Bearing Fault Classification Using Improved Antlion Optimizer and Extreme Learning Machine

Zhuanzhe Zhao,1,2 Yu Zhang,1 Qiang Ma,1 Yujian Rui,1 Guowen Ye,1 Mengxian Wang,1 Yongming Liu,1,3 Zhen Zhang,1,2 Neng Wei,2 and Zhijian Tu2

1 School of Mechanical Engineering, Anhui Polytechnic University, Wuhu, Anhui 241000, China
2 Wuhu Ceprei Robotics Industry Technology Research Institute Co. Ltd., Wuhu, Anhui 241003, China
3 Anhui New R&D Institutions of Human-machine Interaction and Collaboration, Wuhu, Anhui 241000, China

Correspondence should be addressed to Yongming Liu; liuyongming1015@163.com

Received 4 June 2021; Revised 12 April 2022; Accepted 21 April 2022; Published 18 May 2022

1. Introduction

Bearing is one of the most widespread and important parts in mechanical equipment. The classification of early faults of bearings is a very important issue in practical applications, which has attracted the attention of many scholars and engineers. Due to the complex structure of most mechanical equipment, there is a nonlinear coupling relationship between components. And the weak non-stationary bearing fault signal is often submersed in the environmental noise. All these have caused some difficulties for the accurate classification of early faults of bearings. At present, there are three main types of fault diagnosis and classification methods for bearing: model-based methods, knowledge-based reasoning methods, and data-driven methods. The model-based method explored the operating laws of bearings by constructing accurate mathematical models and obtained the related information of normal and abnormal operation by studying the internal relationship between dynamic parameters and response signs under fault conditions. In the literature [1], a dynamic model considering the time-varying deflection excitation (TDE) and the time-
varying contact stiffness excitation (TCSE) is introduced to investigate vibration responses of a lubricated rolling bearing with a uniform and a non-uniform surface waviness on its races. To investigate the vibration mechanism of cylindrical roller bearings with localized defects, Cao et al. [2] developed the dynamic models with gyroscopic moment, centrifugal force, and lubrication traction/slip between roller/raceway, roller/cage, and cage/raceway. However, with the increasing complexity of mechanical systems and more coupling factors, the accurate mathematical models are becoming more and more difficult to establish. The method based on knowledge reasoning does not require the accurate mathematical model of the system, but carries out computational reasoning and classification, such as fault tree-based reasoning [3], DS theory [4], and expert system [5]. These methods usually require a large amount of domain expert knowledge, knowledge expression, and reasoning mechanism. Therefore, it is difficult for its large-scale application in case of incomplete or lack of domain knowledge. The data-driven diagnosis method, such as machine learning, does not need to understand the physical structure and the precise mathematical model, nor does it require a large amount of accumulation of expert knowledge. It only uses measurable condition monitoring signals or historical data to analyze and extract characteristic information, so as to realize the fault diagnosis, classification, and performance evaluation of bearings. A condition monitoring index of bearings was developed based on self-organizing map (SOM) and improved hidden Markov model so as to detect incipient bearing faults quickly [6]. In the literature [7], a three-stage semisupervised learning approach using data augmentation and metric learning is proposed for an intelligent bearing fault diagnosis under limited labeled data. Wen et al. [8] proposed a novel method that integrates graph theory and singular value decomposition (SVD) with the purpose of inspection of dynamic rolling element bearings health conditions. The data-driven method has attracted the great attention of researchers, and its application in the intelligent fault diagnosis has become more and more extensive.

In recent years, various pattern recognition models emerge in an endless stream, such as the support vector machine (SVM) [9], variable prediction model [10], neural network [11], and deep learning [12, 13]. Ravikumar et al. [14] proposed a fault diagnosis model that includes a multi-scale deep residual learning with a stacked long short-term memory (MDRL-SLSTM) to cope with sequence data in a gearbox health prediction process in an internal combustion engine. A novel learning rate scheduler based on the reinforcement learning (RL) for convolutional neural network (RL-CNN) [15] is presented and applied in fault classification of three bearing datasets. As a single hidden layer feedforward neural network, extreme learning machine (ELM) has also been widely used by researchers. But it also has some disadvantages. For example, the input weights and thresholds of the ELM network are randomly generated, and some almost worthless nodes will inevitably be generated, which will cause the solution accuracy of ELM model to decrease seriously, or even fall into local convergence [16]. Some optimization algorithms have been tried in the literature to improve the performance of ELM networks. For example, Zeng et al. [17] combined ELM with an improved particle swarm optimization (PSO) algorithm that added a delay factor to short-term load forecasting of power systems, to avoid the addition of unnecessary hidden layer nodes and overtraining, and to improve the generalization of ELM networks. Jiang et al. [18] proposed an ELM model combined with a compound differential evolution algorithm and applied it to the prediction of the silicon content of molten iron in a blast furnace. Kadri et al. [19] applied an ant colony (ACO) algorithm with ELM in the fault detection and identification of cement rotary kilns. Wang et al. [20] introduced a fault diagnosis scheme based on multi-scale diversity entropy (MDE) and ELM, which was verified by simulated signals and two experimental data from the bearing and the dual-rotator of the aeroengine test in gears.

Swarm optimization algorithm, such as the PSO and ACO, is a kind of algorithm inspired by the intelligent cooperation of biological groups. It has the advantages of easy implementation, good parallelism, and strong adaptability. Recently, new meta-heuristics optimization algorithms based on bionics have sprung up one after another, such as slap swarm algorithm (SSA) [21], the monarch butterfly optimization (MBO) [22], slime mould algorithm (SMA) [23], moth search algorithm (MSA) [24], hunger games search (HGS) [25], colony predation algorithm (CPA) [26], marine predator algorithm (MPA) [27], Harris hawks optimization (HHO) [28], mayfly algorithm (MA) [29], reptile search algorithm (RSA) [30], and honey badger algorithm (HBA) [31]. As a bionic intelligence algorithm, the antlion optimizer (ALO) simulates the predation mechanism of the antlions in the natural environment [32]. Due to few parameters and high convergence accuracy, it has been used in motor control [33, 34], path planning [35], multi-objective optimization [36], scheduling optimization [37, 38], and other fields [39, 40]. According to the no free lunch theorem [41], there is no optimization algorithm that can be universally used in all engineering fields, and the performance of all optimization algorithms on all possible function sets is equivalent. When ALO is used to solve actual complex engineering problems, there are several shortcomings, such as easy precocity of populations. Therefore, some variants of ALO have been proposed in different applications. For example, Yue and Zhang [42] employed a new inertial weight strategy and integrated invasive weed optimization algorithm (IWOA) and bat algorithm (BA) to further improve the performance of the original ALO. Kilic et al. [43] proposed an improved random walk model for the long time-consuming process of ALO, which uses the tournament selection method instead of roulette to improve the global search efficiency of ALO. An improved ALO based on spiral complex path searching pattern is proposed to improve the diversity of the population and the ability of the algorithm to balance exploration and exploitation [44]. Tharwat and Hassani [45] proposed a new ALO algorithm based on chaos theory and applied it to optimize the initial parameters of SVM model. The classical Lévy flight is also used to improve the ALO [46–48]. The Lévy flight random
walk was suggested as a permutation for performing a local search in Ref. [46], and it is utilized to replace the random walk as the motion of ants in Ref. [47]. But in Ref. [48], it is essentially the fusion of the group search optimization (GSO) algorithm and ALO; that is, the better solutions are selected from the initial random solutions, and they are updated using GSO, whereas the remaining initial solutions are updated using ALO. In the above 3 literatures, the step size control factor $w$ of Lévy distribution is constant and is simply used. The deep fusion with the Lévy strategy is not carried out according to the actual situation.

