



科研成果汇总

学号:	231081101001
研究生:	贾文超
导师:	安爱民教授
研究方向:	复杂系统建模与控制
论文题目:	数据驱动下的风机组件健康状态 评估与功率预测方法研究
学科:	控制理论与控制工程
学院:	自动化与电气工程学院
入学时间:	2023 年 9 月

目录

1.检索报告	1
2. Jia Wenchao , An Aimin. An efficient transfer fault diagnosis method integrating feature redundancy selection and multi-strategy parameter optimization.....	4
3. Jia Wenchao , An Aimin. Research on bearing remaining useful life anti-noise prediction based on fusion of color-grayscale time-frequency features.....	34
4. Jia Wenchao , An Aimin. An efficient lightweight gearbox fault diagnosis method based on parameter optimization and adaptive attention enhancement across working conditions.....	58
5. Jia Wenchao , An Aimin. A Multi-Variable Driven Dual-Stage Modal-Decoupling Framework Integrating Deterministic–Uncertainty Modeling for Wind Power Forecasting with Feature Interpretability Analysis.....	90



机构: 兰州理工大学

姓名: 贾文超 [231081101001]

著者要求对其在国内外学术出版物所发表的科技论著被以下数据库收录情况进行查证。

检索范围:

- 科学引文索引 (Science Citation Index Expanded): 1900年-2026年

检索结果:

检索类型	数据库	年份范围	总篇数	第一作者 篇数
SCI-E 收录	SCI-EXPANDED	2024 - 2026	4	



委托人声明:

本人委托兰州理工大学图书馆查询论著被指定检索工具收录情况, 经核对检索结果附件中所列文献均为本人论著, 特此声明。

作 者 (签字): 贾文超

完 成 人 (签字): 陈 超

完 成 日 期 : 2026年3月13日

完成单位 (盖章): 兰州理工大学图书馆信息咨询与学科服务部

(本检索报告仅限校内使用)



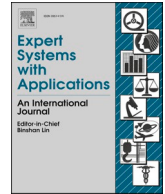


数据库：科学引文索引 (Science Citation Index Expanded) 时间范围：2024年至2026年		作者姓名：贾文超 作者单位：兰州理工大学		检索人员：陈 超 检索日期：2026年3月13日		
检索结果：被 SCI-E 收录文献 4 篇						
#	作者	地址	标题	来源出版物	文献类型	入藏号
1	Jia, WC; An, AM; Gong, B; Shi, YK; Yan, ZM	[Jia, Wenchao; An, Aimin; Gong, Bin; Yan, Zheming] Lanzhou Univ Technol, Coll Elect & Informat Engn, Lanzhou 730050, Peoples R China.; [Shi, Yaoke] Lanzhou Univ Technol, Sch Comp & Commun, Lanzhou 730050, Peoples R China.; [An, Aimin] Lanzhou Univ Technol, Key Lab Gansu Adv Control Ind Proc, Lanzhou 730050, Peoples R China.	A multi-variable driven dual-stage modal-decoupling framework integrating deterministic-uncertainty modeling for wind power forecasting with feature interpretability analysis	ENERGY 2026, 344: 139884.	J Article	WOS:0016 641552000 01
2	Jia, WC; An, AM; Gong, B; Shi, YK; Luo, ZP	[Jia, Wenchao; An, Aimin; Gong, Bin; Luo, Zhipeng] Lanzhou Univ Technol, Coll Elect & Informat Engn, Lanzhou 730050, Peoples R China.; [Shi, Yaoke] Lanzhou Univ Technol, Sch Comp & Commun, Lanzhou 730050, Peoples R China.; [An, Aimin] Lanzhou Univ Technol, Key Lab Gansu Adv Control Ind Proc, Lanzhou 730050, Peoples R China.	An efficient lightweight gearbox fault diagnosis method based on parameter optimization and adaptive attention enhancement across working conditions	MEASUREMENT SCIENCE AND TECHNOLOGY 2025, 36 (5): 056109.	J Article	WOS:0014 733421000 01
3	Jia, WC; An, AM; Gong, B; Shi, YK	[Jia, Wenchao; An, Aimin; Gong, Bin] Lanzhou Univ Technol, Coll Elect & Informat Engn, Lanzhou 730050, Peoples R China.; [Shi, Yaoke] Lanzhou Univ Technol, Sch Comp & Commun, Lanzhou 730050, Peoples R China.; [An, Aimin] Lanzhou Univ Technol, Key Lab Gansu Adv Control Ind Proc, Lanzhou 730050, Peoples R China.	An efficient transfer fault diagnosis method integrating feature redundancy selection and multi-strategy parameter optimization	EXPERT SYSTEMS WITH APPLICATIONS 2025, 279: 127267.	J Article	WOS:0014 650563000 01
4	Jia, WC; An, AM; Du, XJ; Shi, YK; Gong, B	[Jia, Wenchao; An, Aimin; Du, Xianjun; Shi,	Research on bearing remaining useful life anti-	MEASUREMENT SCIENCE AND	J Article	WOS:0012 488709000



	Yaokc; Gong, Bin] Lanzhou Univ Technol, Coll Elect & Informat Engn, Lanzhou 730050, Gansu, Peoples R China.	noise prediction based on fusion of color-grayscale time-frequency features	<i>TECHNOLOGY</i> 2024, 35 (9): 096102.		01
合计					4





An efficient transfer fault diagnosis method integrating feature redundancy selection and multi-strategy parameter optimization

Wenchao Jia^a, Aimin An^{a,c,*}, Bin Gong^a, Yaoke Shi^b

^a College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, China

^b School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China

^c Key Laboratory of Gansu Advanced Control for Industrial Processes, Lanzhou University of Technology, Lanzhou 730050, China

ARTICLE INFO

Keywords:

Feature redundancy
Computational resources
Data scarcity
Feature selection
Parameter optimization
Transfer learning

ABSTRACT

Deep learning has shown substantial progress in fault diagnosis in recent years. Nevertheless, its application is hindered by excessive computational requirements and challenges in addressing data scarcity and multi-condition adaptability in practical settings. Notably, the problem of feature redundancy stands out, as it not only inflates computational demands but also obscures vital features, reducing the generalizability and precision of diagnostic models. To address these challenges, this study proposes an efficient transfer fault diagnosis algorithm based on a multi-scale redundant feature selection network enhanced by parameter optimization. First, the proposed algorithm combines short-time Fourier transform (STFT) and synchrosqueezing wavelet transform (SWT) to comprehensively extract deep-level features from the data. Secondly, a channel and spatial redundancy feature selection unit is employed to effectively reduce channel and spatial redundancy in the features, thereby focusing on the learning of highly representative features. Furthermore, a transformer feature extraction (TFE) module is integrated at the backend of the model to accurately capture global contextual information and detailed features in the data. Finally, a multi-strategy improved Harris Hawk optimization algorithm (MS-HHO) was designed to achieve efficient hyperparameter tuning and global optimal configuration. Experimental results demonstrate that the proposed model achieved peak diagnostic accuracies of 98.63%, 98.83%, and 98.9% on three validation datasets using only 20% of the samples. Even under severe noise conditions, the model maintained an average diagnostic accuracy of 73.75%. Additionally, it exhibited superior diagnostic performance and robustness compared to competing models in diverse transfer tasks.

1. Introduction

Bearings play a crucial role in wind turbines, primarily supporting and carrying the rotating components, such as the main shaft and gearbox, ensuring their smooth operation. The health condition of the bearings directly impacts the turbine's performance. Bearing failures can cause vibrations, equipment wear, or reduced power generation efficiency, ultimately affecting the safety, stability, and reliability of the entire wind turbine. Diagnosing faults in the bearings, a critical component of wind turbines, not only significantly improves equipment efficiency and extends service life but also reduces maintenance costs and minimizes potential safety risks (Table 1).

Fault diagnosis methods for rolling bearings have evolved from approaches based on expert knowledge and mathematical/physical models to data-driven techniques. Expert knowledge-based methods rely on

established knowledge bases and expert experience to infer and analyze equipment faults, thereby identifying them (Wan et al., 2024). On the other hand, mathematical/physical model-based methods establish mathematical and physical models by analyzing the operational mechanisms and processes of the equipment. These models are then validated through experiments and data to achieve fault diagnosis (Yang et al., 2024). Although these traditional methods are effective for diagnosing simple and fixed types of faults, they require extensive specialized knowledge and lack scientific rigor. Additionally, their flexibility is limited when applied to different equipment and systems.

With the rapid progress in data acquisition and analysis technologies in recent years, data-driven approaches have become a major focus in fault diagnosis research. These methods are characterized by their robust data mining potential and reduced dependency on specialized domain knowledge. The key categories of data-driven methods include

* Corresponding author.

E-mail address: anaimin@lut.edu.cn (A. An).

<https://doi.org/10.1016/j.eswa.2025.127267>

Received 31 December 2024; Received in revised form 18 February 2025; Accepted 12 March 2025

Available online 22 March 2025

0957-4174/© 2025 Elsevier Ltd. All rights reserved, including those for text and data mining, AI training, and similar technologies.

Table 1
Overview of the current state of data-driven fault diagnosis methods.

Data-Driven Fault Diagnosis Methods		Advantage	Disadvantage	Representative study
Signal Analysis-Based Methods	Wavelet transform	• Feature extraction is easy; • Highly interpretable.	• Sensitivity to noise; • Feature extraction relies on manual experience; • Unable to learn automatically.	(Zhang et al., 2022)
	Short-time Fourier transform			(Wei et al., 2022)
Machine Learning-Based Methods	Empirical mode decomposition	• Ease of modeling; • Capable of handling some high-dimensional data; • Fast training.	• Model performance is heavily dependent on the extracted features; • Limited non-linear expressiveness; • Weak model generalization performance.	(Wang et al., 2020)
	Variational modal decomposition			(Ni et al., 2022)
	Artificial neural network			(Liang et al., 2023)
	K-Nearest Neighbors			(Lu et al., 2021)
	Support vector machine			(Wang et al., 2021)
Deep Learning-Based Methods	Random forest	• Automatic feature extraction; • Strong generalization performance; • Enables end-to-end learning; • Strong model nonlinear characterization.	• High demand for computing resources; • High data quality and quantity requirements; • Poor model interpretability.	(Buchaiah and Shakya, 2022)
	Naive Bayes			(Kumar and Anand, 2024)
	Convolutional neural network			(Wang et al., 2023)
	Recurrent neural network			(Jin et al., 2022)
	Autoencoder			(He et al., 2021)
	Generative Adversarial Network			(Shao et al., 2023)
	Transformer			(Zhang et al., 2023)

signal analysis, machine learning, and deep learning. Signal analysis approaches, including wavelet transform (WT), short-time Fourier transform (STFT), empirical mode decomposition (EMD), and variational modal decomposition (VMD), have shown promising results in the early detection of sensitive fault states by decomposing and filtering distinct components of vibration signals. However, their reliance on the assumption of signal linearity poses challenges. Under variable-speed and variable-load conditions, the time-varying, nonlinear, and coupled characteristics of wind turbine-bearing vibration signals limit their utility in fault diagnosis for complex systems. Machine learning models, such as artificial neural networks (ANN) (Kumbhar et al., 2021), k-nearest neighbors (KNN) (Li et al., 2024), and support vector machines (SVM) (Shao et al., 2021), achieve diagnostic tasks by uncovering the relationships between features and fault patterns. While shallow machine learning methods reduce reliance on manual expertise, they exhibit certain limitations. First, their diagnostic accuracy heavily depends on the construction of the feature set. Second, their nonlinear representation capabilities are constrained in scenarios involving weak vibration signals or complex coupling of fault features, making it challenging to fully capture the intricate relationships between fault characteristics and categories.

Compared to traditional methods, deep learning can automatically extract high-level features from complex sensor data through large-scale data-driven model training, without the need for manually designed feature engineering. This characteristic enables deep learning to exhibit exceptional diagnostic performance in fault detection of critical wind turbine components such as bearings, gearboxes, and motors (Yan et al., 2021; Zhang et al., 2022). For instance, convolutional neural networks (CNN) excel in processing vibration signals and time-frequency images, while long short-term memory networks (LSTM) are particularly advantageous in capturing the dynamic characteristics of time-series data. Transformer networks, on the other hand, offer significant advantages in capturing long-range dependencies and performing large-scale parallel computations. The application of these technologies has notably enhanced diagnostic accuracy and robustness. Shan et al. (2023) proposed a bearing fault identification method based on Mel-frequency cepstral coefficients and CNN, optimizing the CNN structure to achieve efficient feature extraction and accurate classification in complex environments. Qin et al. (2023) combined LSTM networks with a multi-channel attention mechanism to propose a fault diagnosis method for wind turbine pitch systems, significantly improving diagnostic accuracy under complex conditions. Liu et al.

(2023) integrated CNN with Transformer models, leveraging the CNN's ability to extract local features and the Transformer's capacity to capture global context, thus enhancing fault diagnosis accuracy. Although deep learning-based fault diagnosis methods can automatically extract features from large-scale data, they still face several challenges, including high computational demands and limited generalization ability. (1) During feature extraction, models may learn a substantial number of redundant features, leading to unnecessary computational costs and reduced diagnostic efficiency. (2) Model training requires a large number of labeled samples and assumes that the training and testing data share the same distribution. (3) These models are typically designed for specific operating conditions, resulting in poor generalization performance and complex training procedures. (4) In real-world industrial applications, variations in data distributions across different operating conditions or equipment may lead to a decline in diagnostic accuracy. As a result, the practical applicability and stability of these methods remain constrained in real-world scenarios.

In recent years, feature selection methods have been extensively studied for their effectiveness in enhancing model performance and reducing computational burden (Zhang et al., 2021; Zhao et al., 2023). The primary objective of feature selection is to eliminate redundant and irrelevant features while retaining those most discriminative for classification tasks, thereby improving computational efficiency and decision-making capabilities. Zhang et al. (2021) proposed a feature selection method called conditional-weight joint relevance feature selection (CWJR-FS), which enhances selection discrimination by jointly considering the classification information of candidate and selected features, demonstrating superior performance in multi-label classification tasks. Similarly, Gao et al. (2020) introduced the feature redundancy maximization (FRM) method, which leverages the summation of conditional mutual information and the maximization of the minimum criterion to effectively reduce redundant features and improve selection accuracy. Furthermore, Gao et al. (2021) proposed an information-theoretic feature selection approach that preserves sample similarity to mitigate the impact of outliers and irrelevant features, thereby increasing selection efficiency. Despite their effectiveness in multi-label learning and traditional classification tasks, these methods primarily focus on static data, and their applicability to bearing fault diagnosis remains under-explored. Some studies have employed attention mechanisms for feature selection, dynamically adjusting feature weights during model training to enable adaptive selection (Lv et al., 2022; Liao et al., 2023; Cui et al., 2024). This approach assigns higher weights to key features while

Table 2

Current research status of fault diagnosis methods based on transfer learning.

