

Automatic Feature Extraction and Construction Using Genetic Programming for Rotating Machinery Fault Diagnosis

Bo Peng, Shuting Wan, Ying Bi, *Member, IEEE*, Bing Xue, *Member, IEEE*, and Mengjie Zhang, *Fellow, IEEE*

Abstract—Feature extraction is an essential process in the intelligent fault diagnosis of rotating machinery. Although existing feature extraction methods can obtain representative features from the original signal, domain knowledge and expert experience are often required. In this paper, a novel diagnosis approach based on evolutionary learning, namely automatic feature extraction and construction using genetic programming (AFECGP), is proposed to automatically generate informative and discriminative features from original vibration signals for identifying different fault types of rotating machinery. To achieve this, a new program structure, a new function set and a new terminal set are developed in AFECGP to allow it to detect important subband signals, extract and construct informative features, automatically and simultaneously. More importantly, AFECGP can produce a flexible number of features for classification. Having the generated features, *k*-Nearest Neighbors is employed to perform fault diagnosis. The performance of the AFECGP based fault diagnosis approach is evaluated on four fault diagnosis datasets of varying difficulty and compared with fourteen baseline methods. The results show that the proposed approach achieves better fault diagnosis accuracy on all the datasets than the competitive methods and can effectively identify different fault conditions of rolling bearing, gear, and rotor.

Index Terms—Genetic Programming, Feature Extraction, Feature Construction, Fault Diagnosis, Rotating Machinery

I. INTRODUCTION

ROTATING machinery, such as electric motors, turbine generators, and aero engines, are the important components of modern industry and have been widely used in the aspects of production and daily life [1]. Because these machines often work under unstable loads and extreme operating temperatures, the critical parts of which inevitably have various faults, such as bearing damage, gear breakage,

and rotor imbalance. Damaged parts can prone to cause the machines to be scrapped, leading to huge economic losses and even serious accidents [2]. Therefore, fault diagnosis and condition monitoring of rotating machinery are important for reducing operation and maintenance costs, ensuring production safety, and prolonging the life of machines.

A fault diagnosis and condition monitor system generally contains three steps: data acquisition, feature extraction, and pattern recognition [3]. Because the acquisition of vibration signals can be easily achieved by installing speed or acceleration sensors on the outer surface of a machine, vibration signal based diagnostic and monitor methods have gained much attention. Due to the influence of transmission path and noise, the collected vibration signals are often non-linear and of multi-component, and need to be preprocessed to extract effective features for performing pattern recognition. Decomposing the vibration signals into multiple subband signals can reduce the difficulty of analysis. Wavelet Transform and Wavelet Packet Transform can decompose signals into a set of wavelet details, but the performance is affected by the predefined wavelet basis and levels [4]. The recursive decomposition methods on the basis of the signal waveforms, such as Empirical Mode Decomposition (EMD), Intrinsic Time-scale Decomposition, Local Mean Decomposition, and their improved versions [5], [6], [7], [8], [9], [10], have been developed and used to decompose signals. But the most obvious flaw of these methods is prone to modal aliasing. To address this problem, Variational Modal Decomposition [11], [12], Swarm Decomposition [13], [14], and Time Varying Filtering-Based EMD [15], [16] have been developed and applied to rotating machinery fault diagnosis. Although these methods are effective in decomposing the complex vibration signals into simple subband signals, not all the information of subband signals is important and useful for fault diagnosis. Therefore, expert experience is often need to verify and manually tune these signals to find the important ones.

To accurately describe the dynamic characteristics of the machines under different operation states, it is necessary to generate effective features from the decomposed vibration signals. Time-domain statistic features (i.e., mean value, standard deviation, skewness, and kurtosis), frequency-domain statistic features (i.e., mean frequency, standard deviation frequency, frequency centre, and root mean square frequency), have been used to represent vibration signals and achieved promising results in fault diagnosis of bearing and rotor [17], [18]. Because the characteristics of the mechanical equipment changes

This work was supported in part by the National Natural Science Foundation of China under grant 51777075, the Natural Science Foundation of Hebei Province under grant E2019502064, the Fundamental Research Funds for Central Universities under grant 2019QN131, the Marsden Fund of New Zealand Government under Contracts VUW1509 and VUW1615, the Science for Technological Innovation Challenge (SfTI) fund under grant E3603/2903, the University Research Fund at Victoria University of Wellington grant number 223805/3986, MBIE Data Science SSIF Fund under the contract RTVU1914, and National Natural Science Foundation of China (NSFC) under Grant 61876169. The work of Bo Peng was supported by the Joint Postgraduate Training Program of North China Electric Power University.

B. Peng and S. Wan are with Hebei Key Laboratory of Electric Machinery Health Maintenance & Failure Prevention, North China Electric Power University, Baoding 071000, China (e-mail: ncepupengbo@163.com, wanshuting1@sina.com).

Y. Bi, B. Xue and M. Zhang are with the School of Engineering and Computer Science, Victoria University of Wellington, Wellington 6140, New Zealand (e-mail: ying.bi@ecs.vuw.ac.nz, bing.xue@ecs.vuw.ac.nz, mengjie.zhang@ecs.vuw.ac.nz).

with the occurrence of faults, the entropy index can be used to detect the dynamic changes. Fuzzy Entropy (FE), Sample Entropy (SE), Permutation Entropy (PE), Symbolic Dynamic Entropy (SDE), and Dispersion Entropy (DE), have been adopted to characterize vibration signal for fault diagnosis [19], [20], [21], [22], [23]. These entropy indexes estimate the complexity or irregularity of the vibration signals using a single scale. Multi-scale analysis has been introduced and multi-scale entropy based methods (i.e., multi-scale SE, multi-scale FE, multi-scale PE, multi-scale SDE, and multi-scale DE [24], [25], [26], [23], [27]) have been proposed to extract multi-scale features from the vibration signals. Because there is typically information redundancy between multiple features, feature selection techniques, such as Laplacian score, ReliefF and Max-relevance and Min-redundancy, have been introduced to reduce the number of features to ensure that diagnostic performance is not significantly affected by redundant information [17], [18], [27], [28]. The extracted or selected features are fed into a classification algorithm to train a classifier for classification. Machine learning classification algorithms, such as k -Nearest Neighbours (KNN), Support Vector Machine (SVM), Multilayer Perception (MLP), Decision Tree (DT), and Logistic Regression (LR), have been applied to fault diagnosis of rotating machinery [17], [18], [29], [30], [31]. Besides, Deep Neural Networks (DNNs) based methods have also been developed to fault diagnosis tasks [32], [33], [34].

Although many methods have been developed for fault diagnosis of rotating machinery, they still have drawbacks. These drawbacks are summarized as follows.

- 1) Most existing fault diagnosis methods need multiple independent steps, such as signal preprocessing, feature extraction and feature selection, to generate effective features for fault diagnosis. The results of the previous step often significantly affect the results of the later step. For example, if the extracted features are not effective for a task, feature selection may not be helpful as well.
- 2) Most existing fault diagnosis methods have poor adaptability and robustness [1], which may only be effective for a certain fault diagnose task.
- 3) The DNN based fault diagnosis methods have poor interpretability and often require a large number of training instances. However, in many application of rotational machinery, it is very difficult to obtain a large number of training instances.
- 4) Rich domain knowledge are often required in the process of fault diagnose, such as the determination of useful subband signals, the selection of informative features and the determination of the number of features.

Genetic Programming (GP) is an evolutionary algorithm, which can automatically evolve computer programs for dealing with tasks without human intervention and domain knowledge [35]. GP can evolve tree-based solutions with variable lengths and the solutions often provide high interpretability. The solutions evolved by GP are often creative and even not considered by human experts. GP has been successfully applied to many tasks, including symbolic regression, job shop scheduling and image analysis, and achieved promising results [36], [37], [38],

[28], [39], [40]. Many successful applications show that GP has a great potential to simultaneously perform information detection, feature extraction, feature selection and feature construction for classification [41], [42], [43]. Although GP has been applied to various fields, very few work apply GP for fault diagnosis [44], [45]. To the best of our knowledge, GP has not been applied to automatically and simultaneously detect useful subband signals from the original signals, extract features from the detected subband signals and construct high-level features for effective fault diagnosis.

The overall goal of this paper is to develop a new intelligent fault diagnosis approach based on GP for automatic feature extraction and construction in identifying different fault types of rotating machinery. To achieve this, a new program structure/architecture, a new function set and a new terminal set are developed for AFECGP to generate informative and discriminative features from raw signals by automatically evolving solutions with multiple transformation functions, i.e., decomposition, rescaling, feature extraction, feature construction, and feature combination functions. The AFECGP approach can generate variable numbers of features for classification for different fault diagnosis tasks. The performance of AFECGP will be examined on four fault diagnosis tasks/problems of varying difficulty and compared with fourteen benchmark methods.

The main contributions of this paper are summarized in the following four aspects.

- 1) A new rotating machinery intelligent fault diagnosis approach based on GP, i.e., AFECGP, is proposed. This approach can automatically and simultaneously extract and construct informative discriminative features from the vibration signals under different running states. AFECGP can evolve (nearly) optimal solutions not considered by domain experts. The learning process of AFECGP does not need extra domain expertise or human intervention.
- 2) A new program architecture is designed based on a tree-based representation that contains seven function layers, i.e., an input layer, a decomposition layer, a rescaling layer, a feature extraction layer, a feature construction layer, a feature combination layer, and an output layer. Based on this, AFECGP can adaptively achieve the detection of useful subband signals, features extraction and construction, and determination of the number of features.
- 3) Extensive experiments are conducted to show the effectiveness of the proposed AFECGP approach in comparisons with fourteen baseline methods for fault diagnosis. Compared with the methods using manually extracted features and the other GP based methods, the AFECGP approach can achieve better performance on all the four different diagnosis datasets of varying difficulty. The results also show that the AFECGP approach has high flexibility and adaptability to different fault tasks/problems in diagnosis of rotating machinery.
- 4) Further analysis is conducted to deeply understand the proposed AFECGP approach. The solutions evolved by AFECGP can be easily visualized to provide insights on which part of the input signal is representative, which features are extracted and used to construct high-level

features, and how many features are needed to represent the input signals. In addition, the constructed high-level features are also visualized to show why AFECGP can achieve better performance.