Therefore, in order to accurately classify the early faults of rolling bearings, a new pattern recognition model based on the improved ALO algorithm and ELM network is proposed in this paper. Firstly, the estimation of distribution algorithm (EDA) and a variable-step Lévy flight strategy are introduced into the initial ALO, called ELALO. To avoid the problem of local optimum and instability during ELM training process, the new ELALO is used to optimize the initial parameters of ELM network and the new pattern recognition is formed, denoted as ELALO-ELM. The experimental results show that the recognition rate of ELALO-ELM model is improved significantly through the case of rolling bearing early fault classification.

The remainder of this paper is organized as follows. The proposed ELALO algorithm is dictated in Section 2, where the superiority of ELALO over conventional ALO is demonstrated by a comparative study. Section 3 introduces the ELALO-ELM network model and its structure, and then applies the hybrid model to the early fault classification of rolling bearings in the area. Section 4 introduces the experimental comparative study using ELM and ELALO-ELM network models. Section 5 summarizes this article.

### 2. Improved Antlion Optimizer

#### 2.1. Antlion Optimizer

ALO is a meta-heuristic intelligent algorithm that mimics the hunting mechanism of ant lions in nature [32]. Five hunting steps are extracted on the predation process, that is, the random walk of ants, trap mining of ant lion, ants falling into the trap, ant lion preying on ants, and trap reconstruction. According to the above steps, a mathematical model of antlion hunting and ALO algorithm is established. The interactions between ants and ant lions in the above steps are all reflected in vectors in multi-dimensional space. In order to expound the meaning of the formula, all the following models are constructed based on dimension $D = 1$.

1. The ants walk randomly in the solution space, and the walk step formula can be described as follows:

   \[
   rw(t + 1) = \sum_{t=1}^{T} rw(t). 
   \]

   In Equations (1) and (2), $rw(1)$ represents the initial step value of the first generation of ants to walk. $T$ represents the maximum number of iterations, and $t$ is the number of current iterations. $rw(t)$ is the walking step length of the ant from the starting point in $(t+1)^{th}$ and the step length value is equal to the cumulative sum of all previous generation steps. The $rand$ takes a random number between 0 and 1.

2. Antlion digs traps with itself as the center. The process maps to a multi-dimensional space equivalent to constructing a super-sphere trap model near the antlion. The two-dimensional form of the model is shown in Figure 1. Equations (3) and (4) can be used to transform the initial boundary range of the algorithm into a trap enclosure centered on the antlion as follows:

   \[
   c_r(t) = a_l(t) + c(t), \quad \text{(3)}
   \]

   \[
   d_r(t) = a_l(t) + d(t). \quad \text{(4)}
   \]

   In Equations (3) and (4), $c(t)$ and $d(t)$, respectively, represent the upper and lower bounds of the generation $t$ trap. $a_l(t)$ represents the antlion individual, which is selected as $X_{rs}$ by the roulette strategy or $X_e$ by the elite strategy in the generation $t$ trap. $c_r(t)$ and $d_r(t)$, respectively, represent the upper and lower bounds of the coordinate of the generation $t$ trap relative to the coordinates of the antlion $a_l(t)$.

3. To prevent ants from randomly walking beyond the boundary, the standardization of the walking range is carried out by the following:

   \[
   R_{s.d}(t) = \frac{(rw(t) - a(T)) \cdot (b(T) - c_r(t))}{d_r(t) - a(t)} + c_r(t), \quad \text{(5)}
   \]

   where $R_{s.d}(t)$ represents the standardized random walking step length of the $t^{th}$ generation. $a(T)$ and $b(T)$, respectively, represent the maximum and minimum of the walking step length of the ant after reaching $T$ times of walking around the antlion.

4. The random walk of each ant in the range of $X_{rs}$ and $X_e$ can be expressed as:

   \[
   \text{ant}(t + 1) = \frac{R_{rs}(t) + R_{cr}(t)}{2}. \quad \text{(6)}
   \]
Whether the error accuracy is reached?

![Flow chart of EDA](image)

Figure 2: Flow chart of EDA.

where ant \((t + 1)\) represents the updated ant individual of the generation \(t + 1\). When the center antlion, which is surrounded by ants, is determined by the \(X_{\text{c}}, R_{\text{c}}(t)\) can be recorded as \(R_{\text{c}}(t)\). Similarly, when the center antlion, which is surrounded by ants, is the elite antlion \(X_{\epsilon}, R_{\epsilon}(t)\) can be recorded as \(R(t)\).

(5) The antlion pulls the ant inside sand and consumes its body. This process occurs under the condition that the fitness value of the ant is better than that of the antlion, and the antlion will replace the ant’s position. If the minimum fitness value is optimal, the process can be described as

\[
al(t + 1) = \begin{cases} 
ant(t + 1) & \text{if } f(\text{ant}(t + 1)) < f(al(t)), \\
al(t) & \text{otherwise}, \\
\end{cases}
\]

(7)

(6) When the grain reserves are exhausted, the antlion will repair the traps to wait for the next hunting. For this behavior, ALO algorithm is assumed that the scope of the trap will be adaptively reduced with the iterative process to improve the ability of local optimization. Mathematical models are shown in Equations (8) and (9):

\[
c(t) = \frac{c(t - 1)}{I(t)},
\]

(8)

\[
d(t) = \frac{d(t - 1)}{I(t)},
\]

(9)

where \(I(t) = 10\omega T\), and \(\omega\) is given different constant values in different iterative stages of the algorithm.

2.2. Improved Antlion Optimizer

2.2.1. Introduction of an Estimation of Distribution Algorithm. The EDA [49] is an algorithm like genetic algorithm that uses the strategy of establishing a probability model to replace the population optimization change of the genetic operator. According to the different learning and sampling forms of probabilistic models, EDA has developed many different forms of algorithms, which are widely used in multi-objective optimization [50], path planning [51], machine learning [52], hierarchical cluster analysis [53, 54], and other fields.

