Scholar Publishing Group
International Journal of Big Data Intelligent Technology
https://doi.org/10.38007/IJBDIT.2024.050115
ISSN 2790-0932 Vol. 5, Issue 1: 137-152

Transformer Fault Diagnosis Based on KPCA-IHBA-DHKELM

Changshun Lv[1,a], Ziwei Zhu[1,b*], Yinuo Deng[1,c]

[1]*Information Engineering College, Nanchang University, Nanchang 330031, Jiangxi, China*
[a]*changshun323@gmail.com,* [b]*416100220252@email.ncu.edu.cn,*
[c]*409088220015@email.ncu.edu.cn*
[*]*Corresponding author*

Keywords: transformer; fault diagnosis; kernel principal component analysis; Improved Honey Badger Algorithm; Deep Hybrid Kernel Extreme Learning Machine

Abstract: This paper addresses the low diagnostic accuracy of dissolved gas analysis (DGA) in transformer faults by proposing an improved honey pot algorithm (IHBA) to optimize the deep mixed kernel extreme learning machine (DHKELM). First, the dissolved gas data in transformer oil is preprocessed to reduce the discrepancies in the magnitude of fault data. Kernel Principal Component Analysis (KPCA) is then applied to reduce dimensionality and extract effective features, thus decreasing the correlation among the data.Next, traditional honey pot algorithm (HBA) is improved by introducing Cubic chaotic mapping, random value perturbation strategies, elite tangent search, and differential mutation strategies. The performance of IHBA is tested using three typical benchmark functions, demonstrating that IHBA possesses stronger stability and optimization capabilities. IHBA is then utilized to optimize the parameters of DHKELM, establishing the IHBA-DHKELM model for transformer fault diagnosis. Finally, the features extracted via KPCA are used as the input set for the model and compared with various transformer fault diagnosis models. Simulation results indicate that IHBA-DHKELM achieves higher diagnostic accuracy for transformer faults.

1 Introduction

In modern power systems, transformers are crucial equipment, and their operational stability is directly related to the safety of the entire power grid. However, due to long-term operation and complex working environments, transformers may experience various faults. If these faults are not detected and diagnosed in a timely manner, they could lead to significant economic losses and safety risks. Therefore, developing an efficient and accurate method for transformer fault diagnosis holds great practical significance[1].

When a transformer experiences a fault or potential hazard, the proportions of gases dissolved in

its insulating oil (H_2, CH_4, C_2H_6, C_2H_4, C_2H_2) change. These variations have a nonlinear relationship with the types of transformer faults. Although many transformer fault diagnosis methods exist, Dissolved Gas Analysis (DGA) remains the most mainstream approach. Current transformer DGA fault diagnosis methods can be divided into traditional methods and artificial intelligence-based methods. Traditional methods include the Rogers Ratio Method[3], the IEC Three-Ratio Method[4], and the Duval Triangle Method[5]. These methods are simple and reliable, and have been widely used in fault diagnosis. However, as the number of fault types, data dimensions, and noise increase, the accuracy of traditional methods tends to decrease.With technological advancement, artificial intelligence (AI) methods have become increasingly popular. Bayesian networks[6], fuzzy logic[7], and machine learning[8,9] approaches are being applied more frequently in transformer fault diagnosis. These AI methods can improve diagnostic accuracy compared to traditional techniques. For example, literature [10] applied a BP neural network for transformer fault diagnosis, but the method suffers from slow convergence and a tendency to get stuck in local optima, requiring constant model adjustments in complex situations. Literature [11] utilized the Extreme Learning Machine (ELM) for engine fault diagnosis. While ELM offers fast learning speed and a simple structure, its randomness in hidden layer parameters can lead to poor model stability.To address the shortcomings of ELM, researchers have proposed the Hybrid Kernel Extreme Learning Machine (HKELM) and Deep Hybrid Kernel Extreme Learning Machine (DHKELM) models, which introduce kernel functions and use backpropagation algorithms to train models. This significantly enhances the model's expressive power and generalization ability. However, DHKELM models involve many hyperparameters, making fault diagnosis accuracy sensitive to manual tuning. As a result, more researchers are combining heuristic optimization algorithms with classifiers to improve performance.

Commonly used algorithms include Particle Swarm Optimization (PSO)[14], Whale Optimization Algorithm (WOA)[15], Sparrow Search Algorithm (SSA)[16], Grey Wolf Optimizer (GWO)[17], and Honey Badger Algorithm (HBA)[18]. The Grey Wolf Optimizer achieves good optimization results by adjusting the convergence factor and position weighting, but its convergence factor linearly decays in the later stages of iteration, making it prone to getting stuck in local optima[19]. The Sparrow Search Algorithm has strong global search capabilities and fast convergence speed, but it suffers from insufficient local search ability and is susceptible to the influence of initial individual distribution[20]. The Honey Badger Algorithm (HBA), first introduced in 2022, mimics the exploration and exploitation strategies of honey badgers searching for food, constantly adjusting the position of solutions to find the optimal solution. Compared to other optimization algorithms, HBA demonstrates strong global search capability and rapid convergence, excelling in solving complex optimization problems. However, like many intelligent algorithms, HBA also faces common drawbacks, such as being prone to local optima in later iterations and exhibiting insufficient global search capability.

To address the aforementioned issues, this paper proposes a method that couples an Improved Honey Badger Algorithm (IHBA) with a Deep Hybrid Kernel Extreme Learning Machine (DHKELM). Firstly, considering the impact of the nonlinear and high-dimensional characteristics of transformer fault data on diagnostic accuracy, Kernel Principal Component Analysis (KPCA) is used for dimensionality reduction. The reduced data not only retains the key features of the original data but also enhances the diagnostic speed. Secondly, the traditional Honey Badger Algorithm (HBA) is improved by incorporating Cubic chaotic mapping, random value disturbance strategy, elite tangent search, and differential mutation strategy, which is then applied to optimize the parameters of the DHKELM model. Finally, the proposed IHBA-DHKELM model is compared with multiple diagnostic models, and the results demonstrate that the IHBA-DHKELM model achieves higher accuracy and efficiency in diagnosing fault types.