II. RELATED WORK

A. Genetic Programming (GP)

GP is an evolutionary learning algorithm, which inherits the basic idea of Darwin's theory of natural evolution, that is, breeding offspring by selecting good individuals from their parents [35]. An individual of GP is typically a computer program expressed using a tree structure, which is a variable-length encoding scheme. A GP tree is composed of internal nodes and leaf nodes. The internal nodes are functions or operators selected from the function set and the leaf nodes are arguments/features and constants selected from the terminal set. Fig. 1 shows an example program of GP, which represents a mathematic expression $(y-2\times x)+(x+6)$. In this program, the internal nodes, such as $+$, $-$ and \times , belong to the function set, and the leaf nodes, such as arguments x and y and constants 2 and 6, belong to the terminal set.

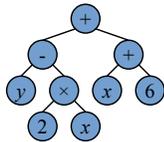


Fig. 1. An example tree of GP.

GP can automatically evolve trees/solutions to deal with a task. The overall process of GP starts with randomly initializing a population of computer programs in the search space. Each individual (program) in the population is evaluated by a fitness function and assigned with a fitness value. During the evolutionary process, a new population of individuals is generated through the *Elitism*, *Crossover*, and *Mutation* operations, at each generation. The *Elitism* operation copies the best individuals from the current generation to the next generation. Based on the *Selection* operation, the individuals with better fitness values have a higher chance to be chosen for the *Crossover* and *Mutation* operations. The *Crossover* and *Mutation* operations are illustrated in Fig. 2 and Fig. 3, respectively. As it can be seen from Fig. 2, in the *Crossover* operation, the subtrees (marked by red color) of parent 1 and parent 2 are exchanged to generate offspring 1 and offspring 2. As it can be seen from Fig. 3, the *Mutation* operation is that randomly selecting a mutation node, the original subtree (marked by red color) at this node is deleted and a new subtree (marked by yellow color) is grown from this node. The process of fitness evaluation is executed on all the individuals of each generation. The overall evolutionary process is terminated when a termination criterion is satisfied. After the evolutionary process, the optimal/best individual is obtained and returned.

GP contains two types of program structure, including Loosely Typed GP (LTGP) and Strongly Typed GP (STGP) [46]. The input and output types of LTGP tree nodes are the same, which do not need to be specified. In contrast, the input and output types of STGP tree nodes are typically different and need to be specified. In STGP, if the output type of a

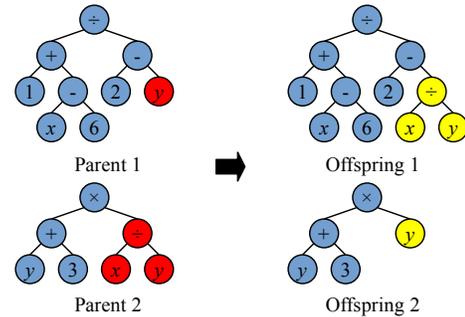


Fig. 2. Crossover operation.

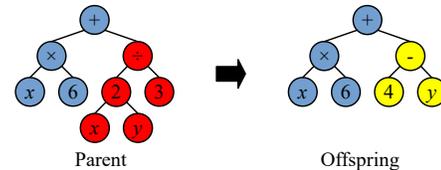


Fig. 3. Mutation operation.

subtree and the input type of its parent node are incompatible, the trees with such a connection are not allowed.

B. GP for Feature Extraction and Construction

In recent years, feature extraction methods based on GP have been successfully proposed and applied to different tasks, such as image classification [36]. Benefiting from the flexible representation of GP, many image-related operators, such as the convolution operator, histogram of orientated gradient, the mean filter, the Gaussian filter, the Gabor filter, and the Sobel filter, have been integrated into the GP programs as functions to extract/learn informative features from raw images [43], [47]. Atkins *et al.* [48] proposed a three-tier GP method, where the mean, max, min, median, and standard deviation of image pixel values are extracted as features for image classification. Al-Sahaf *et al.* [49] proposed a two-tier GP method to extract features from the image region of line, square, rectangle, or circle shape. Bi *et al.* [50] proposed a multi-layer GP method for binary image classification. This method can perform region detection, feature extraction, feature construction and image classification, simultaneously. Bi *et al.* [43] proposed a GP method by selecting and combining five image descriptors to extract global/local features for image classification. To the best of our knowledge, very few work has been reported on GP based feature extraction methods for fault diagnosis of rotating machinery. Guo *et al.* [44] and Xuan *et al.* [45] applied GP based feature extraction methods to fault diagnosis of rolling bearing and gear, where the high order of moments and the frequency features of raw vibration signals are manually extracted. However, neither of them have considered the features of certain frequency bands of signals, which are typically more effective for fault diagnosis than the features of full-band signals.

The feature space transformation is useful for improving the performance of a classification algorithm [51]. Feature construction is one type of feature transformation, which can map the original features into another space for representation [52]. In [44], [45], [53], one high-level feature was constructed

by GP using the original features and the experimental results showed that the constructed feature can improve the classification accuracy. In [54], [55], [56], GP was employed to construct multiple features and the results showed that the constructed features achieve better classification performance than the methods using the original features and the single constructed feature. However, when constructing multiple features, it is necessary to set the number of constructed features, which requires domain knowledge.

Motivated by the above limitations, this study develops a new GP based approach to automatically extracting and constructing a flexible number of features for fault diagnosis of rotating machinery. The new approach will be described in the next section.

III. PROPOSED APPROACH

In this section, a new GP based approach, i.e., AFECGP, is proposed to automatically and simultaneously extract and construct features from raw signals for identifying different fault types of rotating machinery. The proposed AFECGP approach, i.e., the new program structure, the new function set, the new terminal set, and fitness evaluation, are introduced in detail. The detailed process of the AFECGP based fault diagnosis approach is presented.

A. Algorithm Overview

The overall algorithm of AFECGP is described in Algorithm 1. The input of AFECGP is the training set, having the raw signals X_{signal} and the class labels X_{label} . The output of AFECGP is the best program/individual, i.e., $Best_tree$. P_g represents the population of the g th generation. g starts from 0 and increases by 1 with the generation. The AFECGP algorithm begins with randomly generating the initial population P_0 according to the program structure, the function set and the terminal set. Each individual i of the population is evaluated using a fitness function and assigned with a fitness value. At each generation, the best individuals Q_g are selected from P_g through the *Elitism* operation and copied to the next generation. Based on the *Selection* operation, the individuals S_g with better fitness values in P_g are chosen as parents for breeding offspring. The offspring O_g are obtained by performing the *Crossover* and *Mutation* operations. Q_g and O_g are combined to form the new population of the next generations. The new population P_g will be evaluated and evolved again. The best solution is updated at each generation. When g is equal to the maximal number of generations N_{gen} , the evolutionary process stops and the best solution $Best_tree$ is returned as the output of AFECGP.

B. Program Structure

To achieve simultaneous feature extraction and construction in a single GP program, a new program structure is developed based on STGP for handling multiple data types. The left part of Fig. 4 shows the new program structure and an example program that can be evolved by the AFECGP approach. The new program structure has the input, decomposition, rescaling,

Algorithm 1: Algorithm framework of AFECGP

Input: The training set: X_{signal} and X_{label}
Output: The best individual: $Best_tree$

```

1 begin
2    $g \leftarrow 0$ ;
3    $P_0 \leftarrow$  Randomly generate the initial population;
4   for Each  $i$  in  $P_0$  do
5      $f_i \leftarrow$  Calculate the fitness value of  $i$ ;
6   end
7   Update  $Best\_tree$  according to  $P_0$ ;
8   while  $g < N_{gen}$  do
9      $Q_g \leftarrow$  Copy the best individuals from  $P_g$  by Elitism;
10     $S_g \leftarrow$  Select individuals from  $P_g$  as parents for breeding offspring by Selection;
11     $O_g \leftarrow$  Generate offspring from  $S_g$  by Crossover and Mutation;
12     $P'_g \leftarrow Q_g + O_g$ ;
13     $g \leftarrow g + 1$ ;
14     $P_g \leftarrow P'_g$ ;
15    for Each  $i$  in  $P_g$  do
16       $f_i \leftarrow$  Calculate fitness value of  $i$ ;
17    end
18    Update  $Best\_tree$  according to  $P_g$ ;
19  end
20 end
21 Return  $Best\_tree$ .
```

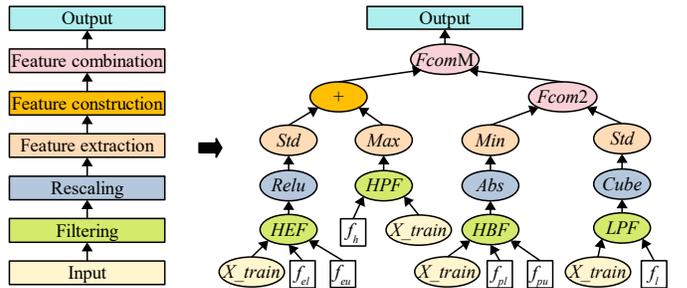


Fig. 4. The program structure of AFECGP and an example program that can be evolved by AFECGP.

feature extraction, feature construction, feature combination, and output layers. These layers are connected in the bottom-up manner. Except for the input and output layers, the other layers have many specific functions for different purposes. The input layer takes raw signals as inputs. The decomposition layer contains filter functions, which operate on the raw signal to reduce noise interference and obtain useful subband signals. The rescaling layer is to change the ranges of the subband signal amplitude and enhance the data characteristics by using transformation functions. The feature extraction layer is to extract statistical features from the subband signals. The feature construction layer is to construct the extracted features into high-level features. The feature combination layer is to generate a feature vector, which contains multiple features and can describe the input signal comprehensively. The output layer returns the generated features as outputs for fault diagnosis.

C. Function Set

The new function set of AFECGP has decomposition functions, rescaling functions, feature extraction functions, and feature combination functions for the corresponding layer. Table I summarizes the functions of each layer and their inputs, outputs and descriptions.