Transfer method	Advantage	Disadvantage	Representative study
Instance-based transfer learning	- High efficiency ; - Easy to implement ; - Reduces data demand.	- Difficulty in instance selection ; - Challenges in handling domain discrepancies ; - High risk of overfitting.	(Tang et al., 2023)
Model-based transfer learning	- Good integrity in transfer ; - Strong versatility ; - Efficient use of data.	- Complex model structure ; - Difficulty in parameter tuning ; - High computational resource demand.	(Dong et al., 2022)
Feature-based transfer learning	- Enhanced feature representation ; - Low data demand ; - Good model robustness.	- Challenges in feature matching ; - Risk of data bias ; - High computational resource demand.	(An et al., 2023)

suppressing redundant ones, thereby improving model robustness. However, most existing methods have been developed for natural language processing (NLP) and image classification tasks, with limited applications in wind turbine-bearing fault diagnosis. Moreover, they have yet to fully integrate transfer learning strategies to optimize feature selection for cross-operating conditions.

In recent years, transfer learning has gained significant attention in the field of intelligent fault diagnosis (Chen et al., 2023), particularly as an effective approach for enhancing model generalization under conditions of cross-operating-condition data distribution shifts and data scarcity (Han and Feng, 2023). The current research status of fault diagnosis strategies based on transfer learning is summarized in Table 2. Some studies have adopted adversarial transfer learning frameworks, achieving significant progress in enhancing cross-operating-condition adaptability. Jiang et al. (2023) proposed the dual domain adversarial network (DDAN), which employs multi-channel parallel feature extraction to capture domain-invariant features and utilizes the L1,2-norm based Wasserstein discrepancy (L1,2 WD) for domain alignment, thereby improving the stability and computational efficiency of the diagnostic model. Building on this, Wu et al. (2023) introduced the conditional distribution-guided adversarial transfer learning network (CDGATLN), which reduces distribution discrepancies between multiple source domains and the target domain through a conditional distribution alignment strategy. Additionally, it incorporates a monotonic importance constraint for adaptive knowledge transfer, thereby enhancing the adaptability of cross-operating-condition fault diagnosis. However, these studies do not consider the potential introduction of redundant features during feature extraction, which may lead to unnecessary computational resource consumption and degraded model performance. Recently, researchers have increasingly combined attention mechanisms with transfer learning to enhance feature extraction and cross-domain adaptability in fault diagnosis (Li et al., 2023; Zhang et al., 2024; Zhong et al., 2022; Zhong et al., 2023). Shao and Kim (2024) proposed the adaptive multi-scale attention convolutional neural network (AmaCNN), integrating multi-scale feature fusion (MSFFCNN) and a multi-level attention scheme (MLAS) to improve cross-domain feature learning robustness. It further employs correlation alignment (CORAL) and semantic alignment loss (SA Loss) for domain adaptation, demonstrating superior performance in small-sample cross-domain fault diagnosis. Liu et al. (2024) developed the interpretable domain adaptation transformer (IDAT), utilizing a multi-layer Transformer framework to extract cross-domain modulation information while applying ensemble attention weights to improve interpretability. Experimental results show that IDAT excels in various cross-operating-condition and

Table 3

HUST dataset working condition numbers.

Condition	20 Hz	25 Hz	40 Hz	60 Hz	70 Hz	80 Hz
Encoding	U _A	U _B	U _C	U _D	U _E	U _F

Table 4

Description of DIRG bearing numbers and their states.

Name	Defect	Dimension (μm)
0A	NO DEFECT	—
1A	Diameter of an indentation on the inner ring	450
2A	Diameter of an indentation on the inner ring	250
3A	Diameter of an indentation on the inner ring	150
4A	Diameter of an indentation on a roller	450
5A	Diameter of an indentation on a roller	250
6A	Diameter of an indentation on a roller	150

cross-machine bearing fault diagnosis tasks. Wang and Zhao (2023) introduced the inter-domain attention mechanism, which generates source sample attention vectors to dynamically learn shared features between source and target domains. They further proposed an Inter-Domain Attention Encoder to enhance the domain adversarial neural network (DANN) for improved transferability. Experiments on the aero-engine and CWRU bearing datasets reveal an average accuracy increase of 1.05 % over conventional methods in certain unsupervised cross-domain tasks (Table 3).

Despite advancements in transfer learning for cross-condition fault diagnosis and data scarcity, several limitations persist: (1) Feature redundancy remains an issue—current models often extract redundant or irrelevant features, increasing computational overhead while reducing generalization capability. In multi-condition scenarios, this problem can significantly (Table 4) impact diagnostic accuracy. (2) Manual hyperparameter tuning is inefficient—most methods do not fully exploit optimization algorithms for hyperparameter adjustment, restricting adaptability and leading to high computational costs (Shen et al., 2024). Although techniques like grid search (Huang et al., 2024), Bayesian optimization (Wang et al., 2024), and competitive swarm optimizers (Zhao et al., 2024) have been applied to fault diagnosis models, they still face challenges such as large search spaces, high computation costs, and local optima. These factors hinder model stability and real-world deployment. Thus, improving diagnostic accuracy while reducing computational demands remains a critical challenge in transfer learning-based fault diagnosis.

To overcome these limitations, we propose an efficient fault diagnosis method that combines parameter optimization and redundant feature filtering, aiming to reduce computational demands while preserving or even improving diagnostic accuracy. To address channel and spatial redundancy in feature extraction, we develop a multi-scale redundant feature filtering network, which consists of channel redundancy filtering and spatial redundancy filtering units. This design effectively removes irrelevant and redundant information, focusing on extracting key diagnostic features to enhance model generalization and accuracy. Additionally, we incorporate a Transformer-based feature extraction (TFE) module, leveraging its strong global context modeling capability to capture both detailed and global information from the data. To further optimize the model, we introduce a multi-strategy improved Harris hawks optimization algorithm, which efficiently explores the hyperparameter search space to achieve optimal parameter selection. To improve diagnostic robustness across different operational conditions, we integrate an unsupervised domain adaptation framework, enhancing adaptability in cross-condition fault diagnosis. This study achieves high diagnostic performance with reduced computational costs, providing a practical and effective solution for fault diagnosis in complex industrial settings. The main contributions are summarized as follows:

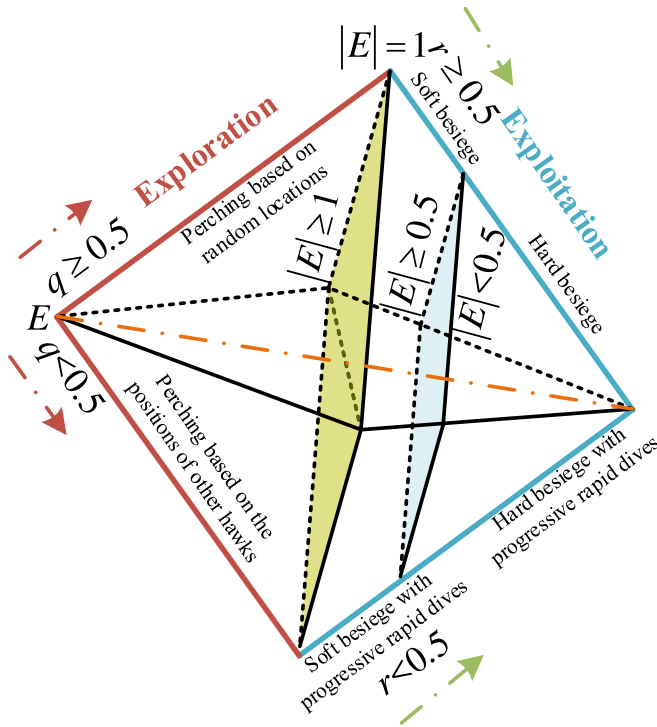


Fig. 1. Different Stages of the HHO Algorithm.

- (1) We propose an enhanced STFT-SWT feature extraction method to improve the quality of feature representation, facilitating more effective fault diagnosis.
- (2) An improved MS-HHO algorithm is introduced to enhance hyperparameter search efficiency, achieving an optimal trade-off between computational cost and model performance.
- (3) A lightweight feature redundancy filtering mechanism is designed to eliminate irrelevant information, thereby improving diagnostic accuracy and enhancing the model's generalization capability.
- (4) A Transformer-based feature extraction strategy is integrated to optimize global information capture, improving the model's adaptability to complex working conditions.

This paper is structured into four sections. Section 1 provides an overview of the current state of fault diagnosis technologies for wind turbine bearings, emphasizing their limitations. Section 2 discusses the theoretical foundations and model design applied in this study. Section 3 centers on experimental validation, detailing the design, simulation, and analysis of the experiments conducted. Finally, Section 4 summarizes the findings and outlines potential avenues for future research.

2. Optimization strategies and model architecture of the proposed method

2.1. Harris hawks optimizer

The HHO algorithm the cooperative hunting behavior of Harris hawks to solve optimization problems (Heidari et al., 2019). Its operation consists of two core phases: exploration, representing global search capabilities, and exploitation, corresponding to local refinement, as depicted in Fig. 1. A critical parameter, the prey escape energy factor E , governs the transition between these phases. The mathematical formulation of E is defined as follows:

$$\begin{cases} E = E_0 \times E_1 \\ E_1 = 2\left(1 - \frac{t}{T}\right) \end{cases} \quad (1)$$

where t and T denote the current iteration count and the maximum number of iterations, respectively. E_1 represents the energy control factor, while E_0 corresponding to the initial energy.

The pseudocode of the HHO algorithm is shown in Algorithm 1.

Algorithm 1 Pseudo-code of HHO algorithm

Inputs: The population size N and maximum number of iterations T .

Outputs: The location of the rabbit and its fitness value Initialize the random population X_i ($i = 1, 2, \dots, N$).

while (stopping condition is not met) **do**

Calculate the fitness values of hawks

Set X_{rabbit} as the location of rabbit (best location)

for (each hawk (X_i)) **do**

Update the initial energy E_0 and jump strength J using Eq:

$E_0 = 2rand - 1$; $J = 2(1 - rand)$

Update the E using Eq: $\begin{cases} E = E_0 \times E_1 \\ E_1 = 2\left(1 - \frac{t}{T}\right) \end{cases}$

(Exploration phase)

if ($|E| \geq 1$) **then**

Update the location vector using Eq:

$X(t+1) = \begin{cases} X_{rand}(t) - r_1 |X_{rand}(t) - 2r_1 X(t)| & q \geq 0.5 \\ (X_{rabbit}(t) - X_m(t)) - r_3 (LB + r_4 (UB - LB)) & q < 0.5 \end{cases}$

(Exploitation phase)

if ($|E| < 1$) **then**

Soft besiege

if ($r \geq 0.5$ and $|E| \geq 0.5$) **then**

Update the location vector using Eq:

$X(t+1) = X_{rabbit}(t) - X(t) - E|JX_{rabbit}(t) - X(t)|$

Hard besiege

else if ($r \geq 0.5$ and $|E| < 0.5$) **then**

Update the location vector using Eq:

$X(t+1) = X_{rabbit}(t) - E|X_{rabbit}(t) - X(t)|$

Soft besiege with progressive rapid dives

else if ($r < 0.5$ and $|E| \geq 0.5$) **then**

Update the location vector using Eq:

$Y = X_{rabbit}(t) - E|JX_{rabbit}(t) - X(t)|$

$Z = Y + S \times Levy(D)$

$X(t+1) = \begin{cases} Y & F(Y) < F(X(t)) \\ Z & F(Z) < F(X(t)) \end{cases}$

Hard besiege with progressive rapid dives

else if ($r < 0.5$ and $|E| < 0.5$) **then**

Update the location vector using Eq:

$Y = X_{rabbit}(t) - E|JX_{rabbit}(t) - X_m(t)|$

$Z = Y + S \times Levy(D)$

$X(t+1) = \begin{cases} Y & F(Y) < F(X(t)) \\ Z & F(Z) < F(X(t)) \end{cases}$

Return X_{rabbit}

2.2. Harris hawks optimizer with multi-strategy enhancements

The traditional HHO faces challenges such as premature convergence to local optima, low convergence accuracy, and slow convergence speed. To overcome these issues, this paper proposes several strategies to enhance the algorithm. To enhance the diversity of the initial population, a chaotic reverse learning strategy is employed to initialize the Harris Hawks population. Next, an adaptive dynamic exploration strategy is introduced to improve the global search performance of the algorithm. Finally, a time-varying Cauchy-enhanced hard trapping strategy is employed to enhance the algorithm's ability to escape local optima.