2 Fault Feature Extraction

2.1 Fault Data Preprocessing

Due to the large span of numerical ranges in DGA data and the wide and uneven distribution of features, the accuracy of fault diagnosis can be reduced and the diagnosis time can be increased. Therefore, data preprocessing must be performed before DGA. Common data preprocessing methods include normalization and standardization.

(1)Normalization

$$x' = \frac{x - \min}{\max - \min} \tag{1}$$

(2)Standardization

$$x' = \frac{x - \mu}{\sigma} \tag{2}$$

where μ and σ are the mean and standard deviation of the sample data, respectively.

Table 1 shows the comparison of accuracy between the original data and the preprocessed data.

Table 1 Comparison of diagnostic results under different data preprocessing

Preprocessing Method	Diagnostic Accuracy/%
Original Data	52.3
Normalization	67.4
Standardization	76.3

2.2 Dimensionality Reduction of Multidimensional Fault Features

Through statistical analysis of transformer faults, it can be observed that when a transformer experiences a fault, the main gases are H_2, C_2H_4, CH_4, C_2H_6, C_2H_2 and TH, where TH represents the sum of H_2, C_2H_4, CH_4, C_2H_6, C_2H_2. When using these six gases as input parameters, the impact of data preprocessing on the model's fault diagnosis rate may be overlooked. Therefore, this paper pairs the aforementioned six gases in ratios to generate DGA data for transformer fault diagnosis, as shown in Table 2.

Table 2 Dissolved gas ratio in oil

Number	Feature Parameters	Number	Feature Parameters
X_1	H_2	X_9	H_2/TH
X_2	CH_4	X_{10}	CH_4/C_2H_6
X_3	C_2H_6	X_{11}	CH_4/C_2H_4
X_4	C_2H_4	X_{12}	CH_4/TH
X_5	C_2H_2	X_{13}	C_2H_6/C_2H_4
X_6	H_2/C_2H_4	X_{14}	C_2H_6/TH
X_7	H_2/CH_4	X_{15}	C_2H_4/TH
X_8	H_2/C_2H_6	X_{16}	C_2H_2/TH

KPCA is a nonlinear dimensionality reduction method based on kernel techniques. It effectively removes redundant features in the data and extracts nonlinear features that contain essential information. It addresses the limitations of PCA, which can only handle linear data, thereby enhancing modeling efficiency and performance.

In this study, KPCA is employed to reduce the dimensionality of high-dimensional nonlinear

DGA data, and the results are shown in Figure 1.

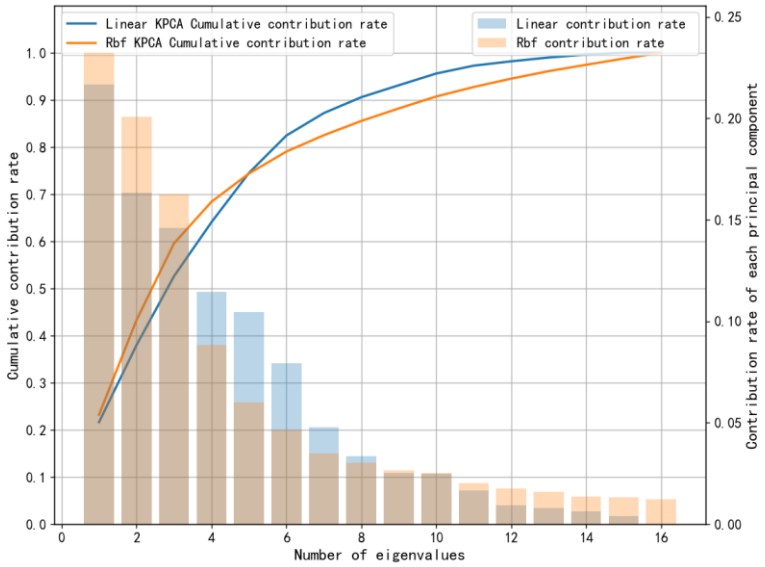


Fig. 1 Characteristic contribution rate and cumulative contribution rate of each principal component

As shown in Figure 1, the cumulative contribution rate of the first nine principal components reaches approximately 95%, which captures most of the required feature information. Therefore, the dimensionality of the reduced features is set to 9.

SVM fault diagnosis is performed on both the original and reduced-dimensional data, comparing the running time and diagnostic accuracy before and after dimensionality reduction, as shown in Table 3.

Table 3 Comparison before and after KPCA extraction

	KPCA before downscaling	KPCA after downscaling
Fault diagnosis accuracy/%	72.3	84.25
Running time/s	23.73	14.62

3 DHKELM Transformer Fault Diagnosis Model

3.1 Hybrid Kernel Extreme Learning Machine

Extreme Learning Machine (ELM) is a learning algorithm used for Single Hidden Layer Feedforward Neural Networks (SLFN)[21]. The weights and biases of the hidden layer are randomly generated and remain fixed, eliminating the need for the iterative optimization process found in traditional neural networks. This makes the training process very fast.

Assume there is a training dataset with N training samples $W=\{(x_t,y_t),t=1,2,...,N\}$, where x_i is the input and y_i is the output. If the number of training samples equals the number of hidden layer neurons, the learning process of ELM can be expressed through the least squares solution:

$$\beta' = H^{+}T \tag{3}$$

In the equation, β' represents the optimal weight matrix of the hidden layer output; H^{+} denotes the Moore-Penrose generalized inverse matrix of the hidden layer output matrix H; and T is the true

target output matrix.

Shang et al[22]. improved the generalization ability of ELM by introducing the kernel parameter HH^T. The kernel function of KELM is as follows:

$$\begin{cases} \Omega_{KELM} = HH^T \\ \Omega_{KELM_{i,j}} = h(x_i)h(x_j) = K(x_i, x_j) \end{cases} \tag{4}$$

Therefore, the model of KELM can be expressed as:

$$y(x) = \begin{bmatrix} K(x, x_1) \\ \mathrm{M} \\ K(x, x_N) \end{bmatrix} (\frac{E}{C} + \Omega_{KELM})^{-1} T \tag{5}$$

In the equation, the constant C is the penalty parameter; E is the identity matrix; and $h(x)$ is the transfer function.