According to the differences of the variable forms constructed by the probability model, EDA can be divided into the discrete type and the continuous type. This article mainly deals with the usage of the continuous EDA. The algorithm flow chart is shown in Figure 2.

Based on the probability model calculated by each generation, EDA generates new individuals based on the probability model calculated by each generation, which breaks through the range limitation of ALO random walk and makes it easier to achieve the global optimum. Since each generation of EDA is generated based on a probabilistic model, it is not only effective in solving nonlinear coupling problems, but also can directly describe the evolutionary trend of the population from a macro level. On this basis, the EDA probability model is introduced into the initial ALO model to realize the idea of population generation. The ALO algorithm in the population generation stage includes the initial population stage and the stage where the antlion and ant populations are mixed to form the next generation population at the end of each generation. In these newly generated populations, individuals with a certain fitness value are selected (the selection rate in this paper is 50%). And the average and variance of each individual dimension are calculated and used to build a Gaussian probability model. Individuals randomly generated by the probability model replace some of the individuals in the population that are lagging in fitness. Given that \(z\) dominant individuals are selected to construct the model, the set of positions of the \(i^{th}\) dimension \(z\) individual positions is \(U^i = (u_1^i, u_2^i, \ldots, u_z^i)\).

\[
A^i = \text{gauss}(\bar{u}^i, \epsilon^i),
\]

(10)

where \(\text{gauss()}\) indicates that a new population \(A^i\) that meets the requirements of mean \(\bar{u}^i\) and variance \(\epsilon^i\) is generated within the scope of the \(i^{th}\) dimensional solution space.
2.2.2. Introduction of Variable Step Size Lévy Flight Strategy.

The Lévy flight is an optimization strategy obeying the Lévy probability distribution [55]. The principle of Brownian motion in physics can correspond to this search strategy well. In the cuckoo algorithm [56], the bird renewal formula is based on this model. Lévy flight can quickly change the orientation in the solution space, so when applied to the optimization field, it can guide the algorithm to quickly jump out of the local optimum and improve the performance of the algorithm. The position updating formula of Lévy flight is as follows [57]:

\[
\begin{align*}
    p_{t+1}^i &= p_t^i + \omega \otimes s, \\
    i &= 1, 2, \ldots, A,
\end{align*}
\]

where \( p_i \) represents the \( i^{th} \) individual position in the population and the superscript indicates the current generation of the individual. The product \( \otimes \) means entry-wise multiplications and \( \omega \) represents the step adjustment factor which should be related to the scales of the problem of interest. \( A \) is the population size and \( s \) represents a random step. As the implementation of the Lévy distribution mathematical model is relatively complicated, the Mantegna algorithm is commonly used in practical applications to simulate the Lévy distribution. The algorithm model is as follows.

The random step length \( s \) is calculated as

\[
s = \frac{\mu}{|v|^{1/\zeta}},
\]

where \( \mu \) and \( v \) are Gaussian distributions and satisfy Equations (13) and (14):

\[
\mu \sim N(0, \sigma_\mu^2),
\]

\[
v \sim N(0, \sigma_v^2),
\]

where \( \sigma_v = 1 \) and \( \sigma_\mu \) is expressed as follows:

\[
\sigma_\mu = \left[ \frac{\Gamma(1 + \zeta)}{\Gamma((1 + \zeta)/2) \zeta^{2(1-\zeta)/2}} \right]^{1/\zeta}.
\]

In (15), the value of symbol \( \zeta \) is generally 1.5.

Taking the coordinate origin \((0, 0)\) as the initial point, the Lévy flight trajectory of the generation 100 is shown in Figure 3.

Figure 3 verifies the characteristics of the combination of long and short jump distances of Lévy flight, and it is very helpful to fully explore the solution space. After the Gaussian probability model of EDA is added to ALO, the search accuracy of ALO is improved, but the number of iterations is limited in practical application, and it is particularly important to accelerate the speed of jumping out of the local optimum. According to the Lévy perturbation method made by the PSO algorithm in [58], the new strategy, which integrated the step shrinkage factor and the particle swarm inertia weight parameter with similar function characteristics (which should follow the previous step to strengthen the global convergence ability and the feature of strengthening the local optimization ability in the later stage), is proposed and applied to ALO. Specific steps are as follows.

Step 1. After each iteration of ALO, it is judged whether the fitness value of the elite individual changes within 3 generations. If there is no change, enter Step 2. Otherwise, enter the next iteration.

Step 2. Take the elite individual as the parent and perform \( g \) Lévy flights to generate a candidate set containing \( g \) points. Elite individuals follow (16)–(18) for Lévy disturbance. One can see that

\[
P_t = \{p_t^1, p_t^2, \ldots, p_t^g\},
\]

\[
p_t^i = e_t + \omega \otimes s, \quad i = 1, 2, \ldots, g,
\]

\[
\omega = \omega_{\max} - \left(\omega_{\max} - \omega_{\min}\right) \frac{t}{T},
\]

where \( P_t \) is the candidate set of the \( g \) elements of the \( t^{th} \) generation and \( p_t^i \) represents the \( i^{th} \) element in \( P_t \). \( e_t \) represents the elite individuals obtained by the generation \( t \) fused with EDA and the symbol \( \omega t \) represents a linearly decreasing random step adjustment factor, where its maximum and minimum values are \( \omega_{\max} = 0.8 \) and \( \omega_{\min} = 0.1 \), respectively.

Step 3. Judge whether the candidate point set is superior to the elite individual of the current generation. And if so, replace the elite individual of the current generation with the elite individual of the next generation. Otherwise, elite individuals of this generation are retained. One can see that

\[
\begin{align*}
    &\begin{cases}
        e_{t+1} = \min\left(f(P_t)\right) & \text{if } \min\left(f(P_t)\right) < f(e_t), \\
        e_{t+1} = e_t & \text{otherwise}.
    \end{cases}
\end{align*}
\]

The new improved algorithm based on the above two strategies is called ELALO. To summarize, the process of ELALO algorithm is shown in Figure 4.
2.3. Verification with Simulation Studies. In order to test the optimize performance of ELALO, sphere function \((F_1)\), Ackley function \((F_2)\), Lévy function \((F_3)\), and Rosenbrock’s valley function \((F_4)\) are used in this simulation experiment. The specific expressions are shown in reference [59].

The sphere function is a typical nonlinear symmetric benchmark function and is mainly used to test the solution accuracy and speed of the algorithm in practical applications. The Ackley function is a multimodal function with particularly dense peaks and troughs. Lévy function is a complex multimodal function, where many algorithms will fall into the premature state and cannot get the global optimal solution. Multimodal functions are mainly used to test the algorithm’s ability to jump out of local optimum. The Rosenbrock’s valley function is a typically ill-conditioned, non-convex, and uni-modal function which is difficult to minimize, and there is an obvious correlation between variables. The mesh graphic of 4 benchmark functions \((D = 2)\) is shown in Figure 5.