2.2.1. Chaos reverse learning strategy

The distribution of the initial population in the search space is crucial for the algorithm's efficiency and accuracy. Random initialization can lead to uneven population distribution, resulting in slow convergence in the early stages. To ensure a more even distribution, this paper combines chaotic mapping and reverse learning strategies for generating the initial population, with a greedy strategy used to select the best

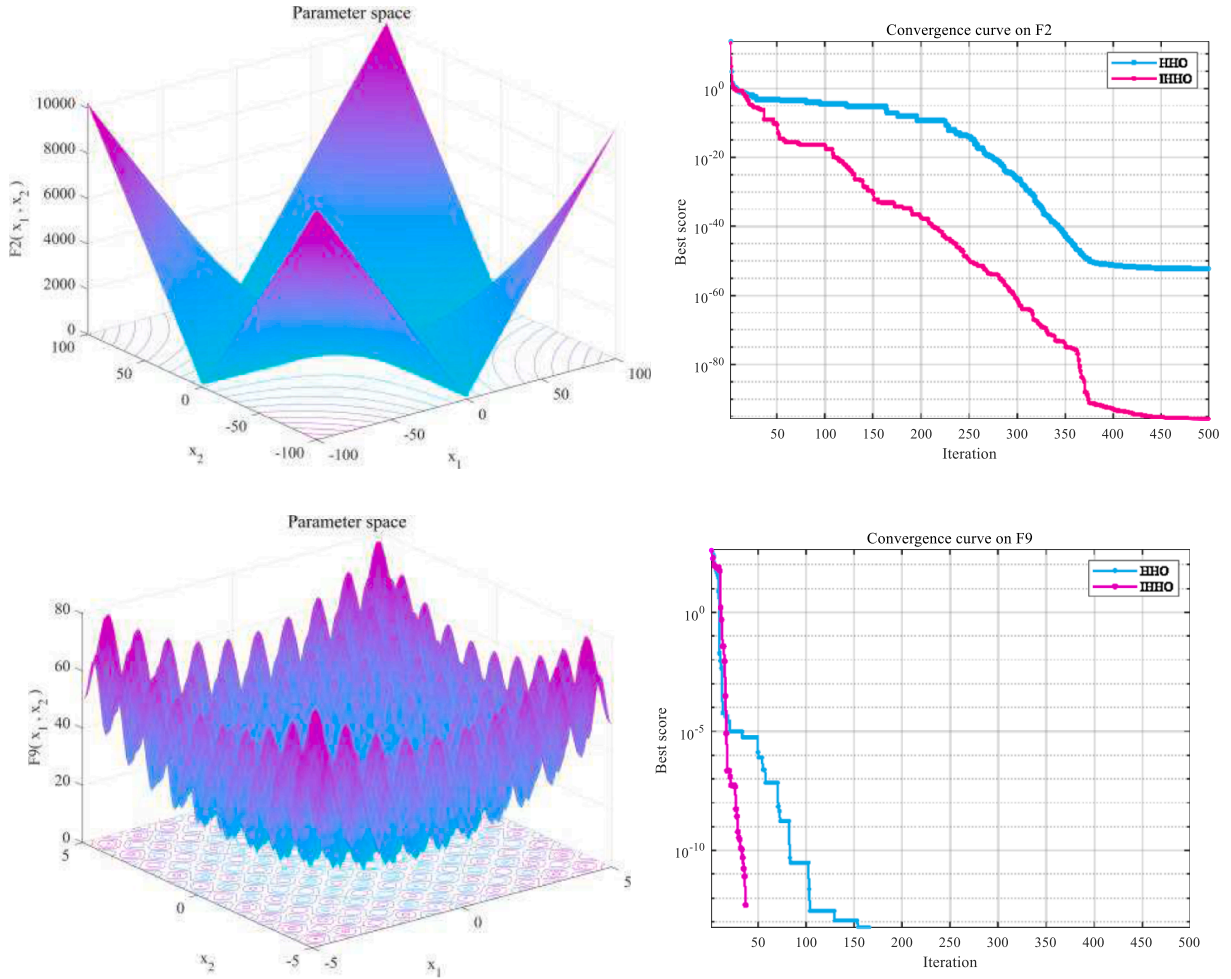


Fig. 2. Validation results of the effectiveness of the chaotic inverse learning strategy.

individuals to improve solution quality. The chaotic sequence is generated using the Tent chaotic mapping method, described as follows:

$$X_{k+1} = \begin{cases} X_k/u & 0 \leq X_k \leq u \\ (1 - X_k)/(1 - u) & u < X_k \leq 1 \end{cases} \quad (2)$$

The reverse learning strategy based on Tent chaotic mapping is described as follows:

$$X_{OBL} = X_{in} + (LB + UB - X_{in})X_{k+1} \quad (3)$$

The new solutions are generated by combining the reverse solution and the initial solution, as described by the following formula:

$$X_{new} = X_{OBL} \cup X_{in} \quad (4)$$

2.2.2. Adaptive dynamic exploration strategy

In the Harris Hawks optimization process, the update strategy for the initial Hawk population relies on the positions of other random individuals, without effectively integrating the global optimum or population center information. This limited approach increases the risk of premature convergence to local optima. To address this issue, this paper introduces an adaptive dynamic exploration strategy that employs a nonlinear weight factor ω to adjust the dependency of new individuals on their current states in the next iteration. In the early exploration phase, a smaller ω is used to reduce the influence of historical solutions, enhancing the algorithm's global search capability. In the later phase, a larger ω is employed to strengthen the dependency on historical solutions, accelerating convergence. Additionally, a sine operator is employed to enhance the diversity of hawk positions, while a nonlinear

step factor r'_1 balances global and local search capabilities during the exploration phase. The improved exploration phase equation is as follows:

$$X(t+1) = \begin{cases} \omega X(t) + r'_1 \sin r'_2 |r'_3 X_{rabbit}(t) - X(t)| & q \geq 0.5 \\ (X_{rabbit}(t) - X_m(t)) - r_3(LB + r_4(UB - LB)) & q < 0.5 \end{cases} \quad (5)$$

where r'_2 and r'_3 are random numbers within the interval $[0, 2\pi]$. r'_2 determines the hawk's movement distance, while r'_3 influencing the impact of the optimal solution on the hawk's next position. The weight factor ω and step factor r'_1 are calculated as follows:

$$\omega = \frac{e^{t/T} - 1}{e - 1} \quad (6)$$

$$r'_1 = \left(1 - \left(\frac{t}{T}\right)^\eta\right)^{1/\eta} \quad (7)$$

where $\eta \geq 1$ serves as the adjustment coefficient.

2.2.3. Time-varying Cauchy-enhanced hard besiege strategy

In the HHO algorithm, when prey escape energy is low, hawks adopt a hard besiege strategy to directly attack the prey. However, this approach depends heavily on the quality of the current optimal solution. Poor quality of the current optimum can degrade new solutions, slowing convergence and reducing precision. Additionally, the rigid nature of this strategy limits global exploration. To address these issues, a time-

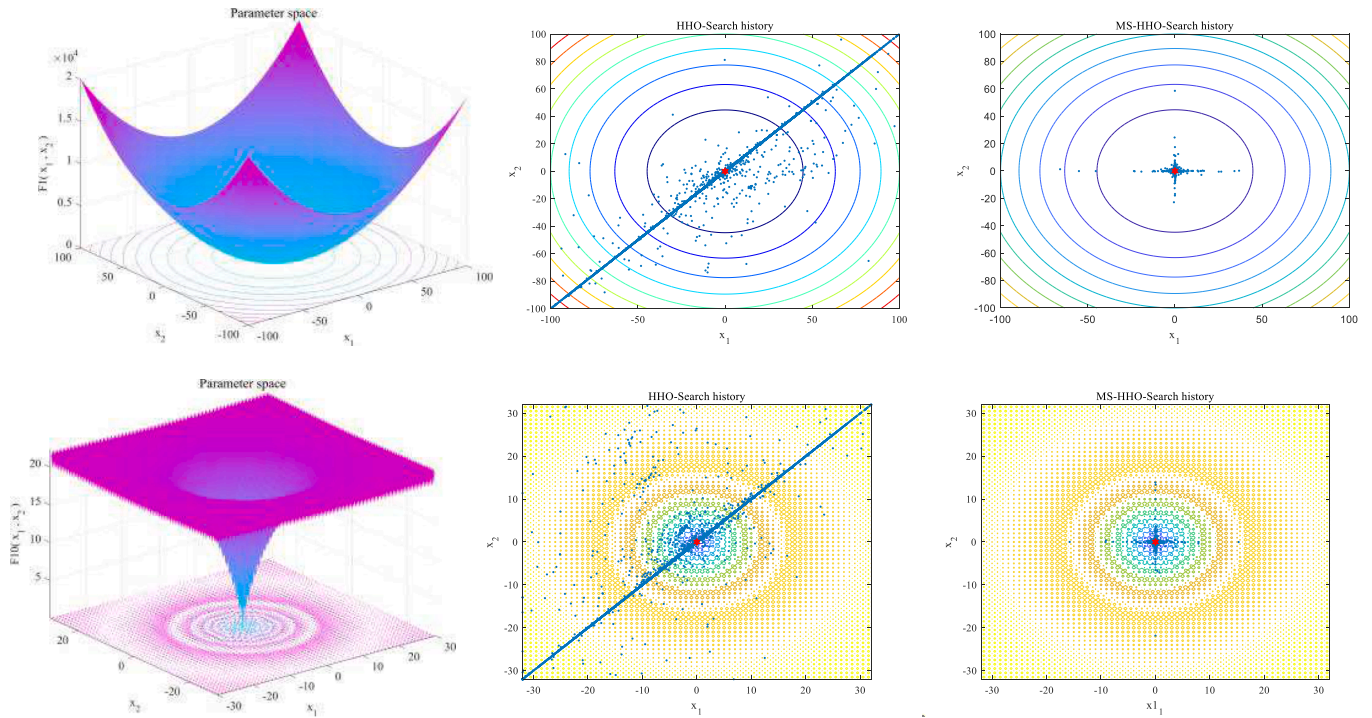


Fig. 3. Validation results of the effectiveness of the adaptive dynamic exploration strategy.

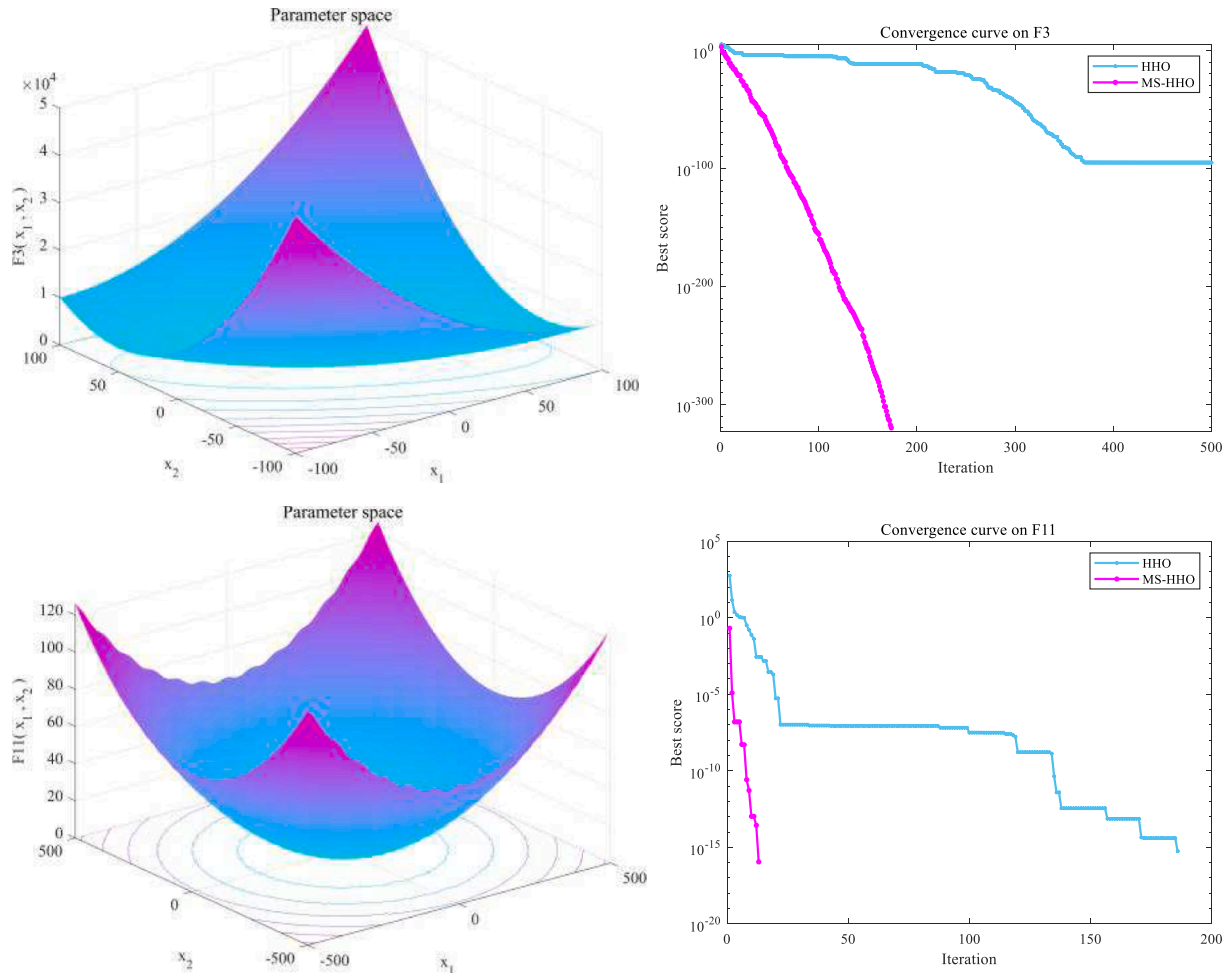


Fig. 4. Validation results of the effectiveness of the time-varying Cauchy-enhanced hard besiege strategy.