To further enhance the learning and generalization capability of KELM, a new hybrid function is constructed using a linear weighting method, combining the Gaussian kernel function from the local kernel function and the polynomial kernel function from the global kernel function. The equation for this hybrid function is:

$$K(x, x_i) = \gamma \exp(-\frac{\|x, x_i\|^2}{2\sigma^2}) + (1-\gamma)((x \cdot x_i) + a)^b \tag{6}$$

In the equation, γ is the weight coefficient of the hybrid function; a and b are the constant parameter and the exponent parameter of the polynomial kernel function, respectively.

3.2 Deep Hybrid Kernel Extreme Learning Machine Based on Autoencoder

According to the condition that the input of the objective function must equal the output in the Extreme Learning Machine Autoencoder (ELM-AE) model[23], effective features can be extracted from the data, thereby improving classification accuracy. As shown in Figure 2, by incorporating the Extreme Learning Machine Autoencoder (ELM-AE) into the Deep Extreme Learning Machine (DELM), a Deep Hybrid Kernel Extreme Learning Machine (DHKELM) is constructed. This integration enhances the model's robustness and generalization ability, leading to improved classification performance and achieving a high level of fault diagnosis accuracy. The specific steps are as follows:

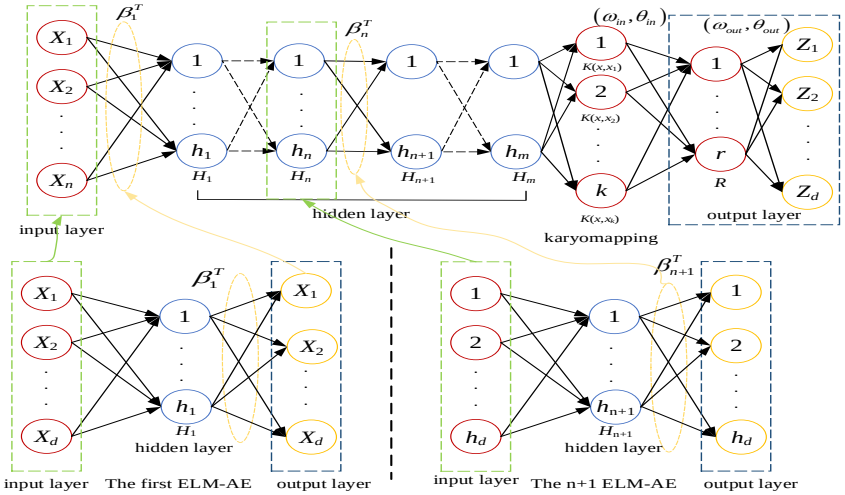


Fig. 2 DHKELM structure diagram

Step 1:Randomly generate orthogonal parameters a_n and b_n between the input layer and the hidden layer in the n-th ELM-AE.

Step 2:Calculate the output H_t of the temporary hidden layer and the weights of the n+1-th hidden layer.

$$H_t = g(a_t H_{t-1} + b_t) \tag{7}$$

$$\beta_{i+1} = \left(\frac{E}{C} + (H_t)^T H_t\right)^{-1} (H_t)^T H_n \tag{8}$$

In the equation, g(x) represents the activation function.

Step 3: Further stack the hidden layers obtained from the ELM-AE in Step 2. The output H_n of the n-th hidden layer serves as the input H_{t+1} for the n+1-th layer. The constructed multilayer structure is as follows:

$$\begin{cases} H_0 = X \\ H_{n+1} = g((\beta_{n+1})^T H_n), n = 0,1,....,m-1 \end{cases} \tag{9}$$

The expression for the output R of the fully connected layer is:

$$R = g\left(\omega_{in} H_{n+m} + \theta_{in}\right) \tag{10}$$

In the expression, ω_{in} and θ_{in} represent the input weights and biases of the fully connected layer.

Step 4: By performing parameter tuning, adjust the weights and biases between the output R of the last hidden layer and the fully connected layer. The output layer is constructed as a traditional fully connected neural network layer, yielding the reconstructed data Z_d, expressed as:

$$Z_d = g(\omega_{out} + \theta_{out}) = g(\omega_{out} g(\omega_{in} H_m + \theta_{in}) + \theta_{out}) \tag{11}$$

In the expression, ω_{out} and θ_{out} represent the output weights and biases of the fully connected layer.

The objective function of the DHKELM model is to minimize the error ε between the reconstructed data and the input data. Its expression is:

$$\varepsilon = \underset{\omega_{in},\theta_{in},\omega_{out},\theta_{out}}{\arg\min} \left\| Z_d - X_d \right\|_2^2 \tag{12}$$

Compared to traditional ELM, the DHKELM gradually extracts and combines the features of the data through a multi-layer structure while leveraging the nonlinear mapping capabilities of hybrid kernels to handle complex nonlinear problems. Its mathematical model integrates the fast learning of ELM with the powerful nonlinear capabilities of kernel methods, while also incorporating the hierarchical feature extraction characteristics of deep learning.

3 Honey Badger Optimization Algorithm and Its Improvement

3.1 Honey Badger Optimization

The Honey Badger Optimization Algorithm[24](HBA) is a nature-inspired swarm intelligence optimization algorithm that draws inspiration from the survival strategies of the honey badger. The HBA mimics the behavior of honey badgers when foraging for food and avoiding danger, using their foraging strategies to solve optimization problems. This algorithm is characterized by strong evolutionary capability, fast search speed, and robust optimization ability. Theoretically, the HBA consists of two phases: exploration and exploitation, which qualifies it as a global optimization algorithm. The steps of the HBA are detailed as follows. In HBA, the candidate solution population is represented as:

$$\begin{pmatrix} x_{11} & x_{12} & x_{13} & \mathrm{L} & x_{1D} \\ x_{21} & x_{22} & x_{23} & \mathrm{L} & x_{2D} \\ & \mathrm{L} & & \mathrm{L} & \\ x_{n1} & x_{n2} & x_{n3} & \mathrm{L} & x_{nD} \end{pmatrix} \tag{13}$$

where the position of the *i*-th honey badger is $x_i = \left[x_i^1, x_i^2, \mathrm{K}, x_i^D \right]$

(1)Initialization: Initialize the number of honey badgers (population size N) and their respective positions according to Equation (14):

$$x_i = lb_d + r_1 \cdot \left(ub_i - lb_i \right) \tag{14}$$

In the equation, r_1 is a randomly selected number from the interval [0,1]; x_i is the position of the *i*-th honey badger, representing a candidate solution in the population; lb_i and ub_i are the lower and upper bounds of the search domain, respectively.