TABLE I
FUNCTION SET

Function	Input	Output	Description
<i>HPF</i>	1 signal, f_h	1 signal	Perform high pass filtering to signal
<i>LPF</i>	1 signal, f_l	1 signal	Perform low pass filtering to signal
<i>BPF</i>	1 signal f_{pu}, f_{pl}	1 signal	Perform band pass filtering to signal
<i>BEF</i>	1 signal f_{eu}, f_{el}	1 signal	Perform band stop filtering to signal
<i>Abs</i>	1 signal	1 signal	Return $abs(x)$ for each x in the signal
<i>ReLU</i>	1 signal	1 signal	Return $max(0, x)$ for each x in the signal
<i>Sqrt</i>	1 signal	1 signal	Return \sqrt{x} for each x in the signal. Return 0 if $x < 0$.
<i>Square</i>	1 signal	1 signal	Return x^2 for each x in the signal
<i>Cube</i>	1 signal	1 signal	Return x^3 for each x in the signal
<i>Min</i>	1 signal	1 feature	Calculate minimum of the signal
<i>Max</i>	1 signal	1 feature	Calculate maximum of the signal
<i>Std</i>	1 signal	1 feature	Calculate standard deviation of the signal
<i>Mean</i>	1 signal	1 feature	Calculate average value of the signal
<i>KthM</i>	1 signal	1 feature	Calculate k th-order moment to signal
+	2 features	1 feature	Add two features
-	2 features	1 feature	Subtract two features
×	2 features	1 feature	Multiply two features
÷	2 features	1 feature	Divide two features. Return 0 if the divisor is 0.
<i>Fcom2</i>	2 features	1 vector	Concatenate two features into a vector
<i>FcomM</i>	2 vectors or 1 vector/feature	1 vector	Concatenate two vectors or a vector and a feature into a vector

1) *Decomposition Functions*: Four functions, i.e., High-Pass Filter (*HPF*), Low-Pass Filter (*LPF*), Band-Pass Filter (*BPF*), and Band-Elimination Filter (*BEF*) [57], are used in the decomposition layer. The inputs of these functions are the raw signals and the function parameters. These filter functions can be used to split the original signal into the subband signals and reduce the noise in the original signal. These filtering functions are implemented based on the *Butterworth* filter [58]. The *HPF* function has the parameters of the high-pass cutoff frequency f_h . The *LPF* function has the parameters of the low-pass cutoff frequency f_l . The *BPF* function has the parameter of the lower and the upper cutoff frequencies (f_{pl} and f_{pu}) of the passband. The *BEF* function has the parameter of the lower and the upper cutoff frequencies (f_{el} and f_{eu}) of the stopband. These parameters (i.e., $f_h, f_l, f_{pu}, f_{pl}, f_{eu}$ and f_{el}) are designed as the terminals of AFECGP, and their optimal values can be automatically learned/determined during the evolutionary process.

2) *Rescaling Functions*: The rescaling layer contains the *Abs*, *ReLU*, *Sqrt*, *Square*, and *Cube* functions that can change the value range of the input signal. The *Sqrt* function is protected by returning 1 if the signal value is negative. The input of each rescaling function is the decomposed signal.

3) *Feature Extraction Functions*: The *Max*, *Min*, *Std*, *Mean*, and *NthM* functions are used to calculate the statistical features of the signal at the feature extraction layer of AFECGP. The inputs of these functions may be the signal processed by the decomposition functions, or the signal processed by both decomposition and rescaling functions. This design allows AFECGP to produce effective features that can describe the signal comprehensively. The output of the *Max*,

TABLE II
TERMINAL SET

Terminal	Type	Description
<i>Signal</i>	Array	Raw signal with a length of M
n	Integer	Order parameter of the N thM function. It is in the range of $n \in [2, 4]$
f_l, f_h	Integer	Cutoff frequency of <i>HPF</i> and <i>LPF</i> . Its value is in the range of $[0, f'_s]$ with a step of $\text{floor}(f'_s/100)$
f_{pl}, f_{pu}	Integer	Upper and lower frequency of <i>BPF</i> . Its value is in the range of $[0, f'_s]$ with a step of $\text{floor}(f'_s/100)$. If $f_{pl} > f_{pu}$, swap f_{pl} and f_{pu}
f_{el}, f_{eu}	Integer	Upper and lower frequency of <i>BEF</i> . Its value is in the range of $[0, f'_s]$ with a step of $\text{floor}(f'_s/100)$. If $f_{el} > f_{eu}$, swap f_{el} and f_{eu}

Min, *Std*, *Mean*, and *KthM* functions, is the maximum, minimum, standard deviation, average and k th-order moment (i.e., $m_x^{(k)} = \frac{1}{N} \sum_{i=1}^N x_i^k$) [44] of the signal, respectively. These features may be used directly to describe the signal or further constructed to generate high-level features for fault diagnosis.

4) *Feature Construction Functions*: The +, -, ×, and ÷ functions are used to construct high-level features at the feature construction layer. The functions are the commonly used functions in GP for feature construction [55]. The inputs of these functions may be the features obtained from the feature extraction layer or the constructed features obtained from the feature construction layer. That is, the constructed features can be used to construct various levels of features. This design allows AFECGP to construct complex high-level features that are effective for addressing the task.

5) *Feature Combination Functions*: The *Fcom2* and *FcomM* functions are used to combine the extracted or constructed features into a feature vector at the feature combination layer. The *Fcom2* function has two child nodes, which can be the feature extraction functions or the feature construction functions. The *FcomM* function also has two child nodes, which can be the feature extraction functions, the feature construction functions or the *Fcom2* function. If one single feature can accurately describe the signal, these two feature combination functions will not be used in a GP tree. With this design, the outputs of a feature construction layer may be one single feature or a vector of multiple features. This design allows AFECGP to produce a dynamic number of features for fault diagnosis.

D. Terminal Set

In AFECGP, eight terminals are employed in the terminal set as listed in Table II. The *Signal* terminal represents the input raw signal, which is a 1-D array with a length of m . The n terminal represents the order of the *KthM* function, which is an integer in the range of $[2, 4]$. The 1st-order moment is equal to *Mean*. The $f_h, f_l, f_{pu}, f_{pl}, f_{eu},$ and f_{el} terminals represent the parameters of the *HPF*, *LPF*, *BPF*, and *BEF* filter functions, respectively. They are integers in the range of $[0, f'_s]$ with a step of $\text{floor}(f'_s/100)$. f_s is the sample frequency determined by different fault datasets. f'_s represents the half of f_s , i.e., $f'_s = f_s/2$. The values of these terminals except for *Signal* are automatically selected/determined from their ranges during the evolutionary process.

E. Fitness Evaluation

The fitness value of a program/individual is obtained from the fitness evaluation process, indicating the performance of each evolved individual. In the fitness evaluation process, KNN is employed to evaluate the classification/diagnosis performance of the features generated by AFECGP. KNN is a simple classification algorithm and easy to implement, which has been widely used [59], [60]. Compared with other classification algorithms, such as SVM, MLP, DT, and LR, KNN treats each feature equally without any feature weighting or selection. With the use of KNN, the proposed GP approach can automatically extract and construct important features, and avoid redundant or irrelevant features.

The evaluation process starts with using the individuals of AFECGP to transform each signal in the training set into features. To ensure the classification performance of KNN is not affected by singular values or feature bias, the min-max normalization method is used to transform the values of these features into the range of [0, 1]. In addition, the stratified 5-fold cross-validation method is employed for improving the generalization ability of the features constructed by AFECGP. That is, these normalized features and their labels are split into five folds with a (nearly) equal size. Then one fold is used as the test set and the remaining four folds are used as the training set. KNN is fed using the training set and evaluated using the test set. This process repeats five times until all the five folds are used as the test set exactly once. The average test classification results of the five folds is used as the fitness value of the individual of AFECGP.

F. Overall AFECGP Based Fault Diagnosis Approach

In order to accurately identify different fault types of rotating machinery, a novel fault diagnosis approach based on AFECGP is proposed to automatically and simultaneously extract and construct high-level features from the original vibration signals for fault classification. Fig. 5 shows the flowchart of the AFECGP based fault diagnosis approach, which has three steps as follows.

- 1) Signal collection and dataset formulation. The vibration signals of different running states are collected from the rotating machines. Randomly selecting a part of vibration signals to form the training set, and the remaining signals are used as the test set.
- 2) AFECGP based feature extraction and construction. The signal features are extracted and constructed from the original signals using AFECGP. In AFECGP, each individual can produce a flexible number of constructed features for fault diagnosis. The AFECGP approach searches for the best individual that can achieve the best performance on the training set using a population of individuals via an evolutionary process. The evolutionary process includes the processes of fitness evaluation and population generations using genetic operators, which have been introduced in previous subsections. The overall evolutionary process stops when the termination criterion (i.e., reaching the maximal number of generations) is satisfied. When the evolutionary process is terminated, the best individual is

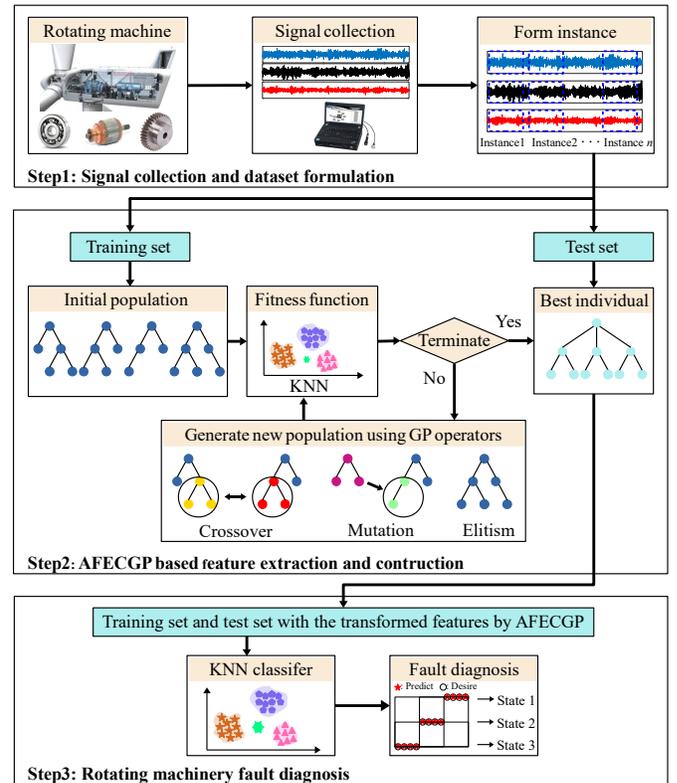


Fig. 5. Flowchart of the AFECGP based rotating machine fault diagnosis.

returned. The vibration signals of the training and tests sets are transformed into features using the best individual found by AFECGP.

- 3) Rotating machinery fault diagnosis. These features extracted and constructed by AFECGP are normalized into the range of [0, 1]. The training set with the transformed features and the corresponding class labels (running states) are fed into KNN as the training set and the class label of each instance in the test set can be predicted by finding the nearest neighbors via KNN.

IV. EXPERIMENT DESIGN

In this section, the proposed AFECGP approach is examined on four rotating machinery datasets, i.e., two bearing fault datasets, one gearing fault dataset and one rotor fault dataset. To show the effectiveness of AFECGP, fourteen baseline methods are used for comparisons. The detailed experiment design, i.e., fault datasets, baseline methods and parameter settings, are presented in this section.