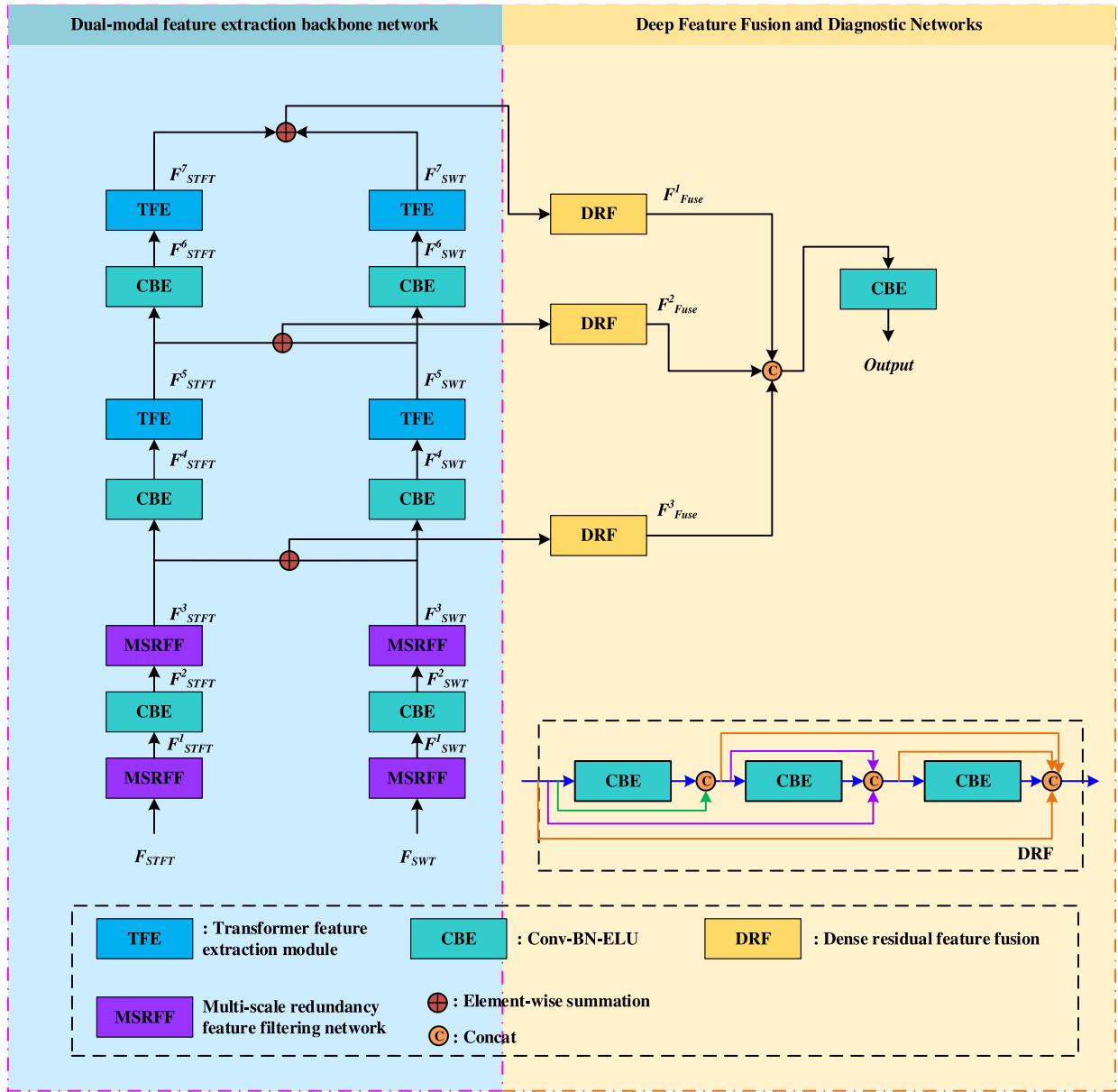


Fig. 5. Overall network structure diagram.

varying Cauchy perturbation is introduced to enhance global search capabilities and effectiveness during the hard besiege phase. The heavy-tailed property of the Cauchy distribution expands the search range, while incorporating time-dependent scale parameters b adjusts the “tail thickness” dynamically, improving diversity and flexibility. The mathematical formulation of this strategy is as follows:

$$X(t+1) = X_{rabbit}(t) - QE|X_{rabbit}(t) - X(t)| \quad (8)$$

$$Q = a + b \tan(\pi(p - 1/2)) \quad (9)$$

$$b = \frac{e^{(-0.1t/T)} - e^{(-0.1)}}{1 - e^{(-0.1)}} \quad (11)$$

where a is the location parameter, and p is a random vector following a normal distribution.

2.2.4. Ablation experiment for the improved strategies

To validate the effectiveness of the proposed improvement strategies, the standard test functions from reference (Takashima et al., 2010)

are used to evaluate the algorithm’s performance.

- (1) Validation of the effectiveness of the chaotic inverse learning strategy

The chaotic inverse learning strategy is used to improve the quality of the algorithm’s initial solution. In the experiment, the population size is set to 100, the maximum number of iterations is 500, and the optimization dimension is set to 30.

As shown in Fig. 2, the use of the chaotic inverse learning strategy to initialize the population improves the algorithm’s convergence speed and accuracy on the unimodal test function F2 and the multimodal test function F9. Although the improvement is not highly significant, it lays the foundation for further enhancement of the algorithm’s performance when combined with other strategies in later stages.

- (2) Validation of the effectiveness of the adaptive dynamic exploration strategy

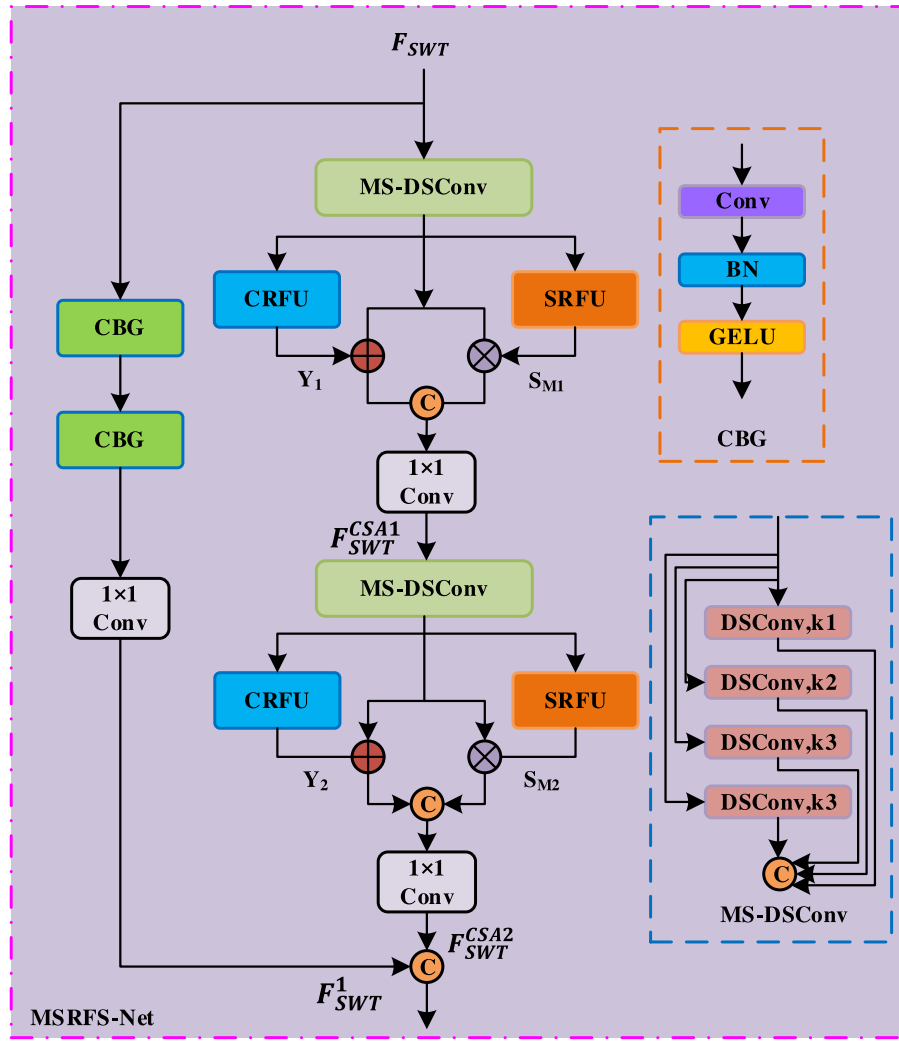


Fig. 6. The structure of MSRFS-Net.

Fig. 3 presents the comparison results before and after the improvement of the adaptive dynamic exploration strategy in the exploration phase. In the single-peak test function F1 and multi-peak test function F10, before the improvement, the update of new individuals was strongly dependent on the positions of previous individuals, causing the algorithm to have an unclear search objective, with some individuals wandering in the search space. After the improvement, the search paths of the population are more evenly distributed, and the points in the population cover the target area effectively, concentrating significantly on the optimal solution. These results demonstrate that the improved algorithm enhances the global exploration ability of the population and improves the convergence performance, enabling the algorithm to more precisely converge to the global optimal solution.

(3) Validation of the effectiveness of the time-varying Cauchy-enhanced hard besiege strategy

As shown in Fig. 4, the algorithm enhanced with the time-varying Cauchy-enhanced hard encirclement strategy exhibits a marked improvement in convergence performance over the baseline algorithm on multi-peak test function F11 and single-peak test function F3. For F11, the modified algorithm converges to a lower objective value with fewer iterations, maintaining stability and showcasing enhanced global search capabilities and an ability to escape local optima. In contrast, the unmodified algorithm converges more slowly. For F3, the improved

algorithm achieves the optimal solution in fewer iterations, significantly accelerating both convergence speed and accuracy, while the baseline algorithm suffers from slow convergence and inadequate exploration capabilities. Overall, the modified algorithm demonstrates high efficiency and robustness, excelling in both complex multi-peak and single-peak optimization tasks.

2.2.5. Time complexity analysis of the improved algorithm

Assume the population size of the HHO algorithm is N , the problem dimension is D , and the maximum number of iterations is T . The time complexity of the original HHO algorithm can be divided into three major components: initialization, fitness evaluation, and individual position update. The time complexity of the HHO algorithm is $O(N) + O(T \times N) + O(T \times N \times D)$. The computational complexity of the MS-HHO algorithm is mainly influenced by the chaotic reverse learning strategy, adaptive dynamic exploration strategy, and time-varying Cauchy-enhanced hard trapping strategy. The computational complexity of the chaotic reverse learning strategy consists of three main parts: the Tent chaotic mapping, the reverse learning strategy, and the solution update, with corresponding time complexities of $O(N)$, $O(2 \times T \times N)$, and $O(T \times N)$, respectively. The total computational complexity of the chaotic reverse learning strategy is $O(N) + O(3 \times T \times N)$, the time complexity of the adaptive dynamic exploration strategy is $O(T \times N \times D)$, and the computational complexity of the time-varying Cauchy-enhanced hard encirclement strategy is $O(T \times N \times D)$.

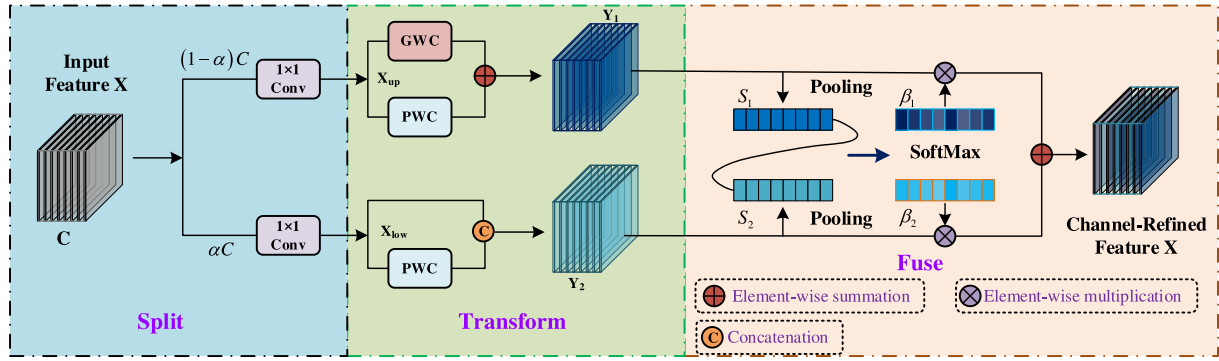


Fig. 7. Structure of the CRSU.

Based on the above, ignoring the constant factors with minimal impact on time complexity, the overall complexity of the MS-HHO algorithm is $O(N) + O(T \times N) + O(T \times N \times D)$, and the improvements do not significantly increase the time complexity.

2.3. Dual-modal redundant feature selection network

The backbone network used in this study is shown in Fig. 5. It takes both STFT and SWT image data as dual-modal inputs and consists of two parts: the dual-modal feature extraction network and the deep feature fusion network.

The dual-modal redundant feature selection network consists of two branches, each incorporating a multi-scale redundant feature selection network (MSRFS), conventional convolutional layers (Conv-BN-Elu, CBE), and a transformer feature extraction layer (TFE). First, the STFT time-frequency image F_{STFT} and the SWT image F_{SWT} are fed into their respective branches. After passing through the MSRFS and CBE layers, the multi-scale features F_{SWT}^2 and F_{STFT}^2 are extracted. The outputs of the MSRFS and TFE layers from both branches are then fused using the dense residual feature fusion network (DRF) to obtain the fused features F_{Fuse}^1 , F_{Fuse}^2 , and F_{Fuse}^3 , enabling deep fusion and reutilization of dual-modal features.

2.3.1. Multiscale redundancy feature selection network (MSRFS-Net)

The MSRFS-Net is illustrated in Fig. 6. This module incorporates a multiscale deeply separable convolutional module (MS-DSCConv), channel redundancy selection unit (CRSU), spatial redundancy selection unit (SRSU), and a standard convolution-BN-GELU layer (CBG). The computational procedure of the MSRFS-Net network is described as follows:

$$F_{SWT}^{CSA1} = \text{Conv}_{1 \times 1}(\text{Concat}[\text{Conv}_{MSC}(F_{SWT}) \oplus Y_1, S_{M1} \otimes \text{Conv}_{MSC}(F_{SWT})]) \quad (11)$$

$$F_{SWT}^{CSA2} = \text{Conv}_{1 \times 1}(\text{Concat}[\text{Conv}_{MSC}(F_{SWT}^{CSA1}) \oplus Y_2, S_{M2} \otimes \text{Conv}_{MSC}(F_{SWT}^{CSA1})]) \quad (12)$$

$$F_{SWT}^{CSA} = \text{Concat}[F_{SWT}^{CSA2}, \text{Conv}_{1 \times 1}(\partial(\text{Conv}_{CBG}(\partial(\text{Conv}_{CBG}(F_{SWT})))))] \quad (13)$$

where F_{SWT} represents the input SWT image features, while Y_1 and Y_2 denotes the features after selection by the channel redundancy unit. S_{M1} and S_{M2} correspond to the spatial importance weight matrices calculated by the first and second spatial redundancy selection units, respectively. The "concat" refers to the concatenation operation, the Conv_{MSC} represents the multi-scale convolution process, and ∂ indicates the GELU activation function in the CBG convolution.

(1) Channel redundancy selection unit

In traditional CNN networks, the features extracted from different

channels are often correlated, which fails to fully exploit complementary feature information, resulting in redundancy between channels (Li et al., 2023). To address this issue, a channel reconstruction unit is introduced to reduce the impact of feature channel redundancy on the model's learning performance. The channel reconstruction unit consists of three steps: split, transform, and fuse, as shown in Fig. 7.

In the separation phase, the input feature $X \in \mathbb{R}^{c \times h \times w}$ is divided into two parts, αC and $(1 - \alpha)C$, based on its channels, the channel split ratio α ranges from $[0, 1]$. Then, a 1×1 convolution is applied for channel compression to improve computational efficiency. Additionally, a channel compression ratio r is used to control the feature channels, balancing the computational cost of the CRSU unit. After these operations, the input feature is split into two parts, X_{up} and X_{low} .