(2) Defining Intensity: The intensity is related to the concentration of the prey's scent and the distance between the prey and the *i*-th honey badger. I_i represents the scent intensity of the prey; if the scent concentration is high, the honey badger will move faster, and vice versa. It is defined by Equation (15):

$$\begin{aligned} I_i &= r_2 \times \frac{S}{4\pi d_i^2} \\ S &= (x_i - x_{i+1})^2 \\ d_i &= x_{prey} - x_i \end{aligned} \tag{15}$$

In the equation, r_2 is a randomly selected number from the interval [0,1]; S represents the source intensity or concentration intensity. In Equation (15), d_i indicates the distance between the prey and the *i*-th honey badger.

(3) Updating the Density Factor: The density factor α controls the time-varying randomization to ensure a smooth transition between exploration and exploitation. The decreasing factor α, which reduces with the number of iterations, is given by Equation (16):

$$\alpha = C \times \exp(\frac{-t}{t_{\max}}) \tag{16}$$

In the equation, t_{max} is the maximum number of iterations, and C is a constant greater than or equal to 1 (default value is 2).

(4) Escaping Local Optima: This step, along with the next two steps, is designed to escape local optimum regions. In this case, the proposed algorithm utilizes a flag F to provide individuals with more opportunities to rigorously explore the search space by changing the search direction.

(5) Updating the Position of Agents: The position update process of the honey badger algorithm $\backslash x_{new}$ is divided into two parts: the "digging phase" and the "bee phase."

1) Digging Phase: In the digging phase, the honey badger performs movements similar to a heart shape, which can be simulated by Equation (17):

$$x_{new} = x_{prey} + F \times \beta \times I \times x_{prey} + F \times r_3 \times \alpha \times d_i \times \left| \cos(2\pi r_4) \times \left[1 - \cos(2\pi r_5) \right] \right| \tag{17}$$

In the equation, x_{prey} represents the position of the prey, which is the best position found so far—the global optimum; $\beta \geqslant 1$ (default value = 6) indicates the honey badger's ability to acquire food; d_i is the distance between the prey and the *i*-th honey badger, as described in Equation (15). r_3 r_4 and r_5 are three random numbers between 0 and 1. F serves as a flag to change the search direction, and its value is determined by Equation (18):

$$F = \begin{cases} 1 & \text{if } r_6 \le 0.5 \\ -1 & else \end{cases} \tag{18}$$

r_6 is a randomly selected number from the interval [0,1].

During the digging phase, the honey badger primarily relies on the scent intensity I of the prey x_{prey}, the distance d_i between the honey badger and the prey, and the time-varying search influence factor α. Additionally, during the digging activity, the honey badger may be influenced by any disturbances F, enabling it to find better prey positions.

Bee Phase: When the honey badger follows the honey guide bird to the hive, this can be simulated using Equation (19):

$$x_{new} = x_{prey} + F \times r_7 \times \alpha \times d_i \tag{19}$$

In the equation, r_7 is a randomly selected number from the interval [0,1]. Here, x_{new} refers to the new position of the honey badger, while x_{prey} is the position of the prey. The values of F and α are determined by Equations (18) and (16), respectively. From Equation (19), it can be seen that the honey badger searches in the vicinity of the prey position x_{prey} based on the distance information d_i of the prey found so far. During this phase, the search is influenced by the time-varying factor α. Additionally, the honey badger may encounter disturbances F.

3.2 Improved Honey Badger Algorithm

To address the shortcomings of the honey badger algorithm, this paper introduces Cubic chaotic mapping, random value perturbation strategy, elite tangent search, and differential mutation strategy to improve the traditional honey badger algorithm (HBA). The details are as follows.

(1)Cubic chaotic mapping

Since the initial population of the basic honey badger search algorithm is generated randomly, it cannot ensure that the individuals' initial positions are uniformly distributed in the search space, which affects the algorithm's search speed and optimization performance. To improve the initialization process of the honey badger algorithm, Cubic mapping is introduced to enhance the exploration of the initial population. The Cubic mapping is defined by Formula (20):

$$\begin{aligned} x_i &= l_b + (u_b - l_b) \cdot z_i \\ z_{i+1} &= \rho z_i (1 - z_i^2) \end{aligned} \tag{20}$$

In the equation, x_i represents the position of the i-th honey badger; l_b and u_b are the lower and upper bounds of the variable boundaries; Z_i is the Cubic chaotic sequence; and$\rho$is a constant, typically set to 3.

(2)Random value perturbation strategy

In the honey badger algorithm, the position of the honey badger is always updated based on the population's best value x_{prey}, which can lead to premature convergence as the population clusters around the optimal individual. To enhance the global optimization capability of the honey badger algorithm, a random search strategy is introduced, which determines the update strategy based on the value of coefficient A. When $|A| \geqslant 1$, a search strategy that perturbs random individuals is executed; otherwise, the honey badger's position is still updated based on x_{prey}. The expression for A is given by Equation (21):

$$A = 2 \times m \times r - m \tag{21}$$

In the equation, $m = 2 - 2 \times \frac{t}{t_{\max}}$,its value decreases linearly from 2 to 0, with r being a random

number in the interval (0, 1). From the above two formulas and the expression for *m*, it can be seen that the value of |A| generally shows a linear decreasing trend. In the early stages of iteration, the algorithm frequently executes the random search strategy to avoid premature clustering of the population, enhancing the exploration of honey badger individuals in the search space and improving global search capability.

