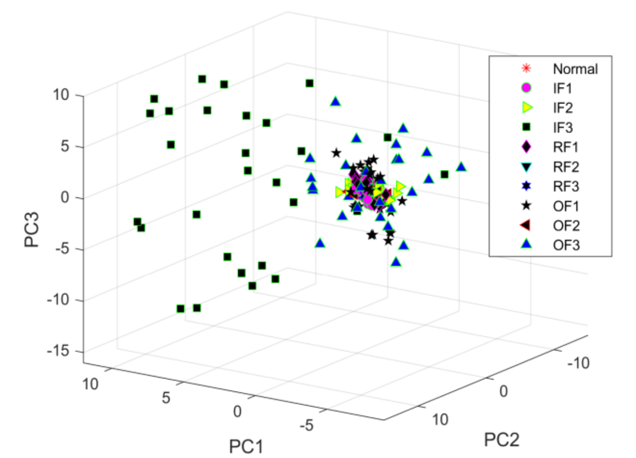

FIGURE 8. Scatter plots of principal components of original data.

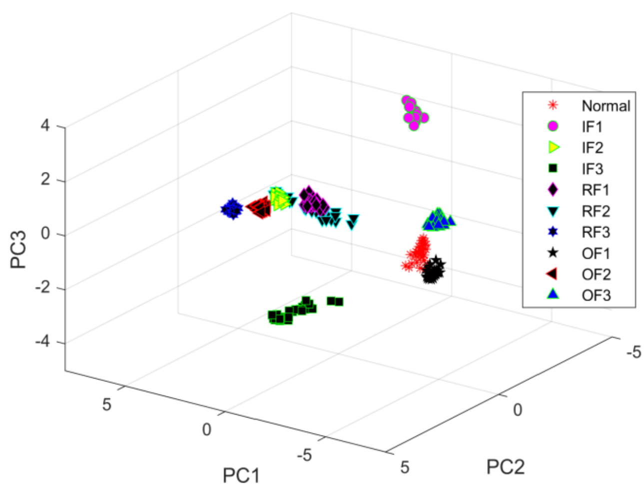

FIGURE 9. Scatter plots of principal components of the feature matrix extracted from the 1st hidden layer.

With gradual increasing of the number of hidden layers, the classification performance is getting better as shown in Figs. 9 to 11. The final output features of the DBN model shown in Fig. 11 indicates that the proposed method has strong ability to learn the fault features for different

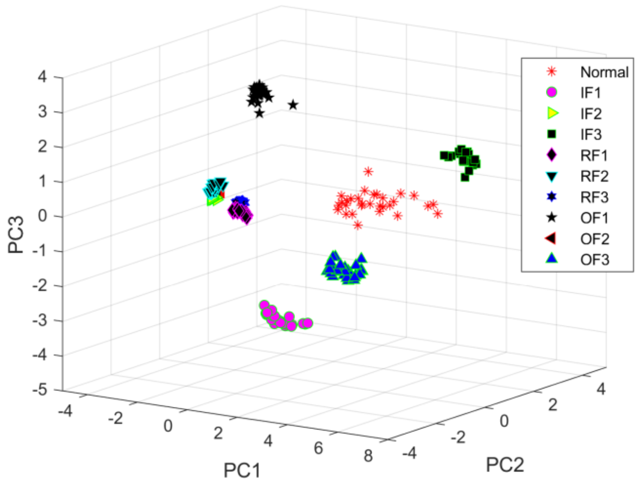

FIGURE 10. Scatter plots of principal components of the feature matrix extracted from the 2nd hidden layer.

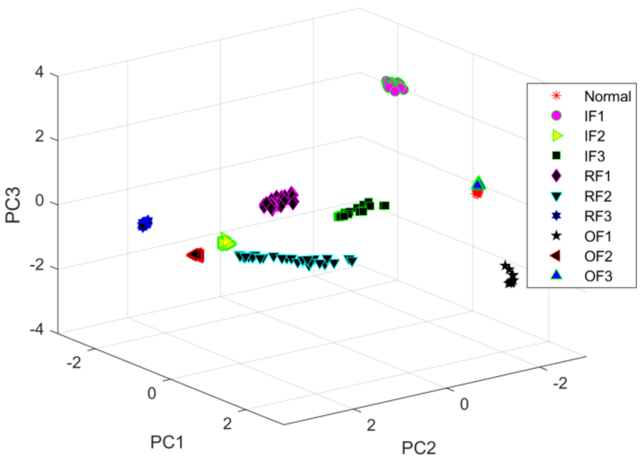

FIGURE 11. Scatter plots of principal components of the feature matrix extracted from the 3rd hidden layer.

fault types adaptively, which ensures high accuracy of the diagnosis results.