In the upper part of the transformation phase, grouped convolution and pointwise convolution are employed to form a "feature enrichment extractor" that fully extracts the features X_{up} . Grouped convolution reduces parameters and computational complexity but disrupts channel interaction. To address this, parallel pointwise convolution is introduced to compensate for the loss of channel information. Grouped convolution with a kernel size of $k \times k$ (with $g = 2$ in this paper) and 1×1 pointwise convolution are applied to all features X_{up} . The outputs from the grouped convolution (GWC) and pointwise convolution (PWC) are then fused to generate enriched features Y_1 . The mathematical description of this phase is as follows:

$$Y_1 = M^G X_{up} + M^{P1} X_{up} \quad (14)$$

where $M^G \in \mathbb{R}^{\frac{ac}{g} \times k \times k \times c}$ and $M^{P1} \in \mathbb{R}^{\frac{ac}{r} \times 1 \times 1 \times c}$ represent the learnable weight matrices of GWC and PWC, respectively. $X_{up} \in \mathbb{R}^{\frac{ac}{r} \times h \times w}$ and $Y_1 \in \mathbb{R}^{c \times h \times w}$ denote the input and output features of this phase, respectively.

In the lower section of the transformation stage, a 1×1 pointwise convolution (PWC) is applied to extract shallow hidden detail information from the feature X_{low} , serving as a complement to the enriched features. Meanwhile, the input feature X_{low} is reused without incurring additional computational costs. The calculation formula for features Y_2 in the lower transformation stage is as follows:

$$Y_2 = M^{P2} X_{low} \cup X_{low} \quad (15)$$

where $M^{P2} \in \mathbb{R}^{\frac{(1-\alpha)c}{r} \times 1 \times 1 \times \left(1 - \frac{1-\alpha}{r}\right)c}$ represents the learnable weight matrix of the PWC, and $\cup$ denotes the concatenation operation. $X_{low} \in \mathbb{R}^{\frac{(1-\alpha)c}{r} \times h \times w}$ and $Y_2 \in \mathbb{R}^{c \times h \times w}$ refer to the input and output features, respectively.

In the fusion phase, global average pooling is used to collect the global information $S_m \in \mathbb{R}^{c \times 1 \times 1}$ of feature channels. The mathematical formulation is as follows:

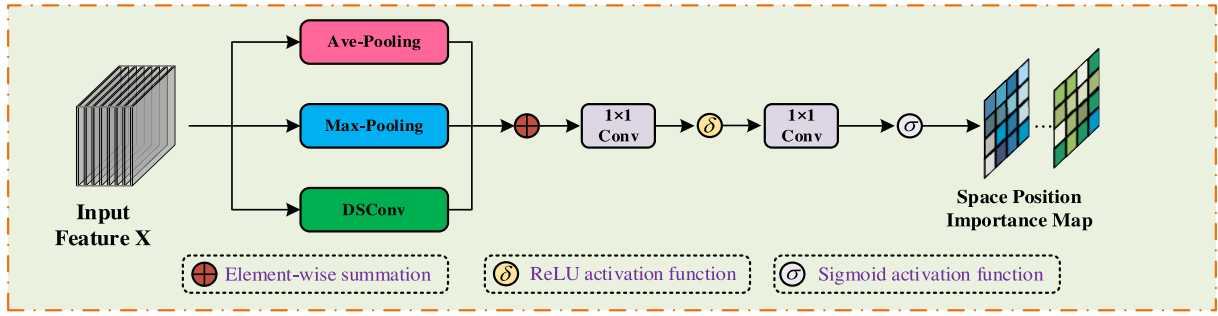


Fig. 8. Structure of the SRSU.

$$S_m = \text{Pooling}(Y_m) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W Y_c(i, j), m = 1, 2 \quad (16)$$

Next, the soft attention operation is employed to generate importance vectors S_1 and S_2 for the global channel vector. The calculation is expressed as:

$$\beta_1 = \frac{e^{S_1}}{e^{S_1} + e^{S_2}} \quad (17)$$

$$\beta_2 = \frac{e^{S_2}}{e^{S_1} + e^{S_2}} \quad (18)$$

where the importance vectors satisfy $\beta_1 + \beta_2 = 1$. Finally, the transformed features Y_1 and Y_2 are merged along the channel dimension to obtain the refined channel features Y . The calculation is expressed as:

$$Y = \beta_1 Y_1 + \beta_2 Y_2 \quad (19)$$

(2) Spatial redundancy selection unit (SRSU)

The feature spatial redundancy unit typically enhances the model's focus on distinct feature positions by learning weights for each feature location, reducing the representational capacity of redundant features. Its structure is shown in Fig. 8.

The computation process for the feature importance map S_M in the SRSU is as follows:

$$A_{AP} = \frac{1}{K_h \times K_w} \sum_{m=0}^{K_h-1} \sum_{n=0}^{K_w-1} X(i \cdot V + m, j \cdot V + n) \quad (20)$$

$$A_{MP} = \max_{m,n} X(i \cdot V + m, j \cdot V + n) \quad (21)$$

$$A_{DSConv} = DSConv(X) \quad (22)$$

$$S_M = \sigma(W_{conv2}^2 * \delta(W_{conv1}^1 * (A_{AP} \oplus A_{MP} \oplus A_{DSConv}) + b_{conv1}^1)) + b_{conv2}^2 \quad (23)$$

where K_h and K_w represent the height and width of the pooling window, while S specifies the stride. Variables m and n range from $m = 0, \dots, K_h - 1$ to $n = 0, \dots, K_w - 1$. $W_{conv1}^1 \in \mathbb{R}^{C \times C \times 1 \times 1}$ represents the weight matrix for the first 1×1 convolution layer, $b_{conv1}^1 \in \mathbb{R}^C$ corresponds to its bias, and ε is the reduction ratio. The spatial attention weight matrix $\beta^{m \times n \times c}$ is generated by the second convolution layer.

2.3.2. Transformer feature extraction network (TFE-Net)

Transformers are particularly adept at analyzing fault signals in complex conditions due to their powerful parallel computation and global feature extraction advantages. The feature extraction module presented in this paper combines the global modeling proficiency of the Transformer with the local modeling strengths of CNN, enabling it to fully exploit essential information in bearing fault signals. Traditional

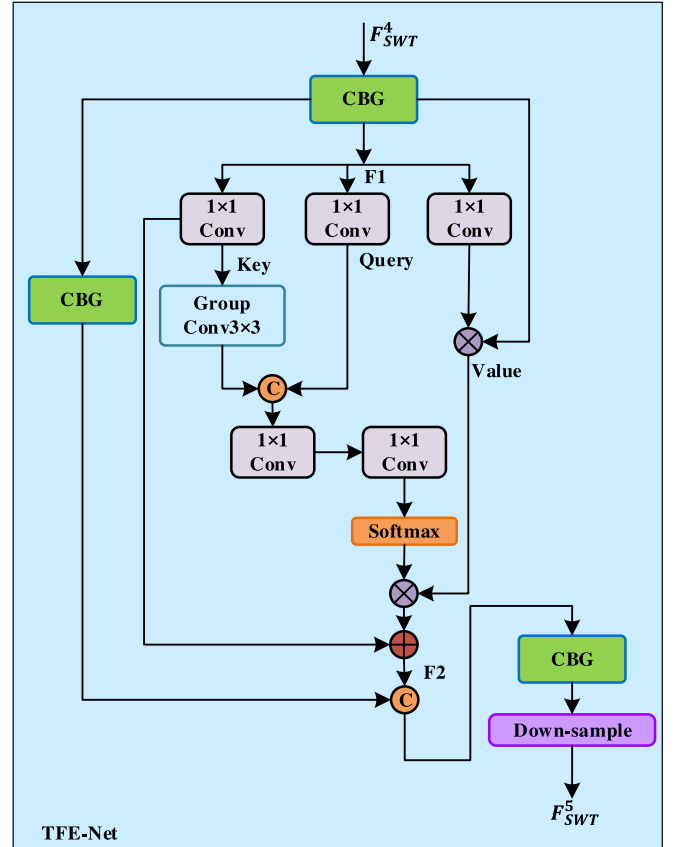


Fig. 9. TFE-Net structure.

CNN face challenges during feature extraction because the limited receptive field of convolutional layers inhibits their ability to capture global context effectively, thereby constraining fault feature extraction. TFE-Net, through its self-attention mechanism, efficiently captures long-distance dependencies among local and global features. Moreover, to alleviate the computational burden of processing high-resolution time-frequency images, TFE-Net adopts convolutional projections instead of traditional linear mappings, effectively reducing computational costs while ensuring robust training and inference performance.

The structure of TFE-Net is shown in Fig. 9, and the output feature F_2 is computed using the following formula:

$$F_2 = \text{reshape}(K + V \times \text{Softmax}(\text{Conv}^{1 \times 1}(\text{Conv}^{1 \times 1}(\text{Concat}(\text{GConv}^{3 \times 3}(K), Q)))) \quad (24)$$

where “reshape” signifies the reshaping function, while “GConv 3×3 ” represents a group convolution with a kernel size of 3×3 .

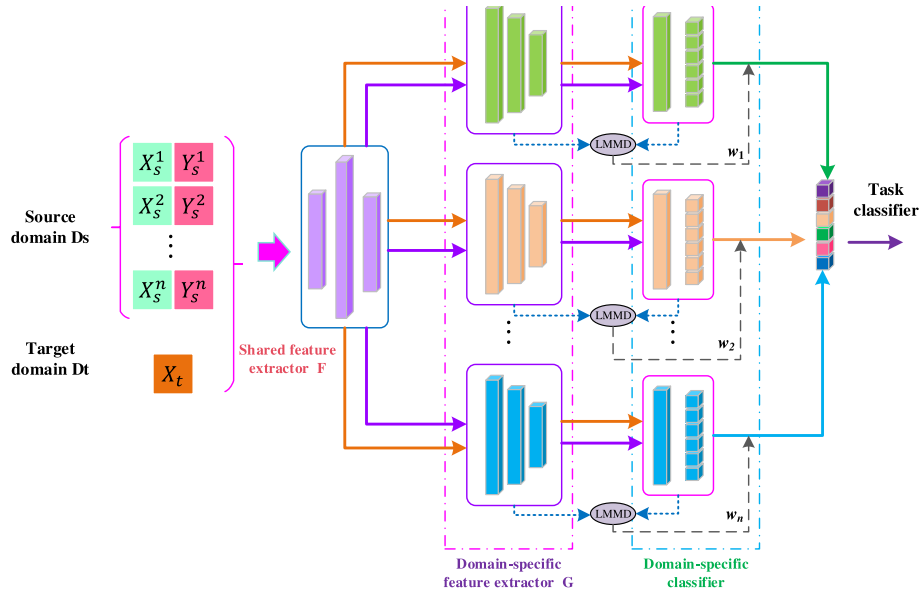


Fig. 10. domain migration fault diagnosis structure diagram.

Finally, the features undergo downsampling to yield the output SWT feature F_{SWT}^5 . The specific procedures are as follows:

$$F_{SWT}^5 = \text{Down}(\text{CBE}(\text{Concat}(F_1, \text{CBE}(\text{CBE}(F_{SWT}^4)))))) \quad (25)$$

2.4. Domain migration fault diagnosis framework

The overall structural framework of fault diagnosis for domain migration adopted in this article is shown in Fig. 10.

The domain transfer fault diagnosis algorithm comprises a single feature extraction network (F), n domain-specific feature extraction networks (G), and n domain-specific state classifiers. In this context, F denotes the STFT-SWT dual-modal feature extraction network designed in this study, and G refers to the two CBE structural networks. The training process mainly consists of two components: feature space alignment and classifier alignment, which are detailed in the following steps:

- 1) Extracting shared features from all domains: For each source domain, namely $X_s^1, X_s^2, \dots, X_s^n$, as well as the target domain X_t , samples are passed through the shared feature extraction network to obtain feature representations $F(x_i^s)$ and $F(x^t)$ in the shared feature space.
- 2) Extraction of domain-specific feature representations: The shared feature representations $F(x_i^s)$ obtained in the previous step are fed into the appropriate feature extraction networks corresponding to each source domain. In contrast, the sample features $F(x^t)$ from the target domain are processed through all domain-specific feature extraction networks. This approach results in the feature representations for the source domains and the target domain, designated as $G_i(F(x_i^s))$ and $G_i(F(x^t))$, respectively.
- 3) The health status $C_i(G_i(F(x_i^s)))$ of the source domain samples is predicted using domain-specific state classifiers. The cross-entropy loss of the classifier is then calculated based on the labels of the source domain samples:

$$L_{cls} = \sum_{j=1}^N E_{x \sim X_{sj}} J(C_i(G_i(F(x_i^{sj}))), y_i^{sj}) \quad (26)$$

- 4) The local maximum mean discrepancy (LMMD) is utilized to align the distribution differences between each source domain and the target domain. LMMD achieves local alignment of same-class samples between the source and target domains by incorporating class information, thereby effectively reducing class confusion. This method is particularly effective in scenarios where there are considerable differences in class distribution between the source and target domains. The calculation formula for LMMD is given below:

$$l_{LMMD} = \text{Dist}(G(F(x^s)), G(F(x^t)))$$

$$= \frac{1}{C} \sum_{c=1}^C \left\| \sum_{x_i^s \in D_s} w_i^{sc} \phi(x_i^s) - \sum_{x_j^t \in D_t} w_j^{tc} \phi(x_j^t) \right\|_H^2 \quad (27)$$

where C represents the number of subdomain categories, “Dist” represents the distance metric function $c \in \{1, 2, \dots, c\}$ indicates the label category, and w_i^{sc} and w_j^{tc} represent the weights of x_i^s and x_j^t that are classified under category c . $\sum_{x_i \in D} w_i^{sc} \phi(x_i)$ represents the weighted sum for category c , and the formula for calculating w_i^{sc} is as follows:

$$w_i^c = \frac{y_{ic}}{\sum_{(x_j, y_j) \in D} y_{jc}} \quad (28)$$

where y_{ic} represents the c -th element of the label vector y_i , while $\sum_{(x_j, y_j) \in D} y_{jc}$ denotes the total number of samples belonging to class c . D encompasses the entire space of both the source and target domains.

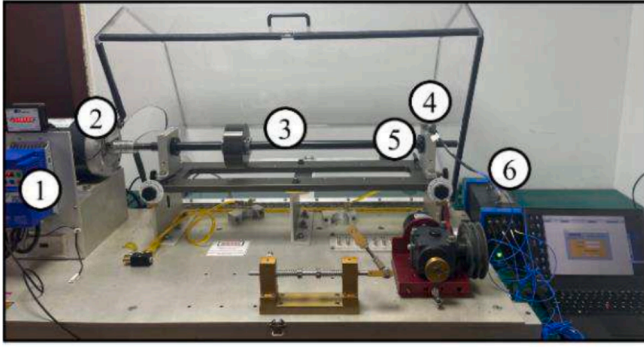
This paper adopts a weighted approach to balance the loss across the various specific feature spaces. The corresponding calculation formula is as follows:

$$L_{LMMD} = w_1 l_1 + w_2 l_2 + \dots + w_N l_N$$

$$= \sum_{j=1}^N w_j \text{Dist}(G_j(F(x^{sj})), G_j(F(x^t))) \quad (29)$$

where w_j represents the weights assigned to each source domain, and x^{sj} indicates that the data comes from source domain subclass j . The weighted approach allows for calculating the actual distance between the restructured source domain and the target.