A. Fault Datasets

CWRU is a rolling bearing fault dataset collected by Case Western Reserve University (CWRU) [61]. The CWRU dataset has been widely used to validate the effectiveness of the diagnosis methods. Vibration signals are collected by the accelerometers placed at the motor drive end of test rig under four different load conditions: i.e., 0, 1, 2, and 3 HP. Four kinds of healthy bearings, i.e., Normal (NOR), Inner Ring Fault (IRF), Outer Ring Fault (ORF), and Rolling Element

Fault (REF), are used in the experiments. Each fault type contains three different fault levels, i.e., 0.007, 0.014 and 0.021 inches, respectively. Ten health states (i.e., *NOR*, *IRF* with three different levels of severity, *ORF* with three different levels of severity, and *REF* with three different levels of severity) of vibration signals under each load are collected with a sampling frequency of 12,000 Hz. In this paper, the same fault type under different load conditions is treated as one class. That is, the fault diagnosis of CWRU is a classification task of ten classes.

DIRG is another rolling bearing fault dataset collected by Politecnico di Torino [62]. It contains vibration signals in seven running conditions, i.e., *normal*, *inner ring fault with three different levels of severity*, and *rolling element fault with three different levels of severity*. These signals are collected with a sampling frequency of 51,200 Hz. Therefore, the fault diagnosis of DIRG is a classification task of seven classes.

CWRU and DIRG only contain the vibration signals under different running conditions, which need to be divided into a number of instances to form the training and test sets for conducting the experiments. For CWRU, the first 102,400 data points of vibration signals under each class are divided into 50 instances on average and there is no overlap between each instance [18]. The vibration signals of four different loads under each class are divided into instances (i.e., each class contains 200 (4×50) instances). That is, the CWRU dataset contains 2,000 (10×200) instances, where each instance is a vibration signal containing 2,048 data points. For DIRG, the first 51,200 data points of vibration signals under each class are divided into 50 instances on average and there is no overlap between each instance either. That is, the WHU dataset contains 350 (7×50) instances, where each instance is a vibration signal containing 1,024 data points.

UCOON [63] is a gear fault dataset collected by University of Connecticut (UCONN). It has nine types of gear vibration signals in different health states, i.e., *normal*, *missing tooth*, *root crack*, *spalling*, and *chipping tip with five different levels of severity*. These signals are collected by an accelerometer with a sampling frequency of 20,000 Hz. UCOON is a fault diagnosis task of nine classes. Unlike the CWRU dataset, UCOON has 104 instances (i.e., a vibration signal contains 3,600 data points) under each running condition. That is, the UCOON dataset contains 936 (9×104) instances.

WHU [64] is a rotor fault dataset collected by Wuhan University (WHU). This dataset has four types of rotor vibration signals in different health states, i.e., *normal*, *contact-rubbing*, *unbalance*, and *misalignment*. These signals are collected by an accelerometer with a sampling frequency of 2,048 Hz. The fault diagnosis of WHU is a classification task of four classes. Each class contains 45 instances (i.e., a vibration signal contains 2,048 data points). That is, the WHU dataset contains 180 (4×45) instances.

For the CWRU, DIRG and UCOON datasets, 50% instances of each class are randomly selected to form the training set and the remaining instances are used as the test set. For the WHU dataset, 23 instances of each class are randomly selected to form the training set and the remaining instances are used to form the test set. Table III lists the detailed information of

TABLE III
DATASET DESCRIPTION

Name	Instance length	Number of classes	Number of training instances	Number of test instances
CWRU	2,048	10	1,000 (10×100)	1,000 (10×100)
DIRG	1,024	7	175 (7×25)	175 (7×25)
UCOON	3,600	9	468 (9×52)	468 (9×52)
WHU	2,048	4	92 (4×23)	88 (4×22)

these four fault datasets, i.e., the number of fault types/classes, the length of the instance, the numbers of instances in the training and test sets.

B. Baseline Methods

To show the effectiveness of the AFECGP based fault diagnosis approach, 14 different methods are used for comparisons. These methods can be divided into three categories. The first category are five classical classification algorithms using Raw Signal Amplitude (RSA), i.e., RSA+KNN, RSA+SVM, RSA+LR, RSA+CART, and RSA+MLP. These classification algorithms take the amplitude values of the raw signals as inputs to train classifiers for classification. The purpose of comparisons is to investigate whether the features extracted and constructed by AFECGP from raw signals are more effective than RSA for fault diagnosis.

The second category includes five methods using different manually extracted/constructed features for fault classification. These features are Time-Domain Features (TDF) [17], Frequency-Domain Features (FDF) [17], Multi-Domain Features (MDF) [18], Modified Multi-scale Symbolic Dynamic Entropy (MMSDE) [23], and Improved Multi-scale Dispersion Entropy (IMDE) [27]. The feature numbers of TDF, FDF, MDF, MMSDE, and IMDE are 12, 4, 37, 20, and 20, respectively. These methods use KNN to perform classification, which is the same as AFECGP. The purpose is to investigate whether the features extracted and constructed by AFECGP can beat these manually extracted/constructed features on fault diagnosis.

The third category are four GP based feature construction methods (i.e., GP-SF, GP-MF, GP-OSF, and GP-OMF) for fault classification. In these methods, KNN is used for classification, which keeps the same as AFECGP. The GP-SF method constructs one high-level feature for classification. The GP-MF method constructs multiple high-level features for classification. The number of constructed features is set as the number of classes, according to [56]. The GP-OSF method constructs one high-level feature but uses the constructed feature and the original features together for classification. The GP-OMF method constructs multiple features (the feature number is the same as the class number [56]) but uses the constructed features and the original features together for classification. These four GP based methods use the commonly used functions for feature construction, i.e., +, -, \times , and protected \div (return 0 if the divisor is 0). In these methods, the terminal set include the TDF features and random constant. The reason for using TDF features as the terminal set is AFECGP only extracts features from the signal in time-domain. KNN is employed for fitness evaluation. For fair comparisons, these methods use the same

TABLE IV
FAULT DIAGNOSIS ACCURACY (%) OF AFECGP AND 14 COMPARISON METHODS ON FOUR FAULT DATASETS

Method	CWRU		DIRG		UCOON		WHU	
	Max	Avg±Std	Max	Avg±Std	Max	Avg±Std	Max	Avg±Std
RSA+KNN	15.00	15.00±0.00+	40.00	40.00±0.00+	80.98	80.98±0.00+	82.95	82.95±0.00+
RSA+LR	19.60	19.60±0.00+	27.43	27.43±0.00+	96.79	96.79±0.00+	35.23	35.23±0.00+
RSA+SVM	15.70	15.70±0.00+	28.00	28.00±0.00+	96.79	96.79±0.00+	34.09	34.09±0.00+
RSA+CART	23.20	21.91±0.62+	22.29	17.39±2.28+	67.95	64.99±1.06+	67.05	57.46±4.26+
RSA+MLP	32.20	30.00±0.78+	32.57	26.19±3.26+	96.79	96.19±0.31+	77.27	74.85±1.70+
TDF	93.80	93.80±0.00+	42.29	42.29±0.00+	91.03	91.03±0.00+	89.77	89.77±0.00+
FDf	85.30	85.30±0.00+	56.00	56.00±0.00+	92.52	92.52±0.00+	67.05	67.05±0.00+
MDF	97.20	97.20±0.00+	61.14	61.14±0.00+	100.0	100.0±0.00-	77.27	77.27±0.00+
MMSDE	87.50	87.50±0.00+	34.86	34.86±0.00+	94.02	94.02±0.00+	60.23	60.23±0.00+
IMDE	97.80	97.80±0.00+	53.71	53.71±0.00+	100.0	100.0±0.00-	97.73	97.73±0.00+
GP-SF	89.40	83.94±2.31+	49.71	43.09±3.05+	89.74	86.07±1.96+	89.77	83.94±3.09+
GP-MF	98.90	98.17±0.59+	61.14	55.81±3.16+	96.15	91.70±2.32+	98.86	96.40±1.86+
GP-OSF	98.40	98.02±0.25+	65.14	61.71±1.58+	82.26	81.37±1.33+	96.59	94.85±1.85+
GP-OMF	98.80	98.80±0.06+	64.00	60.17±2.13+	87.39	86.12±0.77+	96.59	94.62±0.88+
AFECGP	100.0	99.83±0.16	98.86	90.19±3.89	100.0	99.71±0.26	100.0	99.96±0.20
Overall	14+		14+		12+, 2-		14+	
Rank	1/15		1/15		3/15		1/15	

run time parameters as AFECGP. The purpose of comparisons is to investigate whether AFECGP can achieve better diagnosis performance than the other GP based baseline methods for feature construction.

C. Parameter Settings

The parameters of AFECGP are set according to the common settings in the GP community [40]. The initial population is generated using the ramped half-and-half method. The population size is set as 100 and the maximal number of generations is set as 50. Tournament selection with size 7 is used to select individuals for *Crossover* and *Mutation*. The rates of crossover, mutation, and elitism are 80%, 19% and 1%, respectively. The maximal and minimal tree depths are set as 8 and 2, respectively.

The GP based methods and classification algorithms are implemented using the DEAP (*Distributed Evolutionary Algorithm Package*) [65] and *scikit-learn* [66] packages, respectively. The number of neighbors in KNN is set as 3 according to [59], [67]. The parameters in the classification algorithms are the default values of the *scikit-learn* package. The parameter values of the methods using manually extracted features are set according to the corresponding references [17], [18], [23], [27], [67]. The experiments of all the methods on each dataset run 30 times independently to reduce the bias and randomness. The fault diagnosis results of the test sets are reported.

V. RESULTS AND DISCUSSIONS

A. Classification Results

In this section, the fault diagnosis results of the AFECGP method on the four fault diagnosis datasets are discussed and compared with the 14 baseline methods. The results of these methods are listed in Table IV. The results are the maximum (Max) accuracy, the average (Avg) accuracy, and the standard deviation (Std) of the 30 runs. The best classification results of each dataset are highlighted in bold. In addition, the significance of the difference in results between AFECGP and each

of the baseline methods is evaluated by the Wilcoxon rank-sum test with a 5% significance level. The symbols “+”, “-” or “=” in Table IV represent the performance of AFECGP is significantly better, worse than or similar to the corresponding baseline method, respectively. The fault diagnosis performance ranking of the AFECGP approach among all the methods on each dataset are listed in the final row of Table IV.