According to the recommended values in references [32, 55–57], the basic parameters of the four algorithms are the same, such as the population size \(A = 500\), and the maximum number of iterations \(T = 2000\). And in Lévy flight process, the time \(g\) is set to 100. Other parameters of PSO and GA are set according to Ref. [17, 21], as shown in Table 1.

Each algorithm runs 50 times independently. The dimension of the four benchmark functions is \(D = 15\). The mean is used as the accuracy index, the best is used as the global optimization performance index, and the variance is as the stability index. The \(p\)-values mean the result of the Wilcoxon rank-sum test over benchmark functions. The statistical results are shown in Table 2.

It can be seen from Table 2 that the mean value of ELALO algorithm in function \(F_3\) is 4.76E-15. Compared with 1.79E-01 corresponding to initial ALO, ELALO algorithm has 14 orders of magnitude advantages and has similar performance in functions \(F_1, F_2,\) and \(F_4\), which fully indicates that the convergence accuracy of the improved algorithm has been significantly improved. In the test function \(F_2\), the optimal value of ELALO algorithm is 2.95E-13. Compared with 1.30E-06 corresponding to initial ALO, its global optimization performance is improved. And there are similar...
conclusions on functions $F_1$, $F_3$, and $F_4$. The variance of ELALO algorithm in function $F_1$ is 3.21E-25. Compared with the corresponding variance of ALO algorithm (4.73E+04), it has an advantage of nearly 29 orders of magnitude, which strongly proves the stability of ELALO and has similar conclusions on functions $F_1$, $F_3$, and $F_4$. Compared with PSO, ELALO algorithm is better than PSO except the $F_3$ function. When solving the multimodal function $F_3$, the average accuracy of the solution is the same order of magnitude, but its variance is less than PSO, indicating that the stability of ELALO algorithm is better than PSO. Compared with GA, there is a similar situation, except for solving $F_4$. When solving the ill-conditioned uni-modal function $F_4$, the stability of GA is better than ELALO. Overall, the performance of ELALO algorithm has been significantly improved.

The Wilcoxon rank-sum test is a famous nonparametric test in statistics theory that can be applied to determine if two sets of solutions (population) are different statistically significant or not. In this way, each set of pairs in both populations is compared to calculate and analyze their numerical differences. It also tests the null hypothesis as to whether both populations are of the same distribution. In short, the Wilcoxon rank-sum test returns a numerical result called $p$-value. A $p$-value determines the significance level of two algorithms. In this work an algorithm is statistically significant if and only if it results in a $p$-value less than 5%. The $p$ values in Table 2 also show that this superiority is statistically significant since all $p$-values are much less than 5%, which further reflect the robustness of ELALO algorithm.

In order to intuitively distinguish the performance characteristics of the two algorithms, the convergence curves of the two algorithms in each test function are shown in Figure 6. In particular, the value on the convergence curve is the average after 50 independent operations of the two algorithms.

In Figure 6, the abscissa is the number of iterations of the algorithm, and the maximum number of iterations is 2000 generations. In order to make a clear distinction between the algorithm iteration curves, the logarithm of the fitness value with base 10 is taken as the ordinate. The optimization accuracy of ELALO has been significantly improved. When the algorithm reaches the same accuracy in the early stage,
Table 2: Comparison results in four benchmark functions.

<table>
<thead>
<tr>
<th>benchmark function</th>
<th>Algorithm</th>
<th>Mean</th>
<th>Best</th>
<th>Variance</th>
<th>p values</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>ELALO</td>
<td>4.46E-13</td>
<td>6.23E-15</td>
<td>9.46E-22</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>ALO</td>
<td>7.19E-09</td>
<td>2.13E-10</td>
<td>1.57E-19</td>
<td>7.07E-18</td>
</tr>
<tr>
<td></td>
<td>PSO</td>
<td>5.80E-08</td>
<td>9.77E-09</td>
<td>6.60E-08</td>
<td>7.07E-18</td>
</tr>
<tr>
<td></td>
<td>GA</td>
<td>3.01E-06</td>
<td>1.43E-07</td>
<td>2.94E-06</td>
<td>7.07E-18</td>
</tr>
<tr>
<td>F2</td>
<td>ELALO</td>
<td>1.33E-12</td>
<td>2.95E-13</td>
<td>3.17E+00</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>ALO</td>
<td>4.84E-05</td>
<td>1.30E-06</td>
<td>1.08E+01</td>
<td>3.31E-20</td>
</tr>
<tr>
<td></td>
<td>PSO</td>
<td>2.08E+01</td>
<td>2.05E+01</td>
<td>1.22E-01</td>
<td>3.31E-20</td>
</tr>
<tr>
<td></td>
<td>GA</td>
<td>7.37E-05</td>
<td>3.71E-05</td>
<td>1.88E-05</td>
<td>3.31E-20</td>
</tr>
<tr>
<td>F3</td>
<td>ELALO</td>
<td>4.76E-15</td>
<td>1.87E-15</td>
<td>3.21E-25</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>ALO</td>
<td>1.79E-01</td>
<td>4.69E-03</td>
<td>4.73E+04</td>
<td>1.52E-05</td>
</tr>
<tr>
<td></td>
<td>PSO</td>
<td>3.45E-15</td>
<td>2.46E-29</td>
<td>2.28E-14</td>
<td>1.60E-02</td>
</tr>
<tr>
<td></td>
<td>GA</td>
<td>1.83E-11</td>
<td>1.75E-16</td>
<td>4.77E-11</td>
<td>4.68E-04</td>
</tr>
<tr>
<td>F4</td>
<td>ELALO</td>
<td>1.42E+00</td>
<td>8.28E-01</td>
<td>1.56E+02</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>ALO</td>
<td>2.30E+02</td>
<td>5.07E+01</td>
<td>2.44E-02</td>
<td>2.27E-02</td>
</tr>
<tr>
<td></td>
<td>PSO</td>
<td>5.65E+02</td>
<td>1.77E-02</td>
<td>1.38E+03</td>
<td>7.07E-18</td>
</tr>
<tr>
<td></td>
<td>GA</td>
<td>1.96E+00</td>
<td>5.57E-04</td>
<td>2.26E+00</td>
<td>2.05E-09</td>
</tr>
</tbody>
</table>

Figure 6: Comparison of convergence curve. (a) Curve comparison on $F_1$, (b) curve comparison on $F_2$, (c) curve comparison on $F_3$, and (d) curve comparison on $F_4$. 

Table 2: Comparison results in four benchmark functions.
the convergence speed of the ELALO is also higher than the initial ALO. Hence, one can see that the proposed ELALO algorithm has a good convergence.