The formula for calculating the distance metric between the same subclasses of the source and target domains is as follows:



1: Speed control, 2: Motor, 3: Shaft, 4: Acceleration sensor, 5: Bearing, 6: Data acquisition board

Fig. 11. Test rig of HUST bearing dataset.

$$d_l = \text{Dist}(G_j(F(x_r^{sl})), G_j(F(x^l)))$$

$$= \min\{\text{Dist}(G_j(F(x^{s1})), G_j(F(x^{s1}))), \dots, \text{Dist}(G_j(F(x^{sN})), G_j(F(x^{sN})))\} \quad (30)$$

where x_r^{sl} represents data sourced from subclass l in the reassembled source domain dataset.

The weight distributor functions as a binary classifier, determining the origin of each subdomain based on the distributional distance between the source and target domains. The calculation formula is as follows:

$$w(x) = \begin{cases} 1 & \text{Dist}(G_j(F(x_j^{sl})), G_j(F(x^l))) = d_l \\ 0 & \text{Dist}(G_j(F(x_j^{sl})), G_j(F(x^l))) \neq d_l \end{cases} \quad (31)$$

where $j \in \{1, 2, \dots, N\}$ represents the index of the source domain, and $l \in \{1, 2, \dots, L\}$ represents the index of the subdomain.

The optimization goal encompasses three elements: minimizing classification error, addressing domain adaptation loss, and ensuring consistency loss. The complete optimization objective can be formulated as follows:

Condition	20Hz	25Hz	40Hz	60Hz	70Hz	80Hz
Encoding	U_A	U_B	U_C	U_D	U_E	U_F

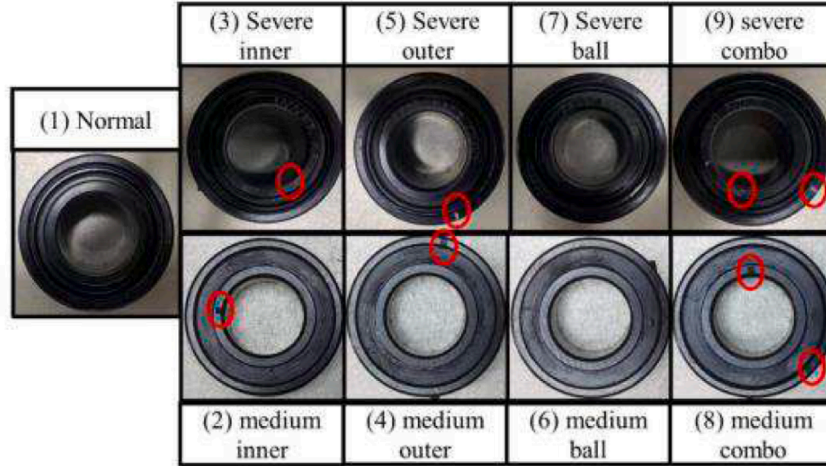


Fig. 12. Photographs of the failure bearings.

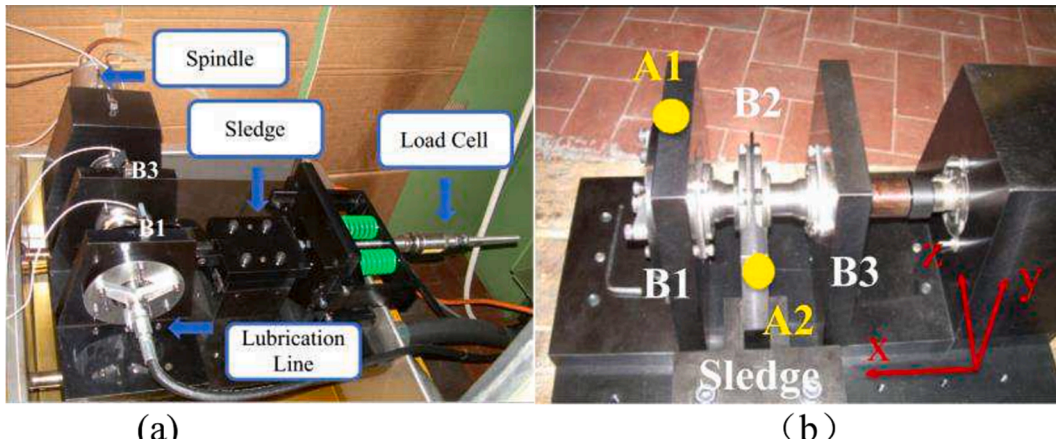


Fig. 13. Experimental platform: (a) General view of the experimental platform; (b) Acceleration sensor position and its orientation reference system.

Table 5

DIRG bearing test speeds and loads.

Nominal load (N)	Nominal speed (Hz)				
	100	200	300	400	500
1000	100	200	300	400	500
1400	100	200	300	400	—
1800	100	300	300	—	—

Table 6

DIRG dataset working condition numbers.

Condition	Encoding	Condition	Encoding	Condition	Encoding
100 Hz-0 N	V _{A1}	200 Hz-1400 N	V _{B3}	400 Hz-0 N	V _{D1}
100 Hz-1000 N	V _{A2}	200 Hz-1800 N	V _{B4}	400 Hz-1000 N	V _{D2}
100 Hz-1400 N	V _{A3}	300 Hz-0 N	V _{C1}	400 Hz-1400 N	V _{D3}
100 Hz-1800 N	V _{A4}	300 Hz-1000 N	V _{C2}	500 Hz-0 N	V _{E1}
200 Hz-0 N	V _{B1}	300 Hz-1400 N	V _{C3}	500 Hz-1000 N	V _{E2}
200 Hz-1000 N	V _{B2}	300 Hz-1800 N	V _{C4}		

$$Loss = L_{cls} + \lambda L_{LMD} \quad (32)$$

where λ represents the weighting parameter.

3. Experiment and analysis

3.1. Data sources

Case 1 utilizes the HUST bearing dataset (Zhao et al., 2024). The test platform for this bearing dataset is shown in Fig. 11. The dataset includes vibration signals of bearings under 11 different operating conditions and 9 distinct health states: healthy, medium inner ring fault, severe inner ring fault, medium outer ring fault, severe outer ring fault, medium ball fault, severe ball fault, medium combined fault, and severe combined fault. The images of bearing faults are displayed in Fig. 12. The test platform employs triaxial accelerometers to capture vibration signals from the bearings in three different directions, with a sampling frequency of 25.6 kHz and a sampling duration of 10.2 s for each data set.

Case 2 utilizes the Turin Polytechnic University bearing dataset (Daga et al., 2019), which was obtained by the dynamic and identification research group (DIRG) at the rolling bearing test bench. The experimental platform is depicted in Fig. 13. This dataset comprises three distinct health states: normal, inner race damage, and rolling element damage, with both inner race damage and rolling element damage each classified into three levels based on severity. The

experiment assessed the vibration characteristics of the bearings under various rotational speeds and loads, with the data summarized in Tables 5 and 6. The data collection frequency was set at 51,200 Hz, with a sampling interval of 10 s. For ease of reference, this bearing dataset is designated as DIRG.

Case 3 utilizes bearing data collected from a wind turbine test rig, where the experimental bearing model is HRB6005. The primary components of this test rig include wind turbine blades (with a radius of 0.6 m), a driving motor, and a planetary gearbox, as illustrated in Fig. 14. The experiment is designed with three fault modes and two operating conditions, detailed in Table 7. The sampling frequency of the experimental data is set at 20,000 Hz. For clarity, this bearing dataset is defined as the WTEP bearing dataset (Fig. 15.)

3.2. Indicators for model evaluation

This research uses four indicators to evaluate the model's predictive performance: Accuracy, Precision, Recall, and F1 Score, as defined below:

(1) Accuracy

$$Accuracy = \frac{\sum_{i=1}^N (TP(i) + TN(i))}{\sum_{i=1}^N (TP(i) + TN(i) + FP(i) + FN(i))} \quad (33)$$

where TP represents the number of true positives, TN represents true negatives, FP represents false positives, FN represents false negatives, i

Table 7

Description of the WTEP dataset.

Labels	Operation condition	Types	Damage size
1	W _A (60r/min); W _B (84 r/min)	Healthy	—
2		Failure of the inner ring	0.6mmx0.3 mm
3		Failure of the outer ring	
4		Failure of the roller	

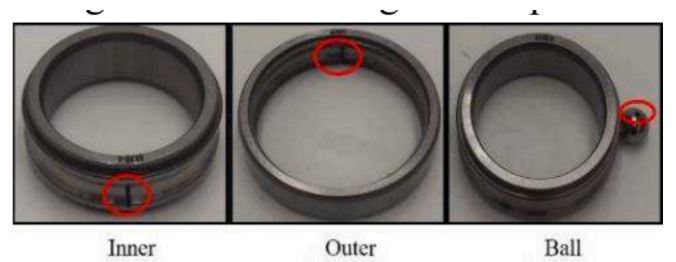


Fig. 15. WTEP dataset of faulty bearings.

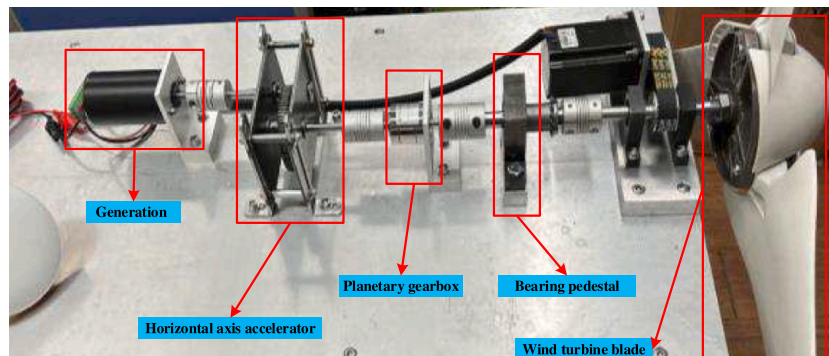


Fig. 14. Wind turbine generator fault diagnosis experimental platform.

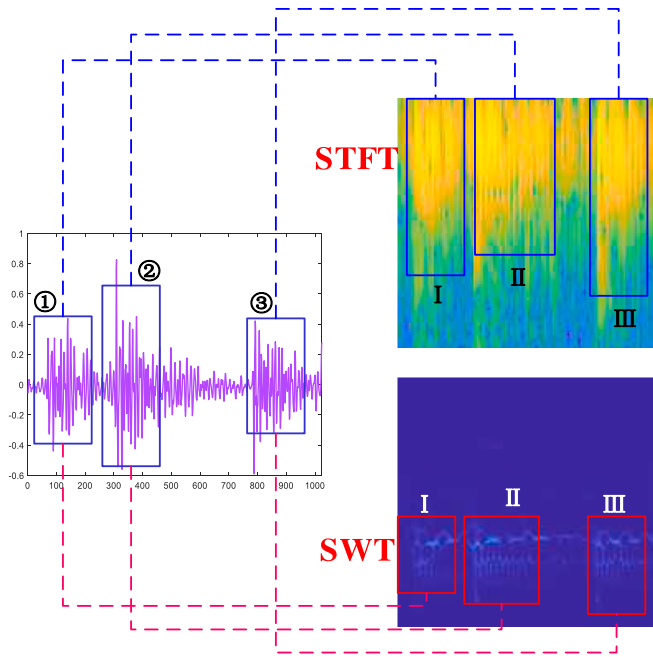


Fig. 16. STFT and SWT image feature data corresponding to vibration signals.

denotes the class.

(2) Recall and Macro-Recall

$$Recall(i) = \frac{TP(i)}{TP(i) + FN(i)} \quad (34)$$

$$Macro - Recall = \frac{1}{N} \sum_{i=1}^N Recall(i) \quad (35)$$

(3) Precision and Macro- Precision

$$Precision(i) = \frac{TP(i)}{TP(i) + FP(i)} \quad (36)$$

$$Macro - Precision = \frac{1}{N} \sum_{i=1}^N Precision(i) \quad (37)$$

(4) F1-Score and Macro-F1 Score

$$F1(i) = 2 \times \frac{Precision(i) \times Recall(i)}{Precision(i) + Recall(i)} \quad (38)$$

$$Macro - F1 = \frac{1}{N} \sum_{i=1}^N F1(i) \quad (39)$$

3.3. Data pre-processing

The frequency and amplitude of bearing vibration signals usually fluctuate over time, and these temporal variations are key for diagnosing the state of mechanical equipment. To capture these time–frequency domain features more effectively, this paper applies the STFT and SWT to extract detailed information from vibration signals, building a feature-rich STFT-SWT dataset.

(1) Short-time Fourier transform.

The STFT divides the signal into short time intervals and applies the Fourier transform within each window, enabling the simultaneous

analysis of the signal's local time and frequency components. Given that $x(t)$ is the input time-domain signal and $w(t)$ is the window function, the STFT of the input signal is defined as:

$$STFT\{x(t)\}(\tau, \omega) = \int_{-\infty}^{\infty} x(t)w(t - \tau)e^{-j\omega t} dt \quad (40)$$

where $w(t - \tau)$ represents the window function centered at τ , and $e^{-j\omega t}$ is the complex Fourier transform kernel.

Fig. 16 illustrates the time–frequency spectrograms of the vibration signal obtained using STFT and SWT. As shown in Fig. 16, three abnormal vibration regions, labeled ①, ②, and ③ in the signal, correspond to regions I, II, and III in both the STFT and SWT spectrograms. These results indicate that the time–frequency feature maps generated by STFT and SWT not only capture the time-domain anomalies in the vibration signal but also extract valuable frequency-domain features, significantly enhancing the diversity of the input feature signals.

(2) Synchrosqueezing wavelet transform

The SWT is a time–frequency analysis method that builds upon the traditional wavelet transform. Its main objective is to achieve higher frequency resolution by further compressing or reassigning the time–frequency map produced by the wavelet transform, thus creating a more accurate time–frequency representation.