At this time, the mathematical expressions for the honey badger's searching during the digging phase and the honey phase change to:

$$x_{new} = x_{rand} + F \times \beta \times I_i \times x_{rand} + F \times r_3 \times \alpha \times d_i \times \left|\cos(2\pi r_4) \times \left[1 - \cos(2\pi r_5)\right]\right| \tag{22}$$

$$x_{new} = x_{rand} + F \times r_7 \times \alpha \times d_i \tag{23}$$

At this point, the expression for d_i is as follows:

$$d_i = x_{rand} - x_i \tag{24}$$

(3)Elite tangent search and Differential mutation strategy

When the population finds the best value unchanged for three iterations, the following elite tangent search and differential mutation strategies are executed. In each iteration, the first half of the population with lower fitness values is designated as the elite subgroup, while the latter half is designated as the exploration subgroup.

1)The migration strategy of the elite subgroup.

When individuals are close to the current optimal solution, they search the surrounding area, a behavior known as local search, which can improve convergence speed and solution accuracy. Since the fitness of the elite subgroup is close to the current optimal solution, allowing the elite subgroup to engage in local development can enhance convergence speed and solution accuracy. The formula for the elite tangent search strategy is as follows:

$$X = \begin{cases} X_{prey} + step * \tan(\theta) * (rand * X_{prey} - X) & \text{if X=optS} \\ X_{prey} + step * \tan(\theta) * (X_{prey} - X) & \text{if X} \neq \text{optS} \end{cases} \tag{25}$$

$$\theta = rand * \frac{\pi}{2.1} \tag{26}$$

$$step = 10 * sign(rand - 0.5) * norm(optS) * \log(1 + 10 * \frac{\dim}{t}) \tag{27}$$

In the equation, norm()represents the Euclidean norm, X is the current solution, and *optS* is the current optimal solution used to guide the search process toward the best solution.

The honey badger individuals perform tangent searches based on the positions of the elite honey badgers, allowing for better exploration of the area around the optimal solution. This evolutionary strategy effectively utilizes the positional information of the current optimal solution, accelerating the convergence speed and improving solution accuracy.

2)Evolutionary Strategy of the Exploratory Subpopulation

As can be seen from the formula, in the honey badger algorithm, the position update of the honey badger individuals in the population generates new individuals around the current individual and the current optimal individual X_{prey}. This means that other individuals in the group move toward X_{prey}. If X_{prey} is a local optimal solution, as the iterations continue, the honey badger individuals in the population will gather around X_{prey}, resulting in poor population diversity and making the algorithm prone to premature convergence.

To address these issues, this paper adopts a differential mutation strategy. Inspired by the

mutation strategy in differential evolution algorithms, random differences are utilized with the current honey badger individual, the current optimal individual, and a randomly selected honey badger individual from the population to generate new individuals, as shown in formula (28):

$$X(t+1) = X_{rand1}(t) + F_0 * (X_{rand2}(t) - X_{rand3}(t)) \tag{28}$$

The formula is as follows: F_0 is the differential evolution scaling factor set to 0.4; t is the current iteration number; X_{rand1},X_{rand2},X_{rand3} are randomly selected honey badger individuals.

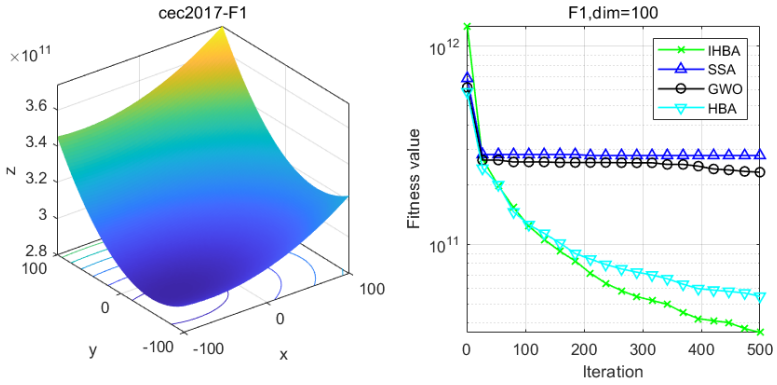
In the formula, F_0 is the differential evolution shrinkage factor set to 0.4; $t$is the current iteration number; and X_{rand1},X_{rand2},X_{rand3} are the randomly selected honey badger individuals.

3.2 Performance Testing of IHBA

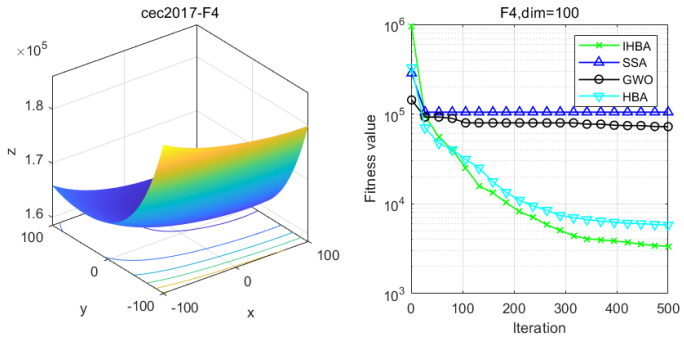
The optimization performance of IHBA is tested using three test functions from CEC 2017: F1, F4, and F7, as shown in Table 4. The results are compared with the GWO, SSA, and HBA algorithms. All four algorithms have a population size of 30 and a maximum of 1000 iterations. The optimization convergence curves are shown in Figure 3. It can be observed that all four functions converge sequentially with the increase in the number of iterations, with IHBA demonstrating the fastest convergence speed, significantly outperforming the other algorithms. This indicates that the IHBA algorithm possesses excellent optimization capabilities.