2.4. Time Complexity Analysis. Time complexity is the key factor of algorithm performance and reflects the operation efficiency of the algorithm. In ALO, the population size is $A$, the individual dimension is $D$, and the maximum number of iterations is $T$. Suppose that the time to generate the random number and calculate the position of randomly distributed ants in each dimension is $\delta_1$, the time to calculate the fitness value of the objective function according to the ant position is $f(D)$, and the time to sort according to the ant fitness and select the optimal solution is $\delta_2$. Because the initial position of the ant and antlion is the same, the initial position of the elite antlion is the optimal solution selected from ant population. Therefore, the time complexity of population initialization stage is $O(A(D\delta_1 + f(D) + \delta_2)) = O(D + f(D))$. After entering the iterative process, when the antlion is selected is $\delta_3$, the time when the ants generate the position and judge the walking direction around an antlion is $\delta_4$ and $\delta_5$, respectively, the time for the ant to change the boundary in the process of walking is $\delta_6$ and the time when the ant obtains the new position after walking is $\delta_7$. The time to calculate the fitness value according to the ant position is $f(D)$ and the time of fitness value comparison and position replacement between ant and antlion is set to $\delta_8$. So, the time complexity of this stage is $O(T(A(D\delta_1 + \delta_2 + \delta_7 + f(D) + \delta_6) + \delta_5 + \delta_4)) = O(D + f(D))$. Therefore, the maximum time complexity of ALO algorithm is $O(D + f(D)) + O(f(D)) = O(D + f(D))$ [60].

The population size, dimension, maximum number of iterations, and the time to solve the value of objective or fitness function of ELALO algorithm are the same as those of ALO. Therefore, the time complexity of population initialization stage is the same as that of ALO; that is, $O(A(D\delta_1 + f(D) + \delta_2)) = O(D + f(D))$. After entering the iteration, suppose that the time to construct the Gaussian distribution probability model with EDA is $\tau_1$, the time to eliminate the individuals with poor fitness value is $\tau_2$, the time to generate new individuals according to EDA is $\tau_3$, the time to generate elite individuals by Lévy flight is $\tau_4$, the time to compare the fitness values of the old and new elite individuals is $\tau_5$, and the time to judge whether to replace is $\tau_6$. Other calculation times are the same as those in ALO algorithm. So, the time complexity of this stage is $O(T(A(D\delta_1 + \delta_2 + 2f(D) + \delta_6) + 3\tau_3 + 3\tau_2 + 3\tau_3 + \tau_5 + \tau_6 + \delta_5 + \delta_4)) = O(f(D))$. Therefore, the total time complexity of ELALO algorithm is also $O(D + f(D)) + O(f(D)) = O(D + f(D))$.

It can be seen from the above analysis that compared with the initial ALO, the time complexity of ELALO algorithm does not change, nor does it reduce the operation efficiency of the original algorithm.

3. ELALO-ELM Network Model

In this section, the mentioned ELALO algorithm is merged with the ELM network to develop a new ELALO-ELM network model for pattern classification.

3.1. ELM Network Fundamentals. The ELM is a new artificial neural network model for single-hidden layer feedforward neural networks (SLFNs), which randomly chooses hidden nodes and analytically determines the output weights of SLFNs [61]. In theory, ELM tends to provide good generalization performance at extremely fast learning speed.

Suppose the set of $N$ training samples is $\{X, Y\} = \{x_i, y_i\}_{i=1}^{N}$, where $x_i = [x_{i1}, x_{i2}, \ldots, x_{iD}]^T \in \mathbb{R}^D$, $y_i = [y_{i1}, y_{i2}, \ldots, y_{im}]^T \in \mathbb{R}^m$, $D$ is the feature dimension contained in the sample set, and $m$ is the total number of categories in the sample set. Supposing the number of hidden layer nodes is $L$ and the excitation function is $g(x)$, then the network output expression is shown as follows:

$$f_L(x) = \sum_{i=1}^{L} \xi_i g(a_i \cdot x_i + b_i), \quad x_i \in \mathbb{R}^D, a_i \in \mathbb{R}^D,$$

$$\xi_i \in \mathbb{R}^m, \quad i = 1, 2, \ldots, L,$$  \hspace{1cm} (20)

where $a_i = [a_{i1}, a_{i2}, \ldots, a_{im}]^T$ represents the weight matrix from the input layer to the $i^{th}$ hidden layer node, $b_i$ is the threshold of the $i^{th}$ hidden layer node, and $\xi_i = [\xi_{i1}, \xi_{i2}, \ldots, \xi_{im}]^T$ is the output weight of the $i^{th}$ hidden layer node.

If the network can accurately approximate $N$ samples, there is $a_i, b_i$ and $\xi_i$ satisfying

$$f_L(x) = \sum_{i=1}^{L} \xi_i g(a_i \cdot x_i + b_i) = y_i, \quad i = 1, 2, \ldots, L.$$  \hspace{1cm} (21)

Then, (21) can be simplified as follows:

$$H\xi = Y.$$  \hspace{1cm} (22)

$H$ represents the output matrix of the hidden layer of the network. In operation, the initial weight and threshold matrix are randomly assigned by the program, so that the training of the ELM model is completely transformed into the least squares solution problem in the mathematical sense. By (22), the least square solution form of the output weight matrix can be expressed as

$$\tilde{\xi} = H^+Y,$$

where $H^+$ represents the Moore-Penrose generalized inverse of matrix $H$.

3.2. ELALO-ELM Network Model. The steps to construct the ELALO-ELM model are as follows:

1. Determine the number $k$ of ELM hidden layer nodes and the number of input layer features $d$.

2. Set the ELALO individual dimension $D = k(d + 1)$ and initialize the parameters of ELALO, and then any individual $\theta$ in the algorithm can be expressed as

$$\theta = [\rho_1, \rho_2, \ldots, \rho_{k(d+1)}].$$  \hspace{1cm} (24)

3. Note that the initial weight matrix of ELM is $W$ and the threshold matrix is $B$, input the individual $\theta$ into ELM, and calculate the fitness value of $\theta$ by selecting
the appropriate fitness function. Then, $W$ and $B_i$ can be expressed as

$$W = \begin{bmatrix} \rho_1 & \rho_2 & \cdots & \rho_k \\ \rho_{k+1} & \rho_{k+2} & \cdots & \rho_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{(d-1)k+1} & \rho_{(d-1)k+2} & \cdots & \rho_{dk} \end{bmatrix}$$

$$B_i = \begin{bmatrix} \rho_{dk+1} & \rho_{dk+2} & \cdots & \rho_{k(d+1)} \end{bmatrix}.$$

(4) Repeat step 3 for each generation of ELALO population individuals to find the weight and threshold corresponding to the optimal fitness value.

(5) Assign the optimal weights and thresholds calculated by ELALO to ELM, complete the establishment of the ELALO-ELM model, and input the test set into the model to evaluate the performance of the model.

Based on the above steps, the realization process of the ELALO-ELM model is shown in Figure 7.