The first five rows of Table IV list the diagnosis results of five traditional classifiers using RSA. It can be found that AFECGP obtains significantly better performance in all the comparisons (4 datasets × 5 baseline methods). Since the features generated by AFECGP are fed into KNN for fault diagnosis, we should pay more attention to the results of KNN using RSA. The diagnosis accuracy difference is the largest on the CWRU dataset, where the maximal and average accuracy of KNN using RSA are 15%, which are 75% and 74.83% lower than that of AFECGP, respectively. The diagnosis accuracy difference is the smallest on the WHU dataset, where the maximal and average accuracy of KNN using RSA are 82.95%, which are 17.05% and 17.01% lower than that of AFECGP, respectively. These results show that the features extracted and constructed by AFECGP are more effective for fault diagnosis than the methods using original signal amplitudes.

Rows 6 to 10 of Table IV are the diagnosis results of the KNN classifier using manually extracted/designed features, i.e., TDF [17], FDF [17], MDF [18], MMSDE [23], and IMDE [27]. The results show that AFECGP achieves significantly better performance in 18 comparisons out of the total 20 comparisons (4 datasets × 5 baseline methods). For the CWRU, DIRH, and WHU datasets, the best maximal and average accuracies of AFECGP are better than that of these methods using manually extracted features. The diagnosis accuracy difference is the largest on the DIRG dataset, where the best maximal and average accuracies of these methods using manually extracted features are 61.14%, which are 37.72% and 29.05% lower than that of AFECGP, respectively. On the UCOON dataset, the maximal accuracy of using MDF, IMDE, and AFECGP are

100%, and the average accuracy of MDF and IMDE is 0.29% higher than that of AFECGP. Although AFECGP achieves worse performance than the methods using the MDF or IMDE features on UCOON, the best result achieved by AFECGP is the same as them, and AFECGP achieves significantly better performance than them on the remaining three datasets. These results show that the features generated by AFECGP perform better on classifying various types of vibration signals, while the manually extracted features may only be effective for classifying a certain type of signal. In summary, the features extracted and constructed by AFECGP are more discriminative and effective than the manually extracted/constructed features for these fault diagnosis problems.

Rows 11 to 14 of Table IV are the diagnosis results obtained the GP based feature construction baseline methods, i.e., GP-SF, GP-MF, GP-OSF, and GP-OMF. The results show that the AFECGP based diagnosis approach achieves significantly better performance in all the comparisons. The diagnosis accuracy difference is the largest on the DIRG dataset, where the best maximal and average accuracy of these traditional GP based methods are 65.14% and 60.17%, which are 33.72% and 30.02% lower than that of AFECGP, respectively. The fault diagnosis accuracy difference is the smallest on the CWRU dataset, where the best maximal and average accuracy of the GP based baseline methods are 98.9% and 98.8%, which are 1.1% and 1.03% lower than that of AFECGP, respectively. The diagnosis accuracy of GP-MF on all the datasets is better than that of GP-SF, which demonstrates that constructing multiple features is more effective for fault diagnosis than constructing a single feature. However, the performance of GP-OMF is only better than the other traditional GP based methods on the CWRU dataset, which demonstrates that more features do not necessarily improve classification accuracy. According to above results, the reasons of AFECGP achieves higher fault diagnosis accuracy than these four GP based methods can be summarized as follows: 1) AFECGP can automatically detect the useful subband signals, extract and construct effective features from the subband signals; 2) AFECGP can automatically determine the number of features for solving a fault diagnosis task. In summary, AFECGP can adaptively identify useful subband signals to extract and construct discriminative features, and automatically determine the optimal number of features to achieve effective fault diagnosis.

To sum up, AFECGP achieves significantly better results in 54 comparisons out of the total 56 comparisons (4 datasets \times 14 baseline methods). On the CWRU, WHU and UCOON dataset, AFECGP achieves the maximal diagnosis accuracy of 100% and the average diagnosis accuracy of above 99%. On the DIRG dataset, AFECGP achieves the maximal diagnosis accuracy of 98.86% and the average diagnosis accuracy of 90.19%. The diagnosis performance of AFECGP ranks the first among all these 15 methods on the CWRU, DIRG, and WHU datasets and ranks the third on the UCOON dataset. The results show that the AFECGP based diagnosis approach has excellent fault diagnosis ability and high adaptability. As a result, the AFECGP approach is an effective and promising approach to fault diagnosis of rotating machinery.

B. Convergence Behaviour

The GP methods are evolutionary learning algorithms. To show the learning ability of these GP based methods for feature extraction and construction, we analyse and compare the convergence curves of these five GP based methods, including the proposed AFECGP approach. Fig. 6 shows the evolution plots on the four datasets, where the horizontal axis of each sub-figure represents the number of generations and the vertical axis represents the average value of the best-of-generation diagnosis accuracy of the 30 runs. Each sub-figure has five colored lines representing GP-SF, GP-MF, GP-OSF, GP-OMF, and AFECGP, respectively. As shown in Fig. 6, the convergence performance of AFECGP is superior to that of GP-SF, GP-MF, GP-OSF, and GP-OMF on the four fault diagnosis datasets. On the CWRU, DIRG, UCOON, and WHU datasets, AFECGP takes 8, 25, 4, and 2 generations to obtain the optimal diagnosis accuracy, respectively, which is faster than the other four GP based methods. In addition, the optimal diagnosis accuracy of AFECGP is better than the other four GP based methods on the four fault diagnosis datasets.

To conclude, the comparisons show that AFECGP has a faster convergence speed and can find better solutions than the other GP based feature construction methods on these four fault diagnosis datasets. The main reasons are the designs of the new program structure, the function set and the terminal set in AFECGP. These new designs allow AFECGP to find better solutions to achieve higher diagnosis performance than the compared GP methods.

C. Computational Time

The computational time of AFECGP includes two parts, i.e., the evolutionary learning time and the fault diagnosis time. The evolutionary learning time is the time needed for conducting the evolutionary process (i.e., the time needed in the second step shown in Fig. 5). Since the non-GP based methods do not have the evolutionary process, the evolutionary learning time of AFECGP is only compared with the GP based methods. Table V lists the average learning time (in second) of the 30 runs of the GP based methods. From this table, it can be found that AFECGP uses about 1,007s, 195s, 607s, and 205s on the CWRU, DIRG, UCOON, and WHU datasets respectively to search for the best solutions. Compared with the other four GP methods, AFECGP uses a longer learning time to find the best solutions. Compared with these four GP methods, the solutions of AFECGP is more complex in order to achieve useful subband signal detection, feature extraction and construction. The sophisticated solutions of AFECGP need more computational time during the fitness evaluation process than the other four GP methods, which use simple trees to construct features from a set of manually extracted features. Although AFECGP needs more time than the other GP based methods, it is noted that the learning time of AFECGP is not long in general, i.e., 17 minutes at most, which is affordable.

The fault diagnosis time is the time for classifying the instances in the test set (i.e., the time needed in the third step of Fig. 5). Table VI lists the average fault diagnosis time of the 30 runs in seconds on the four datasets. It is clear that the methods

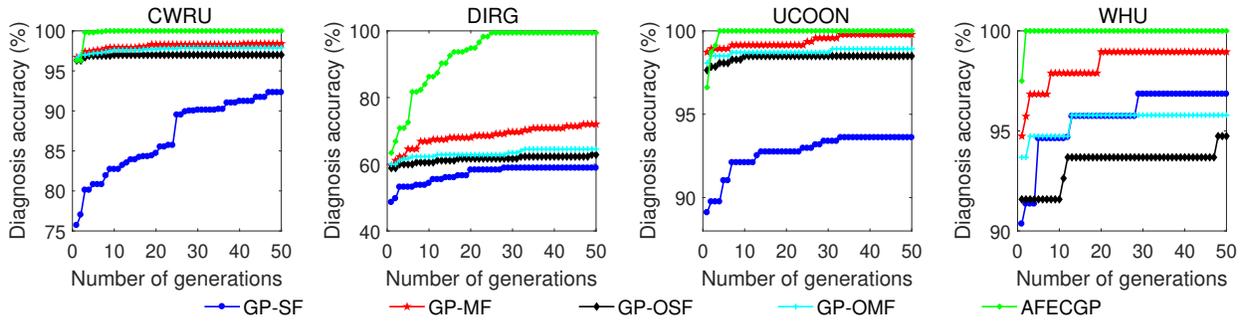


Fig. 6. Convergence curves of the GP based methods.

TABLE V
EVOLUTIONARY LEARNING TIME (SECONDS) OF GP-BASED METHODS
ON FOUR FAULT DIAGNOSIS DATASETS

Method	CWRU	DIRG	UCOON	WHU
	Avg±Std	Avg±Std	Avg±Std	Avg±Std
GP-SF	492.4±104.9	93.4±26.4	228.0±66.29	55.9±13.7
GP-MF	568.6±146.5	111.9±23.2	241.7±73.2	60.4±13.4
GP-OSF	566.3±107.7	105.3±22.5	283.9±55.7	52.6±12.2
GP-OMF	667.8±164.5	122.2±28.7	294.8±48.4	64.9±12.3
AFECGP	1007.7±358.9	195.4±58.4	607.3±301.4	205.4±69.8

TABLE VI
FAULT DIAGNOSIS TIME (SECONDS) ON THE FOUR DATASETS

Method	CWRU	DIRG	UCOON	WHU
	Avg±Std	Avg±Std	Avg±Std	Avg±Std
TDF	0.30±0.016	0.04±0.002	0.22±0.003	0.03±0.002
FDF	0.29±0.002	0.04±0.003	0.24±0.004	0.03±0.002
MDF	275.53±1.266	127.81±2.238	20.15±0.522	15.76±0.585
MMSDE	97.15±2.217	9.71±0.035	74.55±0.263	8.46±0.022
IMDE	176.12±0.417	15.72±0.037	130.19±1.805	13.43±0.085
GP-SF	0.61±0.187	0.08±0.024	0.31±0.076	0.19±0.012
GP-MF	0.60±0.166	0.07±0.019	0.28±0.031	0.04±0.009
GP-OSF	0.53±0.156	0.08±0.024	0.31±0.053	0.03±0.008
GP-OMF	0.64±0.333	0.08±0.033	0.32±0.043	0.04±0.008
AFECGP	0.84±0.329	0.09±0.027	0.34±0.061	0.05±0.022

using the MDF, MMSDE, and IMDE features use a longer time than the methods using the TDF and FDF features and the GP based methods. The reason is that the extraction of the MDF, MMSDE and IMDE features has a complicated signal processing process, while the extraction of the TDF and FDF features only performs statistical calculations on the signals. The AFECGP approach uses about 0.84, 0.09, 0.34, and 0.05 seconds on the CWRU, DIRG, UCOON, and WHU datasets, respectively. Compared with the other four GP based methods, the time of AFECGP is longer on these four datasets because the solutions of AFECGP have more calculations. However, the time of AFECGP is much less than that of the methods using the MDF, MMSDE and IMDE features. Overall, the time complexity of the AFECGP is less than most of the traditional fault diagnosis methods, and not remarkably larger than other GP based methods for feature construction.