The formula for the wavelet transform is shown below:

$$W_x(a, b) = \int_{-\infty}^{\infty} x(t)\psi_{a,b}(t) dt \quad (41)$$

where $\psi_{a,b}(t)$ is the wavelet function under the influence of scale a and translation factor b . During the synchrosqueezing process, the instantaneous frequency at each moment is first estimated. Instantaneous frequency represents the local frequency of the signal at a specific time and is calculated through the phase variation of the wavelet transform coefficients. The specific formula is as follows:

$$\omega_x(a, b) = \frac{\partial}{\partial b} \arg(W_x(a, b)) \quad (42)$$

where $\omega_x(a, b)$ represents the instantaneous frequency, and $\arg(W_x(a, b))$ denotes the phase of the wavelet transform coefficients.

$$T_x(b, \omega) = \int_A W_x(a, b)\eta(\omega - \omega_x(a, b)) \frac{da}{a} \quad (43)$$

where η is the Dirac function, used to compress the energy into a specific frequency location, and $T_x(b, \omega)$ represents the time–frequency coefficients after the SWT.

Fig. 17 illustrates the STFT-SWT time–frequency image features of vibration signals under different fault states in the HUST dataset, recorded at an operating frequency of 20 Hz. The time–frequency characteristics extracted through STFT and SWT reveal variations in amplitude and frequency properties of the vibration signals across different fault conditions. Specifically, as the type of fault changes, the energy distribution of the vibration signals in both the time and frequency domains shows significant differences. Under fault conditions, vibration signals frequently demonstrate sudden changes or unusual energy concentrations, which are typically illustrated through STFT-SWT feature maps. For example, in a healthy state, the energy distribution of the signal is relatively uniform, while in a fault state, energy may be abnormally concentrated in specific frequency ranges or exhibit significant time–frequency changes. These abnormal energy distribution patterns serve as important indicators for fault diagnosis, suggesting that the equipment may have hidden faults. By comparing the STFT-SWT feature maps for different fault states, one can further analyze the signal's frequency characteristics, transient behaviors, and energy distribution in the time–frequency domain. This methodology not only

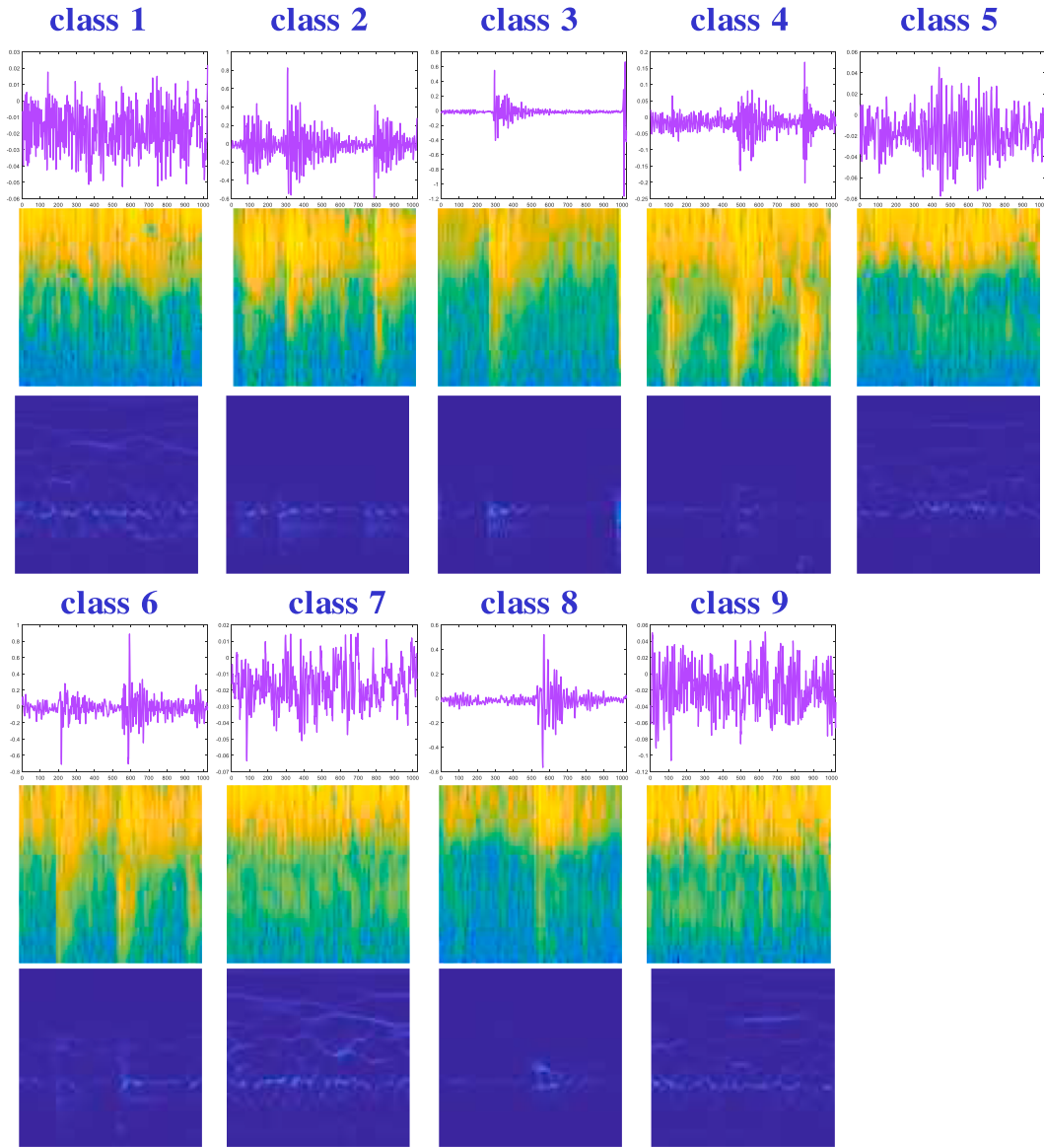


Fig. 17. STFT-SWT feature data for nine vibration signals from the HUST dataset under 20 Hz operating conditions.

captures variations in instantaneous frequency within the vibration signals but also precisely identifies areas of concentrated energy when faults occur, thus providing more comprehensive time–frequency feature information for fault prediction and health management.

3.4. Experiment and result analysis

3.4.1. Validation of optimization algorithm improvement strategy

To further validate the effectiveness of the improved algorithm proposed in this study, we conducted tests using the modified MS-HHO algorithm alongside the bat algorithm (BAT), dragonfly algorithm (DA), and grey wolf optimization (GWO) on several standard test functions. To ensure the validity of the experiments, the population size N for each algorithm was set to 100, the number of iterations T was set to 500, and the dimensionality was fixed at 30. Each algorithm underwent 10 trials, and the experimental results are presented in Fig. 18.

Fig. 18 illustrates the iterative processes of the MS-HHO algorithm, BAT algorithm, DA algorithm, and GWO algorithm on the standard test functions F1 to F12. Compared to the other algorithms, the MS-HHO algorithm demonstrates a faster convergence to the optimal solution on the unimodal test functions F1 to F4. The unmodified HHO algorithm

also exhibits rapid convergence in the later stages of these test functions. For the multimodal test functions F5 to F7, the improved MS-HHO algorithm consistently achieves higher convergence accuracy than the other comparative algorithms. In the F8 test function, both the HHO and MS-HHO algorithms converge near the optimal solution. Furthermore, for the F9 and F11 test functions, the MS-HHO algorithm converges to the vicinity of the optimal solution relatively early, thereby validating the effectiveness of the proposed improvements.

To validate the effectiveness of our proposed optimization method, we compared it with improved versions of the Bat Algorithm (IBA) (Umar et al., 2024), the Dragonfly Algorithm (IDA) (Zhao et al., 2023), and the Grey Wolf Optimizer (IGWO) (Hou et al., 2022). These improved algorithms outperform their classical versions (BAT, DA, and GWO) in terms of convergence speed, search efficiency, and global optimization performance. Specifically, IBA enhances the position and velocity updates of BAT by integrating PSO-inspired strategies. IDA incorporates spiral foraging mechanisms and Cauchy perturbation to improve its ability to escape local optima. IGWO applies dynamic ratio weighting to refine search accuracy and optimization efficiency. These enhancements make them more representative of real-world optimization challenges. We conducted comparative experiments under identical test conditions,

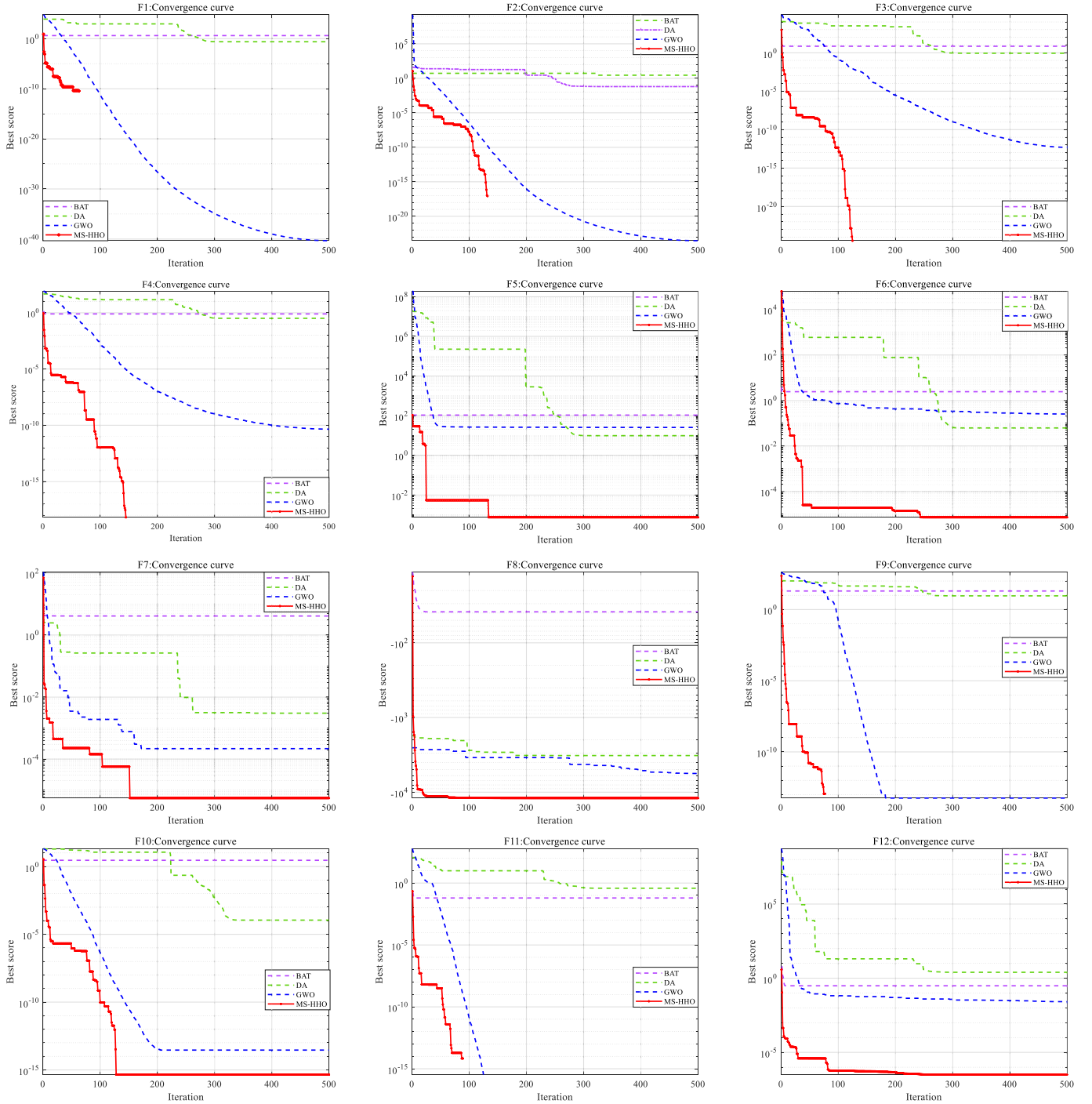


Fig. 18. Comparison results between the MS-HHO algorithm and other algorithms.

evaluating our method against these improved algorithms on multiple benchmark functions. The results are summarized in Table 8.

Table 8 compares the performance of the MS-HHO algorithm with other optimization algorithms on standard test functions. For unimodal functions (F1–F7), MS-HHO achieves the best convergence results on F1 through F6, demonstrating strong global optimization capability and stability. The improved IGWO follows closely, performing well on F1, F2, F3, F4, and F7. Mean-while, IBAT excels on F3 and F4, while IDA shows competitive performance on F1 and F3. For multimodal functions (F8–F12), MS-HHO continues to exhibit strong optimization performance, particularly on F9, F11, and F12, where it achieves the best results. This highlights its ability to maintain high convergence accuracy

in complex search spaces. On F8, MS-HHO slightly lags behind IGWO and HHO but still ranks among the top algorithms, demonstrating strong optimization potential. For F10, its performance is slightly lower than IGWO and IBAT but remains competitive. Most notably, MS-HHO achieves the best convergence result on F12, significantly outperforming other methods, further proving its robustness and adaptability in high-dimensional optimization tasks. In summary, MS-HHO surpasses other optimization algorithms in most test cases, validating its effectiveness and reliability in tackling diverse optimization problems.

To ensure a more appropriate parameter configuration, the improved MS-HHO algorithm was used to optimize the model parameters. Table 9 lists the optimal parameter values of DMRFs-Net obtained through this

Table 8

Test results of the improved algorithm and other algorithms on standard test functions.