### 4. Using ELALO-ELM Network for Early Fault Diagnosis of Roller Bearings

This section uses the fault data from the drive end of the rolling bearing fault platform of Case Western Reserve University in the United States [62]. The experiment uses electric spark technology to cause damage to different parts of the bearing in different sizes to simulate the early fault of the roller bearing. The acceleration vibration sensor is used to collect bearing signal, and sampling frequency is 12 kHz [63].

The damage degree of the rolling bearing fault is distinguished by the diameter of the fault. 8 types of rolling bearing states of different sizes are extracted, rolling ball faults, outer ring faults, and normal bearings to form a fault set, which are denoted as $FM = \{fm_1, fm_2, fm_3, fm_4, fm_5, fm_6, fm_7, fm_8\}$. The $fm_1$, $fm_2$, $fm_3$, and $fm_4$, respectively, represent the faults of the rolling ball with diameters of 0.1778 mm, 0.3556 mm, 0.5334 mm, and 0.7112 mm, and $fm_5$, $fm_6$, and $fm_7$ represent the faults of the outer ring with diameters of 0.1778 mm, 0.3556 mm, and 0.5334 mm, respectively. $fm_8$ means normal bearing. Using data of different fault categories as the input samples of the ELALO-ELM model can make more accurate performance evaluation of the model. According to literature [64], the sampling points of each group of data to the original data are 2048. The time-domain waveforms of the 8 bearing states are shown in Figure 8.

#### 4.1. Fault Feature Extraction Based on LMD and Sample Entropy

**4.1.1. Introduction of Estimation of Distribution Algorithm.**

Local mean decomposition (LMD) [65] is a feature extraction method for adaptive time-frequency analysis of non-stationary signal. The vibration signals of early failures
of rolling bearings are not easy to find obvious features, but they have the characteristics of non-linearity and non-stationarity. LMD can efficiently extract early fault characteristics of rolling bearings, and thus improve the effectiveness and correlation of candidate feature sets. LMD generally decomposes the original signal $x(t)$ through (26) and decomposes it into several PF($t$) components that conform to the essential law of characteristics and a residual component $\mu_k(t)$ with weak characteristic performance. This component can describe the signal characteristics in more details. The detailed steps of this method can be found in the literature [65], which will not be repeated here.

$$x(t) = \sum_{p=1}^{k} \text{PF}_p(t) + \mu_k(t). \quad (26)$$

### 4.1.2. Symbolic Entropy Theory

Symbol entropy [66] is a signal complexity measurement method based on information entropy, which is simple and efficient. A random sequence $\{a_n\}$ is provided, and the sequence is symbolized by

$$s_n = \begin{cases} 1 & a_n > \bar{m} \\ 0 & a_n < \bar{m} \end{cases} \quad (27)$$

where $\bar{m}$ is the mean value of sequence $\{a_n\}$. After converting $\{a_n\}$ into symbol sequence $\{s_n\}$, which is composed of 0 and 1, it needs to be decomposed into multiple short symbol sequences by

$$\bar{s}(i) = s(i), s(i + 1), \ldots, s(i + (L - 1)) \quad i = 1, 2, \ldots, B - L + 1. \quad (28)$$

$L$ represents the length of each symbol sequence after decomposition, and $B$ represents the total length of the original sequence $\{s_n\}$. Convert all sequences $\bar{s}(i)$ into decimal sequences and the code sequence can be obtained [67]. For example, the original symbol sequence $\{s_n\}$ is $\{100101001\}$, when $L = 3$, it can be decomposed into a short sequence of $\{100, 101, 010, 101, 010, 100, 001\}$ and converted into a decimal coding sequence as $\{4,1,2,5,2,4,1\}$.

Then, the entropy of the above symbol sequences is calculated according to

$$H_s(L) = \frac{1}{\log M} \sum \log P_{m,L} \log (P_{m,L}). \quad (29)$$

where $H_s(L)$ represents the symbol entropy value, $P_{m,L}$ is the relative overall occupancy rate of each digital code in the decimal code sequence, and $M$ represents the number of 0–9 digital categories in the sequence.

Figures 9(a)–9(h) show the LMD exploded view of 8 different states of the rolling bearing, respectively.

It can be seen from Figure 9 that the bearing signal is mainly concentrated in the first 3 PF component signals after LMD. Therefore, the symbol entropy of the PF component is calculated separately and used as the feature quantity of the input classifier. And 50 sets of samples are taken for each state of the bearing, and the PF symbol entropy is calculated as shown in Table 3.

It can be seen from Table 3 that the difference between the PF symbol entropy features based on LMD is obvious, and the distribution is stable and the aggregation degree is...
Figure 9: Continued.
high. Using it as the input feature of the classifier can significantly reduce the learning difficulty of the classifier (Table 4).

4.2. Experimental Results. Since the content of this section mainly uses the bearing data set to evaluate the performance of the ELALO-ELM model, the performance evaluation index can be measured by the mean value of the failure recognition rate. So, the fitness function is constructed as

$$\max f(x) = \frac{1}{\text{count}} \sum_{i=1}^{\text{count}} \text{accuracy}_i,$$

where the count represents the bearing state category and \( \text{accuracy}_i \) represents the data recognition rate percentage of the \( i \)th category.

![Figure 9: FF1–PF6 signal of the 8 bearing states: (a) PF1–PF6 signal of LMD on \( fm_1 \), (b) PF1–PF6 signal of LMD on \( fm_2 \), (c) PF1–PF6 signal of LMD on \( fm_3 \), (d) PF1–PF6 signal of LMD on \( fm_4 \), (e) PF1–PF6 signal of LMD on \( fm_5 \), (f) PF1–PF6 signal of LMD on \( fm_6 \), (g) PF1–PF6 signal of LMD on \( fm_7 \), (h) PF1–PF6 signal of LMD on \( fm_8 \).](image)

![Table 3: The first 3 PF symbol entropy of 8 states of bearings.](image)