Table 4 Test functions

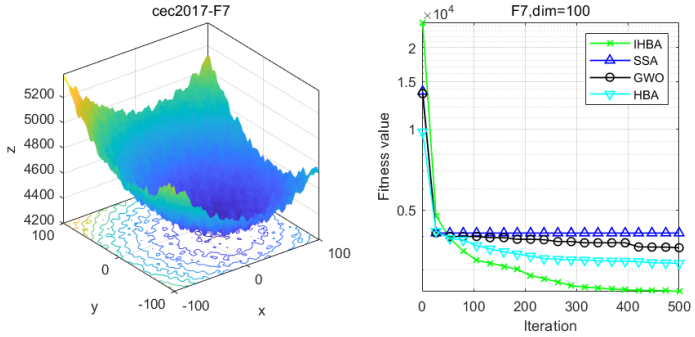
Test functions	Search interval	Theoretical optimum
$f_1(x) = x_1^2 + 10^6 \sum_{i=2}^{D} x_i^2$	$[-10\text{-}10]^d$	0
$f_4(x) = \sum_{i=1}^{D-1} (100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2)$	$[-10\text{-}10]^d$	0
$f_7(x) = \min(\sum_{i=1}^{D} (x_i - \mu_0)^2, dD + s\sum_{i=1}^{D} (x_i - \mu_1)^2) + 10(D - \sum_{i=1}^{D} \cos(2\pi z_i))$	$[-15\text{-}15]^d$	0



(a) F1 convergence curve



(b) F4 convergence curve



(c) F7 convergence curve

Fig. 3 Optimizing convergence curve

4 Transformer Fault Diagnosis Based on IHBA-DHKELM

The fault sample data is selected from reference [25] and the IEC TC 10 database. The dataset includes six different operating conditions of transformers: high energy discharge, low energy discharge, partial discharge, high-temperature overheating, medium and low-temperature overheating, and normal operation[26]. Each operating condition contains DGA sample data of oil-immersed transformers featuring five fault gases: H_2 C_2H_2 C_2H_4 C_2H_6 CH_4, totaling 450 samples. The data is divided into training and testing sets in a ratio of 7:3, with the first 315 samples used as the training set and the last 135 samples used for testing the model. The six different operating conditions are numbered from 1 to 6, as shown in Table 5.

Table 5 Composition of fault sample data

Fault type	Number	Traning sets	Test sets	Total sample
High energy discharge	1	91	39	130
Low energy discharge	2	39	17	56
Partial discharge	3	58	25	83
High-temperature overheating	4	57	25	82
Medium and Low-temperature overheating	5	44	18	62
Normal operation	6	26	11	37
Combined diagnosis rate		315	135	450

To avoid the mixing and overlapping of fault information in the original data, KPCA is utilized to reduce the dimensionality of the fault features, resulting in a new fault dataset composed of 9

principal components. This new dataset is then input into the model for training and testing, and the fault diagnosis accuracy for each model is analyzed. The simulation experimental platform used in this study is a computer running the Windows 10 operating system with Matlab 2020b as the computational environment.

4.1 Comparison of Different Models with DHKELM Performance in Diagnosis

To validate the diagnostic accuracy of the DHKELM model, this study compares it with five other models: Probabilistic Neural Network (PNN), Random Forest (RF), Support Vector Machine (SVM), Extreme Learning Machine (ELM), and XGBoost[27,30]. The new fault dataset obtained from KPCA dimensionality reduction is input into the different models for classification training. Each model is run 30 times to obtain the average diagnostic accuracy for various fault types, as shown in Table 6.

Table 6 Comparison of average diagnostic accuracy of different diagnostic models

Fault type	Comparison of average diagnostic accuracy of different diagnostic models%					
	BP	RF	SVM	ELM	XGB-oost	DHKELM
High energy discharge	66.7	69.0	75.4	83.3	55.6	100
Low energy discharge	94.1	96.7	93.6	81.1	96.4	91.7
Partial discharge	37.5	60.0	56.3	58.8	89.5	93.3
High-temperature overheating	80.0	82.8	94.7	90.0	96.2	83.3
Medium and Low-temperature overheating	85.7	93.8	90.9	77.8	100	71.4
Normal operation	83.8	78.4	79.3	96.8	88.9	100
Combined diagnosis rate	77.48	81.45	80.79	83.44	86.65	89.74

As shown in Table 6, the diagnostic accuracy of the DHKELM model is significantly higher than that of the other models during high-energy discharge, partial discharge, and normal operating conditions. The overall diagnostic accuracy of the DHKELM model reached 89.743%, while the accuracies of the BP, RF, SVM, ELM, and XGBoost models were 77.48%, 81.45%, 80.79%, 83.44%, and 86.65%, respectively, all of which were lower than that of the DHKELM model. This indicates that the DHKELM model performs better in classification diagnostics compared to the other models.

4.2 IHBA-DHKELM Model Diagnosis

The IHBA-DHKELM model is used to diagnose the above samples, and its performance is compared with the HBA-DHKELM, SSA-DHKELM, and GWO-DHKELM models. The parameters for each model are shown in Table 7, while the comparison of the accuracy rates for different diagnostic methods is presented in Table 8.

Table 7 Related algorithm parameters

Algorithm	Parameters
SSA	ST=0.7;PD=0.4;SD=0.2
GWO	Alpha=Beta=Deta=0
HBA	Beta=6;C=2

Table 8 Comparison of fault diagnosis accuracy of different diagnosis models

Fault type	SSA-DHKELM	GWO-DHKELM	HBA-DHKELM	IHBA-DHKELM
High energy discharge	94.1%	100%	94.7%	100%
Low energy discharge	81.3%	76.5%	100%	95.7%
Partial discharge	86.7%	77.1%	88.2%	100%
High-temperature overheating	60.0%	95.7%	92.0%	93.9%
Medium and Low-temperature overheating	75.0%	40.0%	75.0%	95.0%
Normal operation	100%	100%	100%	100%
Combined diagnosis rate	82.96%	81.48%	91.11%	97.03%

From Table 8 and Figures 4 to 7, it can be observed that the IHBA-DHKELM model achieved the highest fault diagnosis accuracy, with both high-energy discharge, partial discharge, and normal conditions reaching 100%. The accuracy rates for low-energy discharge, high-temperature overheating, and medium-low temperature overheating were 95.7%, 93.9%, and 95.0%, respectively. The diagnosis accuracy for all six fault types was above 93%, resulting in a total fault diagnosis accuracy of 97.03%. This represents an improvement of 7.29% over the DHKELM classifier's diagnosis results and is 14.07%, 15.55%, and 5.92% higher than the SSA-DHKELM, GWO-DHKELM, and HBA-DHKELM models, respectively. This indicates that the proposed IHBA-DHKELM model has better diagnostic accuracy.