Function		BAT	IBAT	DA	IDA	GWO	IGWO	HHO	MS-HHO
F1	Mean	1.490E-02	6.433E-05	6.632E + 02	0.000E + 00	1.085E-41	0.000E + 00	8.263E-105	0.000E + 00
	Std	3.340E-02	3.217E-04	8.750E + 02	0.000E + 00	7.035E-42	0.000E + 00	1.861E-104	0.000E + 00
	Rank	5	4	6	1	3	1	2	1
F2	Mean	8.440E-01	3.022E + 00	1.168E + 01	2.230E-117	4.174E-24	0.000E + 00	8.999E-59	0.000E + 00
	Std	1.141E + 00	3.092E + 00	7.461E + 00	0.000E + 00	4.290E-24	0.000E + 00	1.466E-58	0.000E + 00
	Rank	5	6	7	2	4	1	3	1
F3	Mean	4.483E + 00	0.000E + 00	6.459E + 03	0.000E + 00	2.054E-11	0.000E + 00	1.838E-94	0.000E + 00
	Std	9.929E + 00	0.000E + 00	6.746E + 03	0.000E + 00	4.447E-11	0.000E + 00	3.296E-94	0.000E + 00
	Rank	4	1	5	1	3	1	2	1
F4	Mean	6.400E-02	0.000E + 00	1.386E + 01	3.200E-162	2.915E-10	0.000E + 00	1.136E-53	0.000E + 00
	Std	1.430E-01	0.000E + 00	5.637E + 00	0.000E + 00	3.016E-10	0.000E + 00	2.403E-53	0.000E + 00
	Rank	5	1	6	2	4	1	3	1
F5	Mean	3.838E + 01	2.760E + 01	2.952E + 04	5.127E-01	2.653E + 01	2.867E-01	2.767E + 01	3.000E-03
	Std	2.162E + 01	5.543E + 00	5.591E + 04	2.040E + 00	5.450E-01	2.611 E-02	3.160E-01	2.000E-03
	Rank	8	6	8	3	4	2	5	1
F6	Mean	1.481E + 00	2.023E + 00	6.103E + 02	3.340E-03	2.510E-01	2.611E-02	7.255E-04	1.177E-05
	Std	7.337E + 00	1.848E + 00	6.909E + 02	5.144E-03	3.540E-01	2.860E-01	5.968E-04	7.796E-06
	Rank	6	7	8	3	5	4	2	1
F7	Mean	3.300E-01	1.316E-04	6.200E-02	4.220E-05	3.859E-04	2.961E-07	5.016E-05	2.897E-06
	Std	3.910E-01	1.012E-04	2.900E-02	6.070E-05	2.004E-04	2.373E-07	4.203E-05	2.542E-05
	Rank	8	6	7	3	5	1	4	2
F8	Mean	-1.170E + 03	-1.171E + 02	-6.222E + 03	-9.214E + 03	-6.364E + 04	-1.111E + 04	-1.257E + 04	-1.078E + 04
	Std	3.800E-01	1.227E + 00	5.390E + 02	3.105E + 02	8.038E + 02	4.621E-01	3.800E-02	2.810E-01
	Rank	6	8	7	5	4	2	1	3
F9	Mean	3.412E + 01	0.000E + 00	8.296E + 01	0.000E + 00	1.395E + 00	0.000E + 00	0.000E + 00	0.000E + 00
	Std	1.499E + 01	0.000E + 00	5.772E + 01	0.000E + 00	3.118E + 00	0.000E + 00	0.000E + 00	0.000E + 00
	Rank	3	1	4	1	2	1	1	1
F10	Mean	6.110E-01	8.882E-16	4.824E + 00	2.247E-15	3.892E-15	8.811E-16	4.441E-16	4.441E-16
	Std	6.840E-01	0.000E + 00	1.650E + 00	1.220E-15	2.176E-14	1.164E-16	0.000E + 00	0.000E + 00
	Rank	6	1	7	4	5	2	3	3
F11	Mean	1.700E-02	0.000E + 00	6.628E + 00	0.000E + 00	3.000E-03	0.000E + 00	0.000E + 00	0.000E + 00
	Std	1.500E-02	0.000E + 00	6.293E + 00	0.000E + 00	5.000E-03	0.000E + 00	0.000E + 00	0.000E + 00
	Rank	3	1	4	1	2	1	1	1
F12	Mean	6.460E-01	1.425E-01	1.056E + 01	1.090E-05	2.300E-02	2.133E-03	2.602E-05	1.863E-06
	Std	5.350E-01	1.612E-01	5.533E + 00	3.124E-05	1.800E-02	6.895E-03	1.083E-05	2.386E-06
	Rank	6	7	8	3	5	4	2	1

Table 9

Model parameter search results.

Parameter Index	Hyperparameter name	Search Range	Best Search Results
h ₁	InitialLearnRate	[10 ⁻² -10 ⁻⁶]	10 ⁻³
h ₂	MiniBatchsize	[8,16,32, ...,521,1024]	32
h ₃	MaxEpochs	[20-100]	30
h ₄	Reduction ratio r in the CRSU unit	[2-12]	2
h ₅	Increase ratio ε in the SRSU unit	[2-12]	4

optimization. The search ranges for h₁, h₂, and h₃ were based on references (Gong et al., 2024b; Shallue et al., 2019; Gong et al., 2024a), while those for h₄ and h₅ were derived from references (Woo et al., 2018; Hu et al., 2019). These ranges were further refined through preliminary experiments to determine the most suitable search space.

Table 9 indicates that the hyperparameters for the DMRFS-Net model, specifically InitialLearnRate, MiniBatchsize, and MaxEpochs, should be set to 10⁻³, 32, and 30, respectively, to achieve optimal learning performance. Furthermore, with the channel reduction ratio r in the CRFU unit set to 2 and the channel restoration ratio ε in the SRFU unit set to 4, the model effectively emphasizes essential feature information.

3.4.2. Model diagnostic performance verification

(1) Structure verification

Table 10

Ablation experiment on the HUST dataset.

Method	Name	MSRFS	TFE	DRF	Input STFT	SWT
CBE	M1				✓	
CBE	M2					✓
CBE + TFE	M3		✓		✓	
CBE + MSRFS	M4	✓				✓
CBE + MSRFS + TFE	M5	✓	✓		✓	
CBE + MSRFS + TFE	M6	✓	✓		✓	✓
MSRFS + TFE + DRF	M7	✓	✓	✓	✓	✓

To validate the effectiveness of the proposed dual-modal feature extraction network and the dual-modal fusion module, ablation experiments were conducted on the HUST and DIRG bearing datasets. These experiments aimed to assess the impact of various feature extraction and fusion strategies for different modalities. The arrangements for the ablation experiments are presented in detail in Table 10.

Table 11 illustrates the ablation experiment results of different models on three bearing experimental datasets. When single STFT and SWT time-frequency images are input, the average macro precision (Macro-P), macro recall (Macro-Recall), and macro F1 score (Macro-F1) of the M1 and M2 models are 83.15 %, 79.69 %, and 79.59 %, respectively, across the three datasets. The M3 model demonstrates improvements of 11.42 %, 15.02 %, and 14.7 % in Macro-P, Macro-Recall, and Macro-F1, respectively, when compared to the M1 model. The M4 model shows gains of 11 %, 15.09 %, and 15.99 % in these metrics compared to the M2 model. The M5 model further increases the Macro-P, Macro-Recall, and Macro-F1 by 3.49 %, 4.07 %, and 3.96 %, respectively, in comparison to the M3 model. Compared to the M5 model, the M6 model

Table 11

Ablation experiment on the DIRG dataset.

Name	HUST Macro -P	Macro- Recall	Macro -F1	DIRG Macro -P	Macro Recall	Macro F1	WTEP Macro -P	Macro Recall	Macro F1
M1	87.15	81.23	80.64	80.72	77.71	78.69	80.01	78.88	78.97
M2	86.36	82.47	81.70	82.84	78.74	78.25	81.82	79.08	79.28
M3	91.02	90.12	89.90	92.90	91.56	91.52	92.29	91.86	91.92
M4	92.34	91.85	91.97	94.02	92.57	92.65	92.24	92.16	92.86
M5	94.45	93.58	93.60	96.72	96.34	96.35	94.66	94.76	94.22
M6	96.08	95.36	95.37	97.92	97.76	97.77	96.08	96.11	96.64
M7	99.63	99.64	99.63	99.57	99.26	99.27	99.17	99.42	99.46

Table 12

Ablation results for the computational complexity of various models.

Name	Parameters	FLOPS	Training Time (s)	Test Time (s)
M1	30.21 M	1.82G	75	0.06
M2	30.21 M	1.82G	74	0.06
M3	33.64 M	2.12G	86	0.27
M4	20.17 M	0.98G	62	0.04
M5	23.56 M	1.31G	112	0.38
M6	47.24 M	3.42G	152	0.49
M7	49.42 M	3.81G	167	0.51

yields smaller gains of 1.48 %, 1.6 %, and 1.97 %, respectively, in these performance metrics. Finally, the M7 model demonstrates a further increase in average Macro-P, Macro-Recall, and Macro-F1 by 2.86 %, 3.14 %, and 2.96 %, respectively, compared to the M6 model.

The following conclusions can be obtained from the results of the three ablation experiments described above:

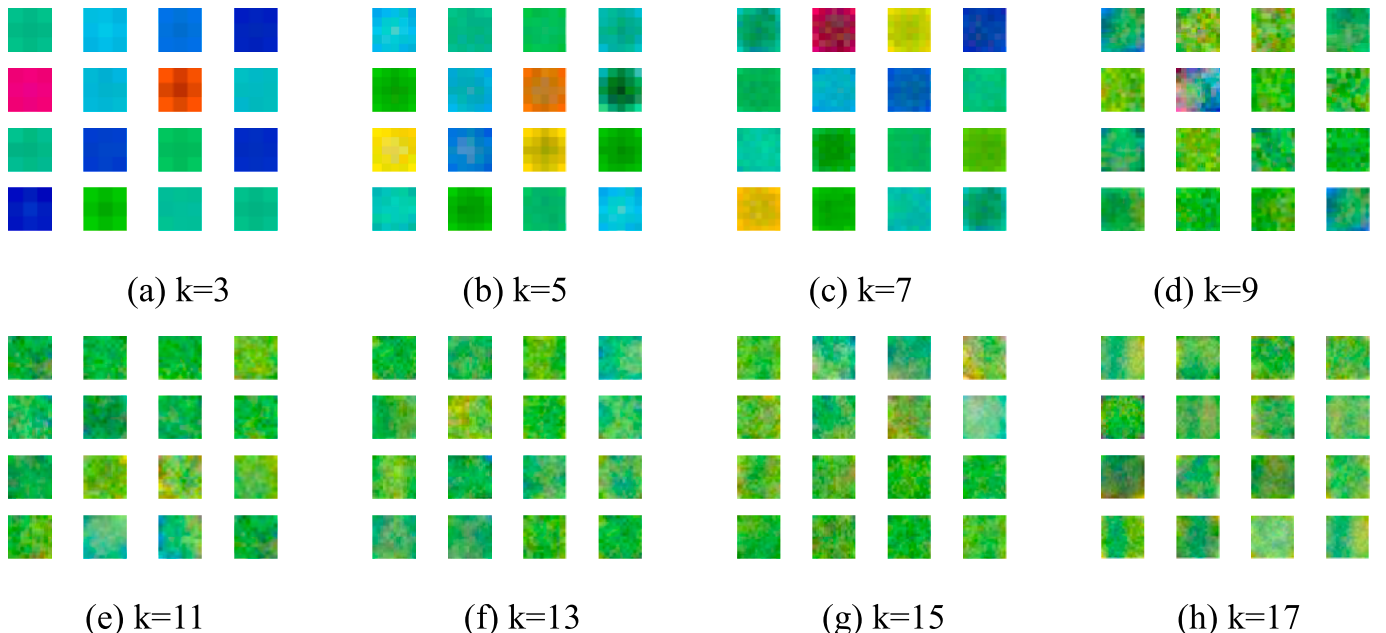
- By adding the TFE module to the single-branch baseline model CBE, the average macro precision increased by 11.42 % across the three bearing datasets. The results show that the TFE module effectively captures long-range dependencies between local and global features, allowing the model to better capture global context and significantly improve diagnostic performance.
- Adding the MSRFS module to the single-branch baseline model CBE led to an 11 % increase in average macro precision across the three bearing datasets. The results show that the MSRFS module refines features in both the channel and spatial dimensions,

removing redundant or irrelevant information, which enhances fault feature sensitivity and improves the model's efficiency, generalization, and robustness.

- The dual-input CBE + MSRFS + TFE model, which uses both STFT and SWT images, achieved a 1.48 % increase in average macro precision compared to the single-input CBE + MSRFS + TFE model. STFT captures global features in the frequency domain, providing broad signal information, while SWT focuses on local time-domain features, highlighting details in fault signals. Combining these inputs allows the model to represent fault features more accurately, reducing information loss.
- By adding the DRF module to the dual-input CBE + MSRFS + TFE model to deeply integrate features at different levels, the model's average macro precision improved by 2.86 % across the three datasets. The DRF module uses dense residual connections, enabling feature transmission and reuse across layers, minimizing feature loss and ensuring key information flows through the network.

To further validate the performance of the proposed model, we evaluate it based on the number of parameters, complexity (FLOPs), and training/testing time. The results are shown in Table 12.

Table 12 shows the ablation results for the CBE model, with M1 to M7 models maintaining consistent depth, and M6 and M7 models using a dual-branch structure. The M3 model, which adds the TFE network to the baseline CBE, increases the number of parameters by 11.26 % and computational complexity by 16.48 %, but achieves an 11.43 % improvement in diagnostic accuracy. The TFE module enhances

**Fig. 19.** Output responses of convolution kernels of different sizes.

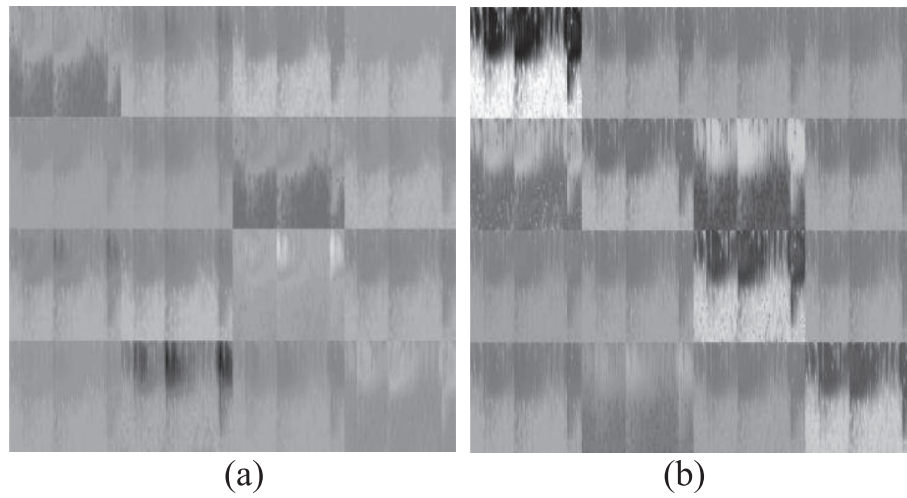


Fig. 20. Comparison results of attention mechanisms: (a) The output response without the redundancy selection unit; (b) The output response with the redundancy selection unit.