VI. CONCLUSION

This paper proposed a DBN based fault diagnosis method with hyperparameter optimization and parallel computing technique. GA was adopted to optimize hyperparameters of the DBN model during its training process. Considering the time consuming problem, parallel computing was introduced to the hyperparameter optimization process to improve the calculation speed by using MatlabMPI in a Master/Slave mode, where one server was chosen as the master node, and the other three servers were set as the slave nodes. The public CWRU dataset was used to verify the proposed method. The results show that with parallel computing, the calculation speed has been improved over 15 times, and the proposed method has better diagnosis performance than several methods presented in previous works. This paper provides a reference for the deep learning algorithm to process massive data with parameter optimization for performance-indicator-related fault diagnosis in the mechanical big data environment. In the future work, GA will be used to optimize other parameters of deep learning model such as the connecting weights, the number of hidden layer nodes, etc., and new experiments will be set up to further verify the proposed method for fault diagnosis.

REFERENCES

[1] Y. Lei, F. Jia, J. Lin, S. Xing, and S. X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data," *IEEE Trans. Ind. Electron.*, vol. 63, no. 5, pp. 3137–3147, May 2016.

[2] Y. Lei, Z. He, and Y. Zi, "Application of an intelligent classification method to mechanical fault diagnosis," *Expert Syst. Appl.*, vol. 36, no. 6, pp. 9941–9948, Aug. 2009.

[3] A. K. S. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mech. Syst. Signal Process.*, vol. 20, no. 7, pp. 1483–1510, Oct. 2006.

[4] C.-Z. Guo, J.-H. Yan, and L. A. Bergman, "Experimental dynamic analysis of a breathing cracked rotor," *Chin. J. Mech. Eng.*, vol. 30, no. 5, pp. 1177–1183, Aug. 2017.

[5] B. Song, X. Zhou, H. Shi, and Y. Tao, "Performance-indicator-oriented concurrent subspace process monitoring method," *IEEE Trans. Ind. Electron.*, vol. 66, no. 7, pp. 5535–5545, Jul. 2019.

[6] B. Song, H. Yan, H. Shi, and S. Tan, "Multisubspace elastic network for multimode quality-related process monitoring," *IEEE Trans. Ind. Informat.*, vol. 16, no. 9, pp. 5874–5883, Sep. 2020, doi: 10.1109/TII.2019.2959784.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, vol. 1, Dec. 2012, pp. 1097–1105.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Jan. 2013, pp. 1–12.

[9] V. T. Tran, F. AlThobiani, and A. Ball, "An approach to fault diagnosis of reciprocating compressor valves using Teager-Kaiser energy operator and deep belief networks," *Expert Syst. Appl.*, vol. 41, no. 9, pp. 4113–4122, Jul. 2014.

[10] F. Jia, Y. Lei, J. Lin, X. Zhou, and N. Lu, "Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data," *Mech. Syst. Signal Process.*, vols. 72–73, pp. 303–315, May 2016.

[11] N. R. Sabar, A. Turky, A. Song, and A. Sattar, "An evolutionary hyper-heuristic to optimise deep belief networks for image reconstruction," *Appl. Soft Comput.*, to be published, doi: 10.1016/j.asoc.2019.105510.

[12] M. V. J. J. Suresh, K. S. Reddy, and A. K. Kolar, "ANN-GA based optimization of a high ash coal-fired supercritical power plant," *Appl. Energy*, vol. 88, no. 12, pp. 4867–4873, Dec. 2011.

[13] I. D. Chivers and J. Sleightholme, "Parallel programming," *ACM Sigplan Fortran Forum*, vol. 24, pp. 15–20, Dec. 2005.

[14] K.-C. Li, H.-C. Chang, C.-T. Yang, L. M. Sato, C.-Y. Yang, Y.-Y. Wu, M. Pel, H.-K. Liao, M.-C. Hsieh, and C.-W. Tsai, "On construction of a visualization toolkit for MPI parallel programs in cluster environments," in *Proc. 19th Int. Conf. Adv. Inf. Netw. Appl. (AINA) (AINA Papers)*, vol. 1, Mar. 2005, pp. 211–214.

[15] T. Berredjem and M. Benidir, "Bearing faults diagnosis using fuzzy expert system relying on an improved range overlaps and similarity method," *Expert Syst. Appl.*, vol. 108, pp. 134–142, Oct. 2018.

[16] Y. Ying, J. Li, Z. Chen, and J. Guo, "Study on rolling bearing on-line reliability analysis based on vibration information processing," *Comput. Electr. Eng.*, vol. 69, pp. 842–851, Jul. 2018.

[17] J. Wang, P. Fu, L. Zhang, R. X. Gao, and R. Zhao, "Multilevel information fusion for induction motor fault diagnosis," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 5, pp. 2139–2150, Oct. 2019.