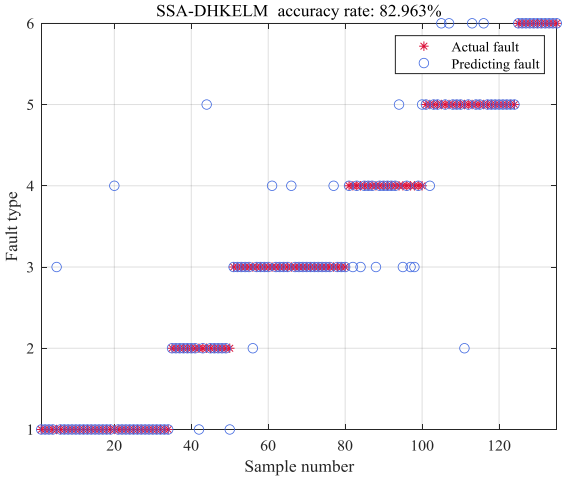


Fig. 4 SSA-DHKELM fault diagnosis classification

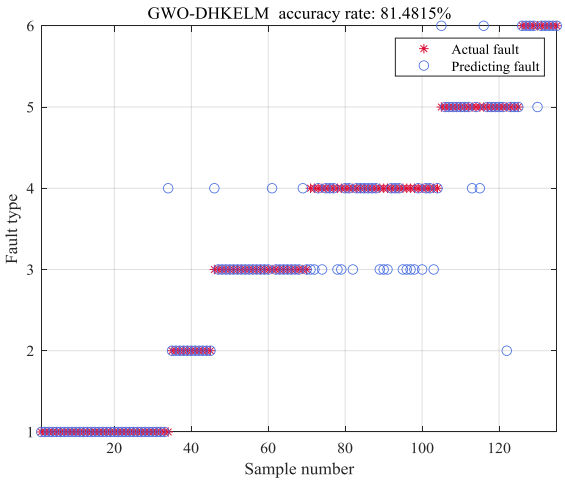


Fig. 5 GWO-DHKELM fault diagnosis classification

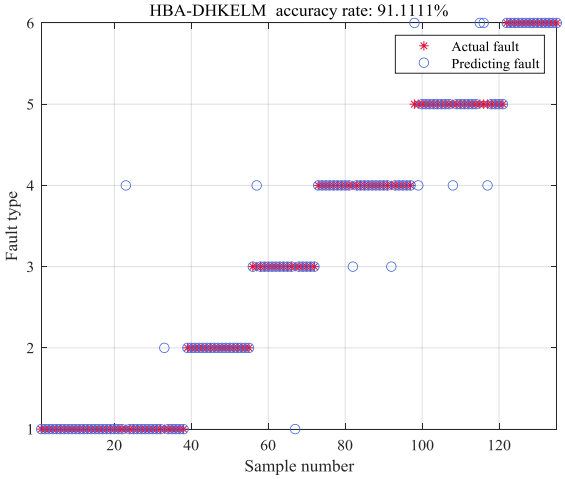


Fig. 6 HBA-DHKELM fault diagnosis classification

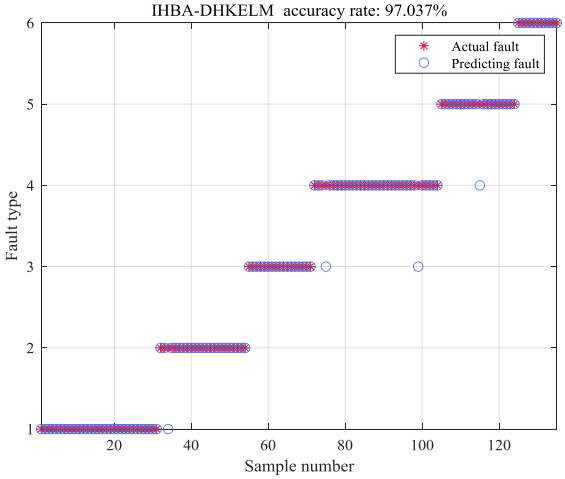


Fig. 7 IHBA-DHKELM fault diagnosis classification

5 Conclusion

(1) This paper utilizes KPCA to extract 9 input features with the highest principal component contribution rates from the 16-dimensional DGA gas concentration data, and compares the results with SVM, thereby improving both diagnosis accuracy and reducing diagnosis time.

(2) The diagnostic results of the DHKELM model compared with BP, RF, SVM, ELM, and XGBoost indicate that DHKELM exhibits better feature extraction capabilities and higher diagnostic accuracy.

(3) Optimization tests on the three benchmark functions from CEC 2017 for the IHBA, SSA, GWO, and HBA algorithms demonstrate that IHBA has superior optimization capabilities and effectively avoids getting trapped in local optima.

(4) Simulation results show that the IHBA-DHKELM model significantly optimizes the hyperparameters in DHKELM, leading to a marked improvement in diagnostic accuracy compared to SSA-DHKELM, GWO-DHKELM, and HBA-DHKELM, while exhibiting higher stability and generalization ability.

References

[1] Cao, Y.. Research on Transformer Fault Diagnosis Method Based on Improved Grey Wolf Algorithm Optimized BP Neural Network. Electrical Switch2024;62(02), 82-85..

[2] Y. Zhang, L. Zhao, J, "Forecasting of Dissolved Gases in Power Transformer Oil Based on DOG -LSSVM Regression and Artificial Bee Colony," 2018 International Conference on Power System Technology (POWERCON), Guangzhou, China, 2018, pp. 3620-3625.

[3] Yuwei Zhang. Transformer fault diagnosis based on Fuzzy rogers four ratio method. Electrotech Technol 2021;(12):89–92.

[4] Yi Liu, Yuanping Ni. Transformer fault diagnosis method based on three-ratio gray correlation analysis. High Volt Technol2002;28(10):16–7.