D. Feature Visualization

To better illustrate the superiority of the features generated by AFECGP, the t-SNE dimensionality reduction technology [68] is used to reduce the original signal and the features generated by AFECGP into two dimensions (2-D) for visualization, respectively. The instances in the test set are

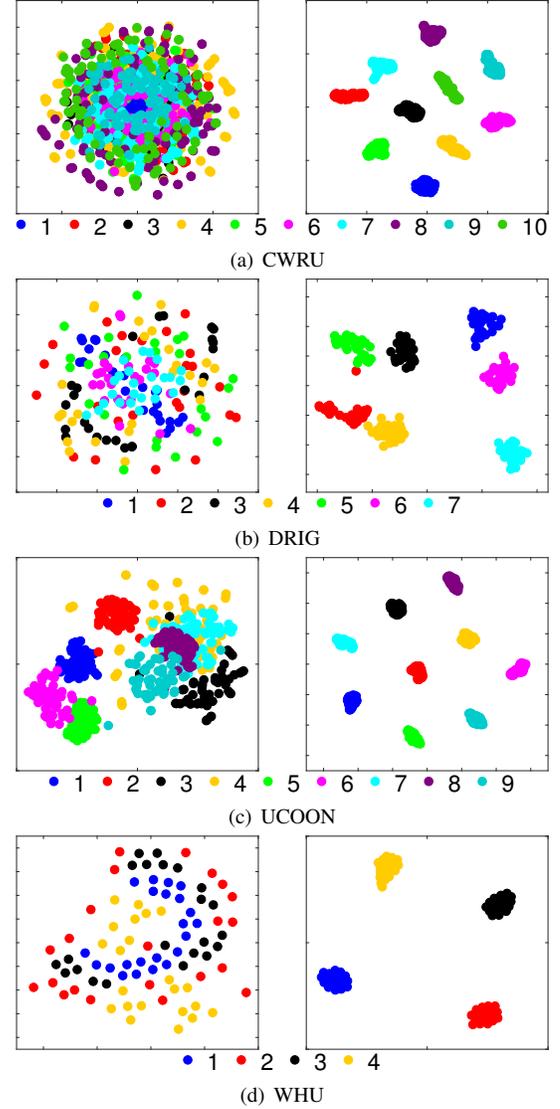


Fig. 7. Data visualization via t-SNE using the original signals and the features generated by AFECGP. Left: raw signals; right: features generated by AFECGP.

used for visualization and the results are shown in Fig. 7, where different running conditions are represented by different colors. For example, 10 colored points of Fig. 7(a) represent 10 running conditions of rolling bearing in the CWRU dataset. The left column of Fig. 7 shows the visualization results using the original signals, while the right column shows the visualization results using the features generated by AFECGP.

As it can be seen from the *left* figures of Fig. 7, for the

CWRU, DIRG, and WHU datasets, the points in different classes are randomly distributed and irregular. For the UCOON dataset, some same colored points are gathered together, but different colored clusters are overlapping. The visualization results of the original signals illustrate that the vibration signals under different running conditions are wrongly clustered, which indicates that the classification based on the raw signal may not be effective.

As it can be seen from the *right* figures of Fig. 7, for the CWRU, UCOON, and WHU datasets, the same colored points are gathered together, and there are obvious gaps between the different colored clusters. For the DIRG dataset, except for one red point is incorrectly gathered together with the green points, the other different colored points are gathered together and do not have overlap. The visualization results of the features generated by AFECGP illustrate that these features have good similarity among the same class and have big differences between different classes, which makes it easier to classify the vibration signals under different running conditions.

By the visualization of the features generated by AFECGP, the difference of the vibration signals under different running conditions can be seen intuitively. This illustrates that AFECGP can effectively extract and construct representative features in various vibration signals to improve the fault classification performance.

E. Example Model Analysis

Two example models/trees/programs evolved by AFECGP are used to further understand how informative features are extracted from original vibration signals and why they achieve effective fault diagnosis of rotating machinery.

1) *Model on CWRU*: An example model evolved/learned by AFECGP on the CWRU dataset is shown in Fig. 8, which achieves 100% accuracy on both the training and test sets. This model contains decomposition functions *HPF*, *LPF*, *BPF*, and *BEF*; rescaling functions: *Abs* and *Sqrt*; feature extraction functions *Std*, *Mean* and *Max*; feature construction functions $+$, $-$ and \div ; and feature combination functions *Fcom2* and *FcomM*.

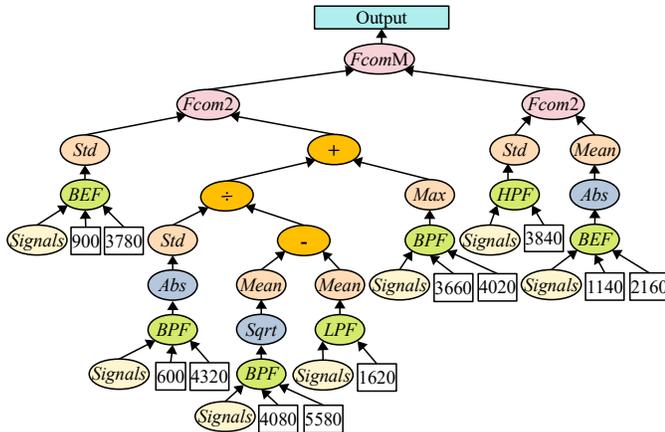


Fig. 8. Example solution evolved by AFECGP on the CWRU dataset.

From the example model on the CWRU dataset shown in Fig. 8, it can be found the information of seven subband

signals are extracted as features, which are further constructed to form an output feature vector of four high-level features. The example model in Fig. 8 has seven branches (from left to right): branch 1 performs *BEF* with a f_{eu} of 3,780 Hz and a f_{el} of 900 Hz on the original signal, and calculates the standard deviation of the filtered signal amplitudes as feature 1; branch 2 performs *BPF* with a f_{pu} of 4,320 Hz and a f_{pl} of 600 Hz on the original signal, uses *Abs* to transform the filtered signal amplitudes and calculates the standard deviation as feature 2; branch 3 performs *BPF* with a f_{pu} of 5,580 Hz and a f_{pl} of 4,080 Hz on the original signal, uses *Sqrt* to transform the filtered signal amplitudes and calculates the average value as feature 3; branch 4 performs *LPF* with a f_l of 1,620 Hz on the original signal, and calculates the average of the filtered signal amplitudes as feature 4; branch 5 performs *BPF* with a f_{pu} of 4,020 Hz and a f_{pl} of 3,660 Hz on the original signal, and calculates the maximum of the filtered signal amplitudes as feature 5; branch 6 performs *HPF* with a f_h of 3,840 Hz on the original signal, and calculates the standard deviation of the filtered signal amplitudes as feature 6; and branch 7 performs *BEF* with a f_{eu} of 2,160 Hz and a f_{el} of 1,140 Hz on the original signal, uses *Abs* to transform the filtered signal amplitudes and calculates the average value as feature 7. The features of branches 2, 3, 4 and 5 are used to construct a high-level feature using a mathematical expression, i.e., $feature\ 2 \div (feature\ 3 - feature\ 4) + feature\ 5$. The outputs of this example solution are the high-level feature constructed by branches 2, 3, 4 and 5, the feature constructed by branch 1, the feature constructed by branch 6, and the feature constructed by branch 7.

2) *Model on WHU*: An example model evolved by AFECGP on the WHU dataset is shown in Fig. 9, which achieves 100% accuracy on both the training and test sets. This model contains decomposition function *BPF* and *BEF*; rescaling functions *Abs* and *Sqrt*; feature extraction functions *Mean*, *Min*, and *Std*, and feature combination functions *Fcom2* and *FcomM*.

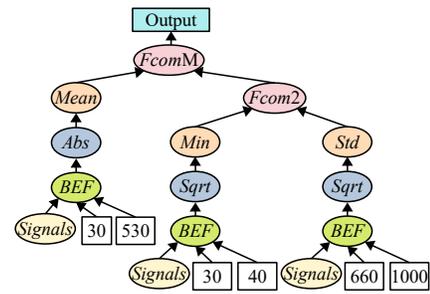


Fig. 9. Example solution evolved by AFECGP on the WHU dataset.

From the example model on the WHU dataset shown in Fig. 9, it can be found that the example model has a small size with three branches (from left to right): branch 1 performs *BEF* with a f_{eu} of 530 Hz and a f_{el} of 30 Hz on the original signal, uses *Abs* to transform the filtered signal amplitudes and calculates the average value as feature 1; branch 2 performs *BEF* with a f_{pu} of 40 Hz and a f_{pl} of 30 Hz on the original signal, uses *Sqrt* to transform the filtered signal amplitudes

and calculates the minimum as feature 2; branch 3 performs *BEF* with a f_{pu} of 1,000 Hz and a f_{pl} of 600 Hz on the original signal, uses *Sqrt* to transform the filtered signal amplitudes and calculates the standard deviation as feature 3. The outputs of this example solution are feature 1, feature 2 and feature 3 that are constructed by the three branches.

By the analysis of the two example models, the processes of feature extraction and construction of AFECGP are clearly displayed and interpreted. Unlike the existing fault diagnosis methods, AFECGP can adaptively determine the features number instead of presetting a fixed number. The example models on CWRU and WHU have different sizes and complexity, which illustrates that AFECGP can adaptively evolve solutions with variable lengths and complexity according to tasks.

F. Further Discussions

To achieve the fault diagnosis of rotating machinery effectively, it is typically better to generate as many features as possible for comprehensively describing the signals of the various health states [3]. AFECGP does not use the existing technical methods, such as multi-domain feature extraction and multi-scale feature extraction, to expand the feature pool. As it can be seen from Fig. 8 and Fig. 9 that these two example models learned/evolved by AFECGP are different from the existing feature extraction methods, which neither uses the whole signal nor uses one certain subband signal. In contrast, AFECGP adaptively determines multiple subband signals, extracts and constructs the features with the highest discriminability between different subband signals. Although these subband signals found by AFECGP may not be the resonance band signals concerned/employed by domain experts, they are very effective for fault diagnosis.