[18] C. Grover and N. Turk, "Rolling element bearing fault diagnosis using empirical mode decomposition and hjorth parameters," *Procedia Comput. Sci.*, vol. 167, pp. 1484–1494, Jan. 2020.

[19] X. Zhang, Y. Liang, J. Zhou, and Y. Zang, "A novel bearing fault diagnosis model integrated permutation entropy, ensemble empirical mode decomposition and optimized SVM," *Measurement*, vol. 69, pp. 164–179, Jun. 2015.

[20] X. Kong, G. Mao, Q. Wang, H. Ma, and W. Yang, "A multi-ensemble method based on deep auto-encoders for fault diagnosis of rolling bearings," *Measurement*, vol. 151, Feb. 2020, Art. no. 107132.

[21] H. Shao, H. Jiang, F. Wang, and Y. Wang, "Rolling bearing fault diagnosis using adaptive deep belief network with dual-tree complex wavelet packet," *ISA Trans.*, vol. 69, pp. 187–201, Jul. 2017.

[22] Y. Ding, L. Ma, J. Ma, M. Suo, L. Tao, Y. Cheng, and C. Lu, "Intelligent fault diagnosis for rotating machinery using deep Q-network based health state classification: A deep reinforcement learning approach," *Adv. Eng. Informat.*, vol. 42, Oct. 2019, Art. no. 100977.

[23] X. Yan, Y. Liu, and M. Jia, "Multiscale cascading deep belief network for fault identification of rotating machinery under various working conditions," *Knowl.-Based Syst.*, vol. 193, Apr. 2020, Art. no. 105484.

[24] Y.-Y. Wu and C.-M. Tsai, "Pedestrian, bike, motorcycle, and vehicle classification via deep learning: Deep belief network and small training set," in *Proc. Int. Conf. Appl. Syst. Innov. (ICASI)*, May 2016, pp. 1–4.

[25] H. Wang, Z. Lei, X. Zhang, B. Zhou, and J. Peng, "A review of deep learning for renewable energy forecasting," *Energy Convers. Manage.*, vol. 198, Oct. 2019, Art. no. 111799.

[26] F. Ghasemi, A. Mehridehnavi, A. Fassihi, and H. Pérez-Sánchez, "Deep neural network in QSAR studies using deep belief network," *Appl. Soft Comput.*, vol. 62, pp. 251–258, Jan. 2018.

[27] M. Ma, X. Xu, J. Wu, and M. Guo, "Design and analyze the structure based on deep belief network for gesture recognition," in *Proc. 10th Int. Conf. Adv. Comput. Intell. (ICACI)*, Mar. 2018, pp. 40–44.

[28] P. Wei, Y. Z. Li, Z. Zhang, T. Hu, Z. Li, and D. Liu, "An optimization method for intrusion detection classification model based on deep belief network," *IEEE Access*, vol. 7, pp. 87593–87604, 2019.

[29] Z. Wang, T. Sun, and X. Tian, "Fault diagnosis of rolling bearing based on SDAE and PSO-DBN," in *Proc. Chin. Control Decis. Conf. (CCDC)*, Jun. 2019, pp. 624–629.

[30] C.-W. Tsai, C.-H. Hsia, S.-J. Yang, S.-J. Liu, and Z.-Y. Fang, "Optimizing hyperparameters of deep learning in predicting bus passengers based on simulated annealing," *Appl. Soft Comput.*, vol. 88, Mar. 2020, Art. no. 106068.

[31] M. Randall and A. Lewis, "A parallel implementation of ant colony optimization," *J. Parallel Distrib. Comput.*, vol. 62, no. 9, pp. 1421–1432, Sep. 2002.

[32] C. Bekas, E. Kokiopoulou, E. Callopoulos, and V. Simoncini, "Parallel computation of pseudospectra using transfer functions on a MATLAB-MPI cluster platform," in *Proc. 9th Eur. Parallel Virtual Mach. Message Passing Interface Group Meeting*. London, U.K.: Springer-Verlag, vol. 2474, Sep. 2002, pp. 199–207.

[33] S. G. Żurek, "Genetic algorithm with peaks adaptive objective function used to fit the EPR powder spectrum," *Appl. Soft Comput.*, vol. 11, no. 1, pp. 1000–1007, Jan. 2011.

[34] L. Zhao, Y. Zhou, H. Lu, and H. Fujita, "Parallel computing method of deep belief networks and its application to traffic flow prediction," *Knowl.-Based Syst.*, vol. 163, pp. 972–987, Jan. 2019.

[35] G. Cybenko, "Parallel computing for machine learning in social network analysis," in *Proc. IEEE Int. Parallel Distrib. Process. Symp. Workshops (IPDPSW)*, May 2017, pp. 1464–1471.