[5] Chengming Zhang, Jufang Xie, Song Yu, Chao Tang, Dong Hu. An improved fault diagnosis method for transformer Duval pentagon1 based on spatial analysis theory. High Volt Technol 2022;1–11

[6] Xiao, Y.; Pan, W.; Guo, X.; Bi, S.; Feng, D.; Lin, S. Fault Diagnosis of Traction Transformer Based on Bayesian Network. Energies 2020, 13, 4966.

[7] Z. Zhan, J. Chen, W. Chen, "Transformer Fault Diagnosis Method Based on Fuzzy Logic and D-S Evidence Theory," 2022 5th International Conference on Energy, Electrical and Power Engineering (CEEPE), Chongqing, China, 2022, pp. 470-475.

[8] Jinglong Jia, Tao Yu, Zijie Wu et al. 2017 Fault diagnosis method of transformer based on convolutional neural network[J] Electrical Measurement & Instrumentation 54 62-67

[9] Yang Fang Ming, Liu Chuan, Sun Yong et al. 2014 Fault Prediction Based on Dissolved Gas Concentration from Insulating Oil in Power Transformer Using Neural Network[J] 2789 312-317

[10] Meng W.D. 2020 Transformer Fault Diagnosis Based on BP Neural Network J. Communication power technology 37 84-86

[11] Yang X, Pang S, Shen W, et al. Aero engine fault diagnosis using an optimized extreme learning machine[J]. International Journal of Aerospace Engineering, 2016, 2016(1): 7892875.

[12] Hu, Beibei, and Yunhe Cheng. "Predicting regional carbon price in China based on multi-factor HKELM by combining secondary decomposition and ensemble learning." Plos one 18.12 (2023): e0285311.

[13] Wang W, Cui X, Qi Y, et al. Prediction model of coal seam gas content based on kernel principal component analysis and IDBO-DHKELM[J]. Measurement Science and Technology, 2024, 35(11): 115113.

[14] Guerbas, F., Benmahamed, Y., Teguar, Y. et al. Neural networks and particle swarm for transformer oil diagnosis by dissolved gas analysis. Sci Rep 14, 9271 (2024).

[15] H. Peng and C. Zhao, "Research on fault diagnosis of KPCA-WOA-BP transformer," 2024 5th International Conference on Computer Engineering and Application (ICCEA), Hangzhou, China, 2024, pp. 1487-1493.

[16] M. Zhang and W. Chen, "Fault Diagnosis of Power Transformer Based on SSA—MDS Pretreatment," in IEEE Access, vol. 10, pp. 92505-92515, 2022.

[17] Leifeng He and Ying Huang 2022 A transformer fault diagnosis method based on grey wolf optimization algorithm optimized support vector machine[J] Hongshuihe 41 84-88

[18] Guomin Xie and Jialiang Wang . "Transformer Fault Identification Method Based on Hybrid Sampling and IHBA-SVM." Journal of Electronic Measurement and Instrumentation, vol. 36, no. 12, 2022, pp. 77-85.

[19] Xuan Chen. "Research on Transformer Feature Selection Method Based on Grey Wolf Algorithm." Electrical Materials, vol. 2024, no. 04, 2024, pp. 90-92, 96. DOI: 10.16786/j.cnki.1671-8887.eem.2024.04.024.

[20] Xin Zheng and Chun Shi et al. "Fault Diagnosis Method for Coal Mine Transformers Based on ISSA-SVM." Electromechanical Engineering Technology, vol. 51, no. 07, 2022, pp. 31-34, 49

[21] Xiaoqin Zhang and Chunqiang Hu. "Data Acquisition and Monitoring System Attack Detection Model Based on Improved Extreme Learning Machine." Journal of Nanjing University of Aeronautics and Astronautics, 2021, pp. 708-717

[22] Liqun Shang,Yadong Hou, et al. "Transformer Fault Diagnosis Based on IDOA-DHKELM." High Voltage Engineering, vol. 49, no. 11, 2023, pp. 4726-4735. DOI: 10.13336/j.1003-6520.hve.20221483.

[23] Jing Yan, Xueying Zhang, et al. "Regression Prediction Model Combining Stack-based Supervised AE and Variable Weight ELM." Computer Engineering, vol. 48, no. 08, 2022, pp. 62-69, 76.

[24] Fatma A. Hashim, Essam H. Houssein,et al. Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems,Mathematics and Computers in Simulation,Volume 192,2022,Pages 84-110,ISSN 0378-4754

[25] Lu W, Shi C, Fu H, et al. Fault diagnosis method for power transformers based on improved golden jackal optimization algorithm and random configuration network[J]. IEEE Access, 2023, 11: 35336-35351.

[26] Y. Wu, Y. Zhang, X. Zhong and L. Cheng, "A Power Transformer Fault Diagnosis Method-Based Hybrid Improved Seagull Optimization Algorithm and Support Vector Machine," in IEEE Access, vol. 10, pp. 17268-17286, 2022.

[27] Zhang, X.; Sun, Z. Application of Improved PNN in Transformer Fault Diagnosis. Processes 2023, 11, 474.

[28] L. Kou, C. Liu, G. -w. Cai, Z. Zhang, X. -j. Li and Q. -d. Yuan, "Fault Diagnosis for Power Converters based on Random Forests and Feature Transformation," 2020 IEEE 9th International Power Electronics and Motion Control Conference (IPEMC2020-ECCE Asia), Nanjing, China, 2020, pp. 1821-1826, doi: 10.1109/IPEMC-ECCEAsia48364.2020.9367970.

[29] Pengfei Cao and Yongping Gan,. "Transformer Fault Diagnosis Based on Vector Weighting Algorithm Optimized ELM." Environmental Technology, vol. 41, no. 10, 2023, pp. 136-142.

[30] Jiang J, Liu Z, Wang P, et al. Improved Crow Search Algorithm and XGBoost for Transformer Fault Diagnosis[C]//Journal of Physics: Conference Series. IOP Publishing, 2023, 2666(1): 012040. [4] BRAND K P,KOPAINSKY J. Particle densities in a decaying SF6 plasm[J]. Applied Physics,1978,16(4):425-432.