Typically, the manually extracted features are defined by domain experts using mathematical operators to combine various basic statistical indexes. For example, the pulse index [18] has the equation of $\max|x(n)|/(\frac{1}{N} \sum_{n=1}^N |x(n)|)$ using *Mean*, *Max*, *Abs* and \div operators for feature generation. AFECGP contains the basic statistical indexes (i.e., *Max*, *Min*, *Std*, *Mean*, and *KthM*) and mathematical operators (i.e., $+$, $-$, \times , \div and *Abs*), and has the capability to generate the same features as the features defined by domain experts. For example, the pulse index can be constructed by $\text{Max}(\text{Abs}(x)) \div \text{Mean}(\text{Abs}(x))$. However, from the example models evolved by AFECGP shown in Fig. 8 and Fig. 9, it can be found that the features generated by AFECGP are not same as the manually extracted features, but are more effective for classifying fault types.

In summary, rotating machinery fault diagnosis is an engineering problem traditionally requiring domain expertise, but it can also be solved using computational intelligence algorithms. The proposed AFECGP approach is based on evolutionary computation techniques, which can deal with the processes of information detection, feature extraction, feature construction and classification, automatically and simultaneously, without domain expertise and human intervention.

VI. CONCLUSIONS

The goal of this paper was to develop a new GP based approach to automatically and simultaneously extract and construct features from raw vibration signals for fault diagnosis of rotating machinery. This goal has been successfully achieved by developing the AFECGP approach having a new individual representation of a multi-layer structure, a new function set with various functions for different tasks, and a new terminal set with the raw signal and the corresponding parameters of functions. With these designs, the AFECGP approach can detect useful subband signals, extract features from the detected subband signals, construct high-level features from the extracted features, and combine these high-level features to form the output feature vector for fault diagnosis.

The performance of the AFECGP based diagnosis approach was examined on four fault datasets, i.e., two rolling bearing fault datasets, one gear fault dataset and one rotor dataset. The proposed approach was compared with fourteen baseline methods. The results showed that the features generated by AFECGP were more effective than raw signal, manually extracted features and the features constructed by the traditional GP based methods for fault diagnosis.

The investigations in this paper further reveal that when the rotating machinery running conditions become complex (e.g. more uncertain, dynamic and noisy), the traditional features or knowledge extracted/designed by human experts may not be effective for diagnosing the faults. In contrast, the automatically learned features by automated learning methods can adapt better and achieve more effective results. This further suggests that the use and development of automated learning algorithms for complex fault diagnosis is a good direction of research, and this paper opens the door for further investigation.

REFERENCES

- [1] R. Liu, B. Yang, E. Zio, and X. Chen, "Artificial intelligence for fault diagnosis of rotating machinery: A review," *Mech. Syst. Signal Process.*, vol. 108, pp. 33–47, 2018.
- [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, 2015.
- [3] T. W. Rauber, F. de Assis Boldt, and F. M. Varejão, "Heterogeneous feature models and feature selection applied to bearing fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 637–646, 2014.
- [4] X. Gu, S. Yang, Y. Liu, and R. Hao, "A novel pareto-based bayesian approach on extension of the infogram for extracting repetitive transients," *Mech. Syst. Signal Process.*, vol. 106, pp. 119–139, 2018.
- [5] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis," *Proc. Roy. Soc. Lond. A, Math., Phys. Eng. Sci.*, vol. 454, pp. 903–995, 1998.
- [6] M. G. Frei and I. Osorio, "Intrinsic time-scale decomposition: time-frequency-energy analysis and real-time filtering of non-stationary signals," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 463, no. 2078, pp. 321–342, 2007.
- [7] J. Zheng, J. Cheng, and Y. Yang, "A rolling bearing fault diagnosis approach based on lcd and fuzzy entropy," *Mech. Mach. Theory*, vol. 70, pp. 441–453, 2013.
- [8] Z. Wu and N. E. Huang, "Ensemble empirical mode decomposition: a noise-assisted data analysis method," *Adv. Adapt. Data Anal.*, vol. 1, no. 01, pp. 1–41, 2009.

- [9] Z. Feng, X. Lin, and M. J. Zuo, "Joint amplitude and frequency demodulation analysis based on intrinsic time-scale decomposition for planetary gearbox fault diagnosis," *Mech. Syst. Signal Process.*, vol. 72, pp. 223–240, 2016.
- [10] Z. Liu, Y. Jin, M. J. Zuo, and Z. Feng, "Time-frequency representation based on robust local mean decomposition for multicomponent am-fm signal analysis," *Mech. Syst. Signal Process.*, vol. 95, pp. 468–487, 2017.
- [11] K. Dragomiretskiy and D. Zosso, "Variational mode decomposition," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 531–544, 2013.
- [12] Y. Wang, R. Markert, J. Xiang, and W. Zheng, "Research on variational mode decomposition and its application in detecting rub-impact fault of the rotor system," *Mech. Syst. Signal Process.*, vol. 60, pp. 243–251, 2015.
- [13] G. K. Apostolidis and L. J. Hadjileontiadis, "Swarm decomposition: A novel signal analysis using swarm intelligence," *Signal Process.*, vol. 132, pp. 40–50, 2017.
- [14] Y. Miao, M. Zhao, V. Makis, and J. Lin, "Optimal swarm decomposition with whale optimization algorithm for weak feature extraction from multicomponent modulation signal," *Mech. Syst. Signal Process.*, vol. 122, pp. 673–691, 2019.
- [15] H. Li, Z. Li, and W. Mo, "A time varying filter approach for empirical mode decomposition," *Signal Process.*, vol. 138, pp. 146–158, 2017.
- [16] X. Zhang, Z. Liu, Q. Miao, and L. Wang, "An optimized time varying filtering based empirical mode decomposition method with grey wolf optimizer for machinery fault diagnosis," *J. Sound Vib.*, vol. 418, pp. 55–78, 2018.
- [17] X. Zhang, Q. Zhang, M. Chen, Y. Sun, X. Qin, and H. Li, "A two-stage feature selection and intelligent fault diagnosis method for rotating machinery using hybrid filter and wrapper method," *Neurocomputing*, vol. 275, pp. 2426–2439, 2018.
- [18] X. Yan and M. Jia, "A novel optimized svm classification algorithm with multi-domain feature and its application to fault diagnosis of rolling bearing," *Neurocomputing*, vol. 313, pp. 47–64, 2018.
- [19] J. S. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy," *Am. J. Physiol. Heart Circ. Physiol.*, vol. 278, no. 6, pp. H2039–H2049, 2000.
- [20] W. Chen, Z. Wang, H. Xie, and W. Yu, "Characterization of surface emg signal based on fuzzy entropy," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 2, pp. 266–272, 2007.
- [21] C. Bandt and B. Pompe, "Permutation entropy: a natural complexity measure for time series," *Phys. Rev. Lett.*, vol. 88, no. 17, p. 174102, 2002.
- [22] M. Rostaghi and H. Azami, "Dispersion entropy: A measure for time-series analysis," *IEEE Signal Process. Lett.*, vol. 23, no. 5, pp. 610–614, 2016.
- [23] Y. Li, Y. Yang, G. Li, M. Xu, and W. Huang, "A fault diagnosis scheme for planetary gearboxes using modified multi-scale symbolic dynamic entropy and mrmr feature selection," *Mech. Syst. Signal Process.*, vol. 91, pp. 295–312, 2017.
- [24] M. Costa, A. L. Goldberger, and C.-K. Peng, "Multiscale entropy analysis of biological signals," *Phys. Rev. E*, vol. 71, no. 2, p. 021906, 2005.
- [25] H. Azami, S. E. Arnold, S. Saneii, Z. Chang, G. Sapiro, J. Escudero, and A. S. Gupta, "Multiscale fluctuation-based dispersion entropy and its applications to neurological diseases," *IEEE Access*, vol. 7, pp. 68 718–68 733, 2019.
- [26] J. Zheng, H. Pan, and J. Cheng, "Rolling bearing fault detection and diagnosis based on composite multiscale fuzzy entropy and ensemble support vector machines," *Mech. Syst. Signal Process.*, vol. 85, pp. 746–759, 2017.
- [27] X. Yan and M. Jia, "Intelligent fault diagnosis of rotating machinery using improved multiscale dispersion entropy and mrmr feature selection," *Knowl. Based Syst.*, vol. 163, pp. 450–471, 2019.
- [28] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, 2015.
- [29] Y. Li, Y. Wei, K. Feng, X. Wang, and Z. Liu, "Fault diagnosis of rolling bearing under speed fluctuation condition based on vold-kalman filter and rcmfe," *IEEE Access*, vol. 6, pp. 37 349–37 360, 2018.
- [30] D. Wang, "K-nearest neighbors based methods for identification of different gear crack levels under different motor speeds and loads: Revisited," *Mech. Syst. Signal Process.*, vol. 70, pp. 201–208, 2016.
- [31] U. Jung and B.-H. Koh, "Wavelet energy-based visualization and classification of high-dimensional signal for bearing fault detection," *Knowl. Inf. Syst.*, vol. 44, no. 1, pp. 197–215, 2015.
- [32] C. Zhang, J. H. Sun, and K. C. Tan, "Deep belief networks ensemble with multi-objective optimization for failure diagnosis," in *Proc. IEEE Int. Conf. Syst. Man, Cybern.*, 2015, pp. 32–37.
- [33] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, "Real-time motor fault detection by 1-d convolutional neural networks," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7067–7075, 2016.
- [34] G. Jiang, H. He, J. Yan, and P. Xie, "Multiscale convolutional neural networks for fault diagnosis of wind turbine gearbox," *IEEE Trans. Ind. Electron.*, vol. 66, no. 4, pp. 3196–3207, 2018.
- [35] J. R. Koza, *Genetic Programming: on the Programming of Computers by Means of Natural Selection*. MIT press, 1992.
- [36] Y. Bi, B. Xue, and M. Zhang, "A survey on genetic programming to image analysis," *J. Zhengzhou Univ. Eng. Sci.*, vol. 39, no. 06, pp. 3–13, 2018.
- [37] Q. Chen, M. Zhang, and B. Xue, "Feature selection to improve generalization of genetic programming for high-dimensional symbolic regression," *IEEE Trans. Evol. Comput.*, vol. 21, no. 5, pp. 792–806, 2017.
- [38] Y. Mei, S. Nguyen, B. Xue, and M. Zhang, "An efficient feature selection algorithm for evolving job shop scheduling rules with genetic programming," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 1, no. 5, pp. 339–353, 2017.
- [39] H. Al-Sahaf, Y. Bi, Q. Chen, A. Lensen, Y. Mei, Y. Sun, B. Tran, B. Xue, and M. Zhang, "A survey on evolutionary machine learning," *J. Roy. Soc. N. Z.*, vol. 49, no. 2, pp. 205–228, 2019.
- [40] Y. Bi, B. Xue, and M. Zhang, "Genetic programming with a new representation to automatically learn features and evolve ensembles for image classification," *IEEE Trans. Cybern.*, 2020. DOI: 10.1109/TCYB.2020.2964566.
- [41] K. Nag and N. R. Pal, "A multiobjective genetic programming-based ensemble for simultaneous feature selection and classification," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 499–510, 2015.
- [42] W. Fu, M. Johnston, and M. Zhang, "Low-level feature extraction for edge detection using genetic programming," *IEEE Trans. Cybern.*, vol. 44, no. 8, pp. 1459–1472, 2013.
- [43] Y. Bi, B. Xue, and M. Zhang, "An effective feature learning approach using genetic programming with image descriptors for image classification [research frontier]," *IEEE Comput. Intell. Mag.*, vol. 15, no. 2, pp. 65–77, 2020.
- [44] H. Guo, L. B. Jack, and A. K. Nandi, "Feature generation using genetic programming with application to fault classification," *IEEE Trans. Syst., Man, Cybern. C, Cybern.*, vol. 35, no. 1, pp. 89–99, 2005.
- [45] J. Xuan, H. Jiang, T. Shi, and G. Liao, "Gear fault classification using genetic programming and support vector machines," *Int. J. Inf. Technol.*, vol. 11, no. 9, p. 37, 2005.
- [46] D. J. Montana, "Strongly typed genetic programming," *Evol. Comput.*, vol. 3, no. 2, pp. 199–230, 1995.
- [47] Y. Bi, B. Xue, and M. Zhang, "Genetic programming with image-related operators and a flexible program structure for feature learning to image classification," *IEEE Trans. Evol. Comput.*, 2020. DOI:10.1109/TEVC.2020.3002229.
- [48] D. Atkins, K. Neshatian, and M. Zhang, "A domain independent genetic programming approach to automatic feature extraction for image classification," in *Proc. IEEE Congress of Evolutionary Computation (CEC)*, 2011, pp. 238–245.
- [49] H. Al-Sahaf, A. Song, K. Neshatian, and M. Zhang, "Two-tier genetic programming: Towards raw pixel-based image classification," *Expert Syst. Appl.*, vol. 39, no. 16, pp. 12 291–12 301, 2012.
- [50] Y. Bi, B. Xue, and M. Zhang, "An automatic feature extraction approach to image classification using genetic programming," in *Proc. Int. Conf. Appl. Eov. Comput.*, 2018, pp. 421–438.
- [51] S. Ma, M. Chen, J. Wu, Y. Wang, B. Jia, and Y. Jiang, "High-voltage circuit breaker fault diagnosis using a hybrid feature transformation approach based on random forest and stacked autoencoder," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9777–9788, 2019.
- [52] A. Lensen, B. Xue, and M. Zhang, "Genetic programming for evolving a front of interpretable models for data visualization," *IEEE Trans. Cybern.*, 2020.
- [53] M. Muharram and G. D. Smith, "Evolutionary constructive induction," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 11, pp. 1518–1528, 2005.
- [54] K. Neshatian, M. Zhang, and P. Andraea, "A filter approach to multiple feature construction for symbolic learning classifiers using genetic programming," *IEEE Trans. Evol. Comput.*, vol. 16, no. 5, pp. 645–661, 2012.
- [55] J. Ma and G. Teng, "A hybrid multiple feature construction approach for classification using genetic programming," *Appl. Soft Comput.*, vol. 80, pp. 687–699, 2019.