[36] I.-H. Chung, T. N. Sainath, B. Ramabhadran, M. Picheny, J. Gunnels, V. Austel, U. Chauhari, and B. Kingsbury, "Parallel deep neural network training for big data on blue Gene/Q," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 6, pp. 1703–1714, Jun. 2017.

[37] G. E. Hinton, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[38] J. Song, S. Qin, and P. Zhang, "Chinese text categorization based on deep belief networks," in *Proc. IEEE/ACIS 15th Int. Conf. Comput. Inf. Sci. (ICIS)*, Jun. 2016, pp. 1–5.

[39] C. Zhang, L. Xu, X. Li, and H. Wang, "A method of fault diagnosis for rotary equipment based on deep learning," in *Proc. Prognostics Syst. Health Manage. Conf. (PHM-Chongqing)*, Oct. 2018, pp. 958–962.

[40] H. Zhao, F. Liu, H. Zhang, and Z. Liang, "Research on a learning rate with energy index in deep learning," *Neural Netw.*, vol. 110, pp. 225–231, Feb. 2019.

[41] C. Yu, X. Qi, H. Ma, X. He, C. Wang, and Y. Zhao, "LLR: Learning learning rates by LSTM for training neural networks," *Neurocomputing*, vol. 394, pp. 41–50, Jun. 2020.

[42] S. Ramachandran, M. L. Jayalal, A. Riyas, R. Jehadeesan, and K. Devan, "Application of genetic algorithm for optimization of control rods positioning in a fast breeder reactor core," *Nucl. Eng. Design*, vol. 361, May 2020, Art. no. 110541, doi: 10.1016/j.nucengdes.2020.110541.

[43] H. Karimi and F. Yousefi, "Application of artificial neural network-genetic algorithm (ANN-GA) to correlation of density in nanofluids," *Fluid Phase Equilibria*, vol. 336, pp. 79–83, Dec. 2012.

[44] H. Shekofteh and A. Masoudi, "Determining the features influencing theS soil quality index in a semiarid region of iran using a hybrid GA-ANN algorithm," *Geoderma*, vol. 355, Dec. 2019, Art. no. 113908.

[45] Y. Zhang, G. Pan, B. Chen, J. Han, Y. Zhao, and C. Zhang, "Short-term wind speed prediction model based on GA-ANN improved by VMD," *Renew. Energy*, vol. 156, pp. 1373–1388, Aug. 2020, doi: 10.1016/j.renene.2019.12.047.

CHAOZHONG GUO was born in Henan, China, in 1982. He received the B.S. degree in mechanical engineering and automation from Zhengzhou University, in 2007, and the M.S. and Ph.D. degrees in mechanical engineering from the Harbin Institute of Technology, in 2009 and 2014, respectively. From 2010 to 2012, he studied at the University of Illinois at Urbana–Champaign as a joint-training Ph.D. Student. From 2014 to 2017, he was an Assistant Professor with the School of Mechatronics Engineering, Harbin Institute of Technology, where he has been an Associate Professor with the School of Mechatronics Engineering, since 2018. His research interests include data processing, diagnosis, rotor dynamics, and intelligent manufacturing.

LIN LI was born in Shandong, China, in 1987. He received the Ph.D. degree from the School of Mechatronics Engineering, Harbin Institute of Technology, in 2016. He was a Visiting Researcher with Purdue University, USA, from 2014 to 2015. He is currently an Assistant Professor with the College of Electromechanical Engineering, Qingdao University of Science and Technology. He has authored or coauthored 15 research articles. His research interests include computer-aided intelligent manufacturing, energy consumption modeling, and sustainable manufacturing.

YUANYUAN HU was born in Jiangxi, China, in 1994. She received the B.S. degree in mechanical engineering from Harbin Engineering University, Harbin, China, in 2017, and the M.S. degree in mechanical engineering from the Harbin Institute of Technology, Harbin, in 2019. Her research is mainly about the intelligent diagnosis of rotating equipment.

JIHONG YAN was born in 1972. She received the Ph.D. degree from the Harbin Institute of Technology, China, in 1999. She was a Postdoctoral Researcher with Tsinghua University, from 1999 to 2001, the University of Wisconsin, from 2001 to 2004, and Pennsylvania State University, from 2004 to 2005. She is currently a Professor with the Harbin Institute of Technology. She has authored or coauthored over 100 research articles and published three books. Her two articles were ranked as the ESI highly cited articles. As the PI, she has worked on and accomplished 20 projects in sustainability and intelligent manufacturing-related areas, funded by the NSF of China (NSFC), the NSF–NSFC Joint-Project, the National High-Tech Project, and high-tech funding projects from industries. Her research interests include industrial big data, prognostics, sustainable manufacturing, and intelligent manufacturing.