<table>
<thead>
<tr>
<th>bearing status</th>
<th>Index</th>
<th>PF1</th>
<th>PF2</th>
<th>PF3</th>
</tr>
</thead>
<tbody>
<tr>
<td>( fm_1 )</td>
<td>Mean</td>
<td>1.938</td>
<td>1.913</td>
<td>1.565</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
<td>0.0002</td>
<td>0.0074</td>
<td>0.0019</td>
</tr>
<tr>
<td>( fm_2 )</td>
<td>Mean</td>
<td>2.020</td>
<td>1.884</td>
<td>1.542</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
<td>0.0015</td>
<td>0.0054</td>
<td>0.0023</td>
</tr>
<tr>
<td>( fm_3 )</td>
<td>Mean</td>
<td>2.031</td>
<td>1.872</td>
<td>1.563</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
<td>0.0003</td>
<td>0.0035</td>
<td>0.0026</td>
</tr>
<tr>
<td>( fm_4 )</td>
<td>Mean</td>
<td>1.917</td>
<td>1.920</td>
<td>1.622</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
<td>0.0003</td>
<td>0.0034</td>
<td>0.0022</td>
</tr>
<tr>
<td>( fm_5 )</td>
<td>Mean</td>
<td>2.026</td>
<td>1.873</td>
<td>1.559</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
<td>0.0038</td>
<td>0.0055</td>
<td>0.0020</td>
</tr>
<tr>
<td>( fm_6 )</td>
<td>Mean</td>
<td>2.096</td>
<td>1.872</td>
<td>1.564</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
<td>0.0003</td>
<td>0.0061</td>
<td>0.0025</td>
</tr>
<tr>
<td>( fm_7 )</td>
<td>Mean</td>
<td>2.076</td>
<td>1.801</td>
<td>1.493</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
<td>0.0004</td>
<td>0.0054</td>
<td>0.0019</td>
</tr>
<tr>
<td>( fm_8 )</td>
<td>Mean</td>
<td>1.933</td>
<td>1.585</td>
<td>1.227</td>
</tr>
<tr>
<td></td>
<td>Variance</td>
<td>0.0002</td>
<td>0.0040</td>
<td>0.0003</td>
</tr>
</tbody>
</table>

![Table 4: Feature input and ideal output.](image)

<table>
<thead>
<tr>
<th>Bearing status</th>
<th>Input feature</th>
<th>PF1</th>
<th>PF2</th>
<th>PF3</th>
<th>Ideal output</th>
</tr>
</thead>
<tbody>
<tr>
<td>( fm_1 )</td>
<td></td>
<td>1.943</td>
<td>2.087</td>
<td>1.575</td>
<td>1000 0000</td>
</tr>
<tr>
<td>( fm_2 )</td>
<td></td>
<td>2.029</td>
<td>1.899</td>
<td>1.426</td>
<td>0100 0000</td>
</tr>
<tr>
<td>( fm_3 )</td>
<td></td>
<td>2.051</td>
<td>1.809</td>
<td>1.569</td>
<td>0010 0000</td>
</tr>
<tr>
<td>( fm_4 )</td>
<td></td>
<td>1.903</td>
<td>1.978</td>
<td>1.488</td>
<td>0001 0000</td>
</tr>
<tr>
<td>( fm_5 )</td>
<td></td>
<td>2.026</td>
<td>1.798</td>
<td>1.538</td>
<td>0000 1000</td>
</tr>
<tr>
<td>( fm_6 )</td>
<td></td>
<td>2.111</td>
<td>1.955</td>
<td>1.578</td>
<td>0000 0100</td>
</tr>
<tr>
<td>( fm_7 )</td>
<td></td>
<td>2.093</td>
<td>1.851</td>
<td>1.400</td>
<td>0000 0010</td>
</tr>
<tr>
<td>( fm_8 )</td>
<td></td>
<td>1.944</td>
<td>1.656</td>
<td>1.229</td>
<td>0000 0001</td>
</tr>
</tbody>
</table>

![Figure 10: The average recognition rate curve.](image)

Because the number of hidden layer nodes of ELM has a great influence on the results of the model. According to the literature [68], the number of hidden layer nodes of ELM is selected in order from 1 to 80. And taking (26) as the evaluation index, the curve of the average recognition rate with hidden layer nodes is obtained. As shown in Figure 10, the peak value of the curve shows that the best number of hidden layer nodes is 56.
The ELM, ALO-ELM, and ELALO-ELM network models are used to conduct comparative experiments. Set population size $A = 500$, Lévy flight times $g = 100$, $T = 300$. The three PF symbol entropy features are input to the classifier, so the input node is 3. And the number of bearing status labels is 8. So, the output node is also 8. Since the hidden layer node is set to 56, the method ELALO-ELM model establishment according to Section 3.3 is as follows: $D = 56 \times (3 + 1) = 224$.

The actual output results of the standard ELM and ELALO-ELM networks are shown in Table 5. The actual output of the ELM network is far from the ideal value, and classification-errors occur in faults fm2 and fm3. The actual output of the ELALO-ELM network is more consistent with the ideal output, indicating that the network model of the fusion improved algorithm ELALO has excellent performance and can more accurately distinguish the different fault characteristics of rolling bearings.

The comparison results of the recognition rates of the three networks solved 20 times are shown in Table 6. The recognition rate of ELM is the lowest when the bearing status are fm3 and fm5, reaching 23.33% and 44.03%, respectively. However, the overall recognition rate of new ELM models after parameter optimization is no less than 62.78%, and the recognition rate of each bearing status is significantly improved. When the fault type is fm7, the recognition rate reaches 98.41%, which is 7.46% higher than the original ELM network. It shows that the ALO algorithm fused in ELM parameter optimization can significantly improve the classification performance. Among them, ELALO-ELM has the best network recognition rate, with an average recognition rate of 88.00% and has even a recognition rate of 100% for fm7, which is 16.26% and 2.09% higher than ELM and ALO-ELM respectively. It verifies the effectiveness and superiority of the new ELALO-ELM to improve accuracy of classification model on the bearing early fault.

The broken lines of the recognition rates obtained by using the three networks model for 20 times are shown in Figure 11. The abscissa represents the experiment number, and the ordinate is the bearing status recognition rate calculated by the 3 networks model in each experiment. The fault recognition rate obtained by using only the ELM network reached a peak of 79.65% at the experimental number 12, and the worst value of the ELM fault recognition rate based on the ALO algorithm was 80.94% at the experimental number 17, which shows that the recognition rate of ELM has been significantly improved after the ALO parameter optimization. And most points in the broken line corresponding to ELALO-ELM network model are higher than ALO-ELM, which shows that the overall classification accuracy on bearing early fault of ELALO-ELM is better than that of ALO-ELM.

In order to reflect the advantages of the ELALO-ELM network more intuitively, in this paper, the average of the 20 experimental iteration curves of the two ALO network fusion is shown in Figure 12. It can be seen from Figure 11 that the two kinds of network iteration curves stabilized and converged before 50 generations. Finally, the average recognition rate of ALO-ELM stabilized at 85.91%, and the average recognition rate of ELALO-ELM stabilized at

---

**Table 5: Actual output comparison of ELM and ELALO-ELM.**

<table>
<thead>
<tr>
<th>Status</th>
<th>ELM</th>
<th>ELALO-ELM</th>
</tr>
</thead>
<tbody>
<tr>
<td>fm1</td>
<td>0.512</td>
<td>0.790</td>
</tr>
<tr>
<td>fm2</td>
<td>-0.069</td>
<td>0.505</td>
</tr>
<tr>
<td>fm3</td>
<td>0.148</td>
<td>0.025</td>
</tr>
<tr>
<td>fm4</td>
<td>0.358</td>
<td>0.237</td>
</tr>
<tr>
<td>fm5</td>
<td>-0.029</td>
<td>0.019</td>
</tr>
<tr>
<td>fm6</td>
<td>-0.009</td>
<td>0.039</td>
</tr>
<tr>
<td>fm7</td>
<td>0.058</td>
<td>-0.023</td>
</tr>
<tr>
<td>fm8</td>
<td>0.034</td>
<td>-0.082</td>
</tr>
</tbody>
</table>