- [56] B. Tran, B. Xue, and M. Zhang, "Genetic programming for multiple-feature construction on high-dimensional classification," *Pattern Recognit.*, vol. 93, pp. 404–417, 2019.
- [57] H. S. Jeong and E. A. Elsayed, "On-line surveillance and monitoring," in *Maintenance, Modeling and Optimization*. Springer, 2000, pp. 309–343.
- [58] T. Narendiranath Babu, T. Manvel Raj, and T. Lakshmanan, "Application of butterworth filter for fault diagnosis on journal bearing," *J. Vibroeng.*, vol. 16, no. 3, pp. 1602–1617, 2014.
- [59] J. Tian, C. Morillo, M. H. Azarian, and M. Pecht, "Motor bearing fault detection using spectral kurtosis-based feature extraction coupled with k-nearest neighbor distance analysis," *IEEE Trans. Ind. Electron.*, vol. 63, no. 3, pp. 1793–1803, 2016.
- [60] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: A review and roadmap," *Mech. Syst. Signal Process.*, vol. 138, p. 106587, 2020.
- [61] "Case western reserve university bearing data center. [online]," <http://csegroups.case.edu/bearingdatacenter/home/>, accessed April, 2018.
- [62] A. P. Daga, A. Fasana, S. Marchesiello, and L. Garibaldi, "The politecnico di torino rolling bearing test rig: Description and analysis of open access data," *Mech. Syst. Signal Process.*, vol. 120, pp. 252–273, 2019.
- [63] P. Cao, S. Zhang, and J. Tang, "Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning," *IEEE Access*, vol. 6, pp. 26 241–26 253, 2018.
- [64] D. Liu, Z. Xiao, X. Hu, C. Zhang, and O. Malik, "Feature extraction of rotor fault based on eemd and curve code," *Measurement*, vol. 135, pp. 712–724, 2019.
- [65] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *J. Mach. Learn. Res.*, vol. 13, pp. 2171–2175, 2012.
- [66] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
- [67] Y. Bi, B. Xue, and M. Zhang, "Evolving deep forest with automatic feature extraction for image classification using genetic programming," in *Proc. PPSN*. Springer, 2020, pp. 3–18.
- [68] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *J. Mach. Learn. Res.*, vol. 9, no. Nov, pp. 2579–2605, 2008.



Bo Peng received the B.E. degree from Agricultural University of Hebei, Hebei, China, in 2014, and the M.E. degree from North China Electric Power University, Hebei, China, in 2017.

He is currently pursuing the Ph.D. degree in power machinery and engineering with the School of Energy, Power and Mechanical Engineering, North China Electric Power University, Beijing, China. He is currently visiting the Evolutionary Computation Research Group, School of Engineering and Computer Science, Victoria University of Wellington,

Wellington, New Zealand.

His current research interests include fault diagnosis, evolutionary computation, and machine learning.



Shuting Wan received the B.E., M.E., and Ph.D. degrees from North China Electric Power University, Hebei, China, in 1991, 1994, and 2006, respectively.

He is currently a Professor with the department of Mechanical Engineering, and the Dean with the Hebei Key Laboratory of Electric Machinery Health Maintenance & Failure Prevention, North China Electric Power University. His research focuses mainly on fault diagnosis and condition monitoring on rotation machinery, signal analysis and processing, and structural dynamics of electric equipment.

He is the Vice-Chair of Hebei Provincial Society for Vibration Engineering and the Committee Member of Chinese Society for Vibration Engineering.



Ying Bi (M'17) received the B.Sc. degree in 2013 from Wuhan Polytechnic University, Hubei, China, the M.Sc. degree in 2016 from Shenzhen University, Shenzhen, China, and the PhD degree in 2020 from Victoria University of Wellington, New Zealand.

She is currently a Post-Doctoral Research Fellow with the School of Engineering and Computer Science, Victoria University of Wellington. Her current research interests include evolutionary computation, computer vision, and machine learning. She has published over 30 papers in this field, including top

journal and conference papers.

She is a member of the IEEE Computational Intelligence Society and has been severing as reviewers for top international journals and conferences, such as IEEE Transactions on Evolutionary Computation, IEEE Transactions on Cybernetics, IEEE Congress on Evolutionary Computation (CEC), and the Genetic and Evolutionary Computation Conference (GECCO).



Bing Xue (M'10) received the B.Sc. degree from the Henan University of Economics and Law, Zhengzhou, China, in 2007, the M.Sc. degree in management from Shenzhen University, Shenzhen, China, in 2010, and the PhD degree in computer science in 2014 at Victoria University of Wellington (VUW), New Zealand.

She is currently an Associate Professor and Program Director of Science in School of Engineering and Computer Science at VUW. She has over 200 papers published in fully refereed international journals and conferences and her research focuses mainly on evolutionary computation, machine learning, classification, symbolic regression, feature selection, evolving deep neural networks, image analysis, transfer learning, multi-objective machine learning.

Dr Xue is currently the Chair of IEEE Computational Intelligence Society (CIS) Data Mining and Big Data Analytics Technical Committee, and Vice-Chair of IEEE Task Force on Evolutionary Feature Selection and Construction, Vice-Chair of IEEE CIS Task Force on Transfer Learning & Transfer Optimization, and of IEEE CIS Task Force on Evolutionary Deep Learning and Applications. She is also served as associate editor of several international journals, such as IEEE Computational Intelligence Magazine and IEEE Transactions of Evolutionary Computation.



Mengjie Zhang (M'04-SM'10-F'19) received the B.E. and M.E. degrees from Artificial Intelligence Research Center, Agricultural University of Hebei, Hebei, China, and the Ph.D. degree in computer science from RMIT University, Melbourne, VIC, Australia, in 1989, 1992, and 2000, respectively.

He is currently Professor of Computer Science, Head of the Evolutionary Computation Research Group, and the Associate Dean (Research and Innovation) in the Faculty of Engineering. His current research interests include evolutionary computation, particularly genetic programming, particle swarm optimization, and learning classifier systems with application areas of image analysis, multi-objective optimization, feature selection and reduction, job shop scheduling, and transfer learning.

He has published over 500 research papers in refereed international journals and conferences. Prof. Zhang is a Fellow of Royal Society of New Zealand and have been a Panel member of the Marsden Fund (New Zealand Government Funding), a Fellow of IEEE, and a member of ACM. He was the chair of the IEEE CIS Intelligent Systems and Applications Technical Committee, and chair for the IEEE CIS Emergent Technologies Technical Committee and the Evolutionary Computation Technical Committee, and a member of the IEEE CIS Award Committee. He is a vice-chair of the IEEE CIS Task Force on Evolutionary Feature Selection and Construction, a vice-chair of the Task Force on Evolutionary Computer Vision and Image Processing, and the founding chair of the IEEE Computational Intelligence Chapter in New Zealand. He is also a committee member of the IEEE NZ Central Section.