**Table 6: Recognition rates comparison of 3 networks.**

<table>
<thead>
<tr>
<th>Network</th>
<th>fm1</th>
<th>fm2</th>
<th>fm3</th>
<th>fm4</th>
<th>fm5</th>
<th>fm6</th>
<th>fm7</th>
<th>fm8</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELM</td>
<td>86.83</td>
<td>23.33</td>
<td>44.03</td>
<td>87.39</td>
<td>56.38</td>
<td>89.07</td>
<td>90.95</td>
<td>95.94</td>
<td>71.74</td>
</tr>
<tr>
<td>ALO-ELM</td>
<td>91.22</td>
<td>62.78</td>
<td>67.03</td>
<td>97.74</td>
<td>77.65</td>
<td>96.20</td>
<td>98.41</td>
<td>96.25</td>
<td>85.91</td>
</tr>
<tr>
<td>ELALO-ELM</td>
<td>94.40</td>
<td>62.78</td>
<td>72.39</td>
<td>99.02</td>
<td>78.46</td>
<td>98.53</td>
<td>100</td>
<td>98.42</td>
<td>88.00</td>
</tr>
</tbody>
</table>
88.00%. When the number of iteration steps is the same, the iteration curve corresponding to ELALO-ELM is always higher than that of ALO-ELM, indicating that the convergence accuracy of ELALO-ELM is always higher than that of ALO-ELM. When the two network models reach the same recognition rate in the early stage of iteration, the number of iteration steps required by ELALO-ELM is less than that of ALO-ELM, indicating that ELALO-ELM has higher convergence efficiency and faster convergence speed.

5. Conclusions

(1) This work proposes an improved ALO algorithm, called ELALO, integrates the EDA and the variable-step Lévy flight strategy. The test results on 4 benchmark functions and theoretical analysis show that the ELALO is the same as the original ALO in terms of computational complexity, but is not easy to fall into the local optimum and superior to ALO algorithm in terms of solution accuracy and convergence speed.

(2) Aiming at the highly nonlinear and strong coupling characteristics of early fault data of rolling bearing, a method combining LMD and symbolic entropy is proposed to extract the most effective features as the input features of classifier model, which effectively reduces the difficulty of classifier pattern recognition.

(3) Aiming at the defect of random initialization of initial weight and threshold in the calculation process of ELM network, ELALO is used to optimize its initial parameters to form a new pattern recognition model called ELALO-ELM. The experimental results show that new model can effectively overcome the inherent defect of falling into local optimization caused by random initialization of ELM network, resulting in high accuracy of early fault classification of rolling bearing. The ELALO-ELM model can also be used for other weak fault classification problems.

(4) In our future work, we will focus on more complex and weaker fault pattern classification and recognition, which is challenging to the global convergence performance of ELALO algorithm and the classification ability of ELALO-ELM model.

Appendix

A

Pseudo-code of ELALO algorithm

```c
//ELALO args initialization
Init all args of ELALO: static GAUSS_POP, ALO_POP, EDA_POP, ...

//Interface implementation
function void __ALO_INIT(void)
do
    Initialize ALO_POP
done
function EDA_POP __GAUSS_RND(ALO_POP)
do
    EDA_POP = __GAUSS_RND(ALO_POP) and Replace ALO_POP
done
function void __EDA_INIT(void)
do
    EDA_POP = __GAUSS_RND(ALO_POP)
done
function void __ELALO_INIT(void)
do
    ELALO-ELM
```

Figure 11: The recognition rate results of the three types of networks 20 times.

Figure 12: Comparison of two network iteration curves.
__ALO_INIT()
__EDA_INIT()
done

function ELITE_POP __ALO_SEARCH(ELALO_TEMP_POP)
do
  According to ALO walk
done

function FVAL __FITNESS_VALUE(POP)
do
  Calculate the fitness of POP
done

function CANDIDATE_SET __LEVY_FLIGHT(ELITE_POP)
do
  According to formula (11)
done

function BEST_POP __ELALO_SEARCH(void)
do
  ELALO_TEMP_POP = EDA_POP
  while (t < max_iterations)
do
    ELITE_POP = __ALO_SEARCH(ELALO_TEMP_POP)
    FVAL = __FITNESS_VALUE(ELITE_POP)
    if (FVAL get trapped in continuous 3 generation)
do
      CANDIDATE_SET = __LEVY_FLIGHT(ELITE_POP);
      if (__FITNESS_VALUE(CANDIDATE_SET) is good than __FITNESS_VALUE(ELITE_POP))
do
        Replace CANDIDATE_SET to ELITE_POP
        Update ELITE_POP
done
  else
    Reserve ELITE_POP
done
  Update ELALO_TEMP_POP
end while
BEST_POP = ELALO_TEMP_POP
return BEST_POP
done
//Main function implementation

void main
  __ELALO_INIT()
  BEST_POP = __ELALO_SEARCH()
done

Data Availability
The data used to support the findings of this study are included within the article.

Conflicts of Interest
The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments
This work was supported in part by the Anhui Province Natural Science Foundation under Grant No. 1808085ME127, the Overseas Visiting and Research Project for Outstanding Young Backbone Talents in Universities of Anhui Province under Grant No. gxxwfx2019041, the Innovation Project for Returned Overseas Students in Anhui Province under Grant No. 2020LCX013, the Key Research and Development Projects of Anhui Province under Grant No. 202004b11020006, the Scientific Research Foundation of Anhui Polytechnic University under Grant No. 2020YQ010, the Anhui Polytechnic University Research Initiation Fund for Introducing Talents under Grant No. 2019YQ004, the Anhui Polytechnic University Research Project under Grant No. Jxyj2019010, the Industrial Collaborative Innovation Fund of Anhui Polytechnic University and Jiuhu District under Grant No. 2021cyxtb9, the Open Project of Anhui Provincial Engineering Laboratory on Information Fusion and Control of Intelligent Robot under Grant No. IFCIR2020001, the Science and Technology Planning Project of Anhui Market Supervision and Administration Bureau under Grant no. 2021MK005, and the Open Project of Key Laboratory of Industrial Equipment Quality Big Data Ministry of Industry and Information Technology under Grant No. 2021-IEQBD-05.

References
Advances in Materials Science and Engineering 17


system using multi-objective ant lion optimizer and evidential reasoning,” *Scien


[62] “Case Western Reserve University Bearing Data Center Website,” 2020, [https://www.ecce.case.edu/laboratory/bearing/](https://www.ecce.case.edu/laboratory/bearing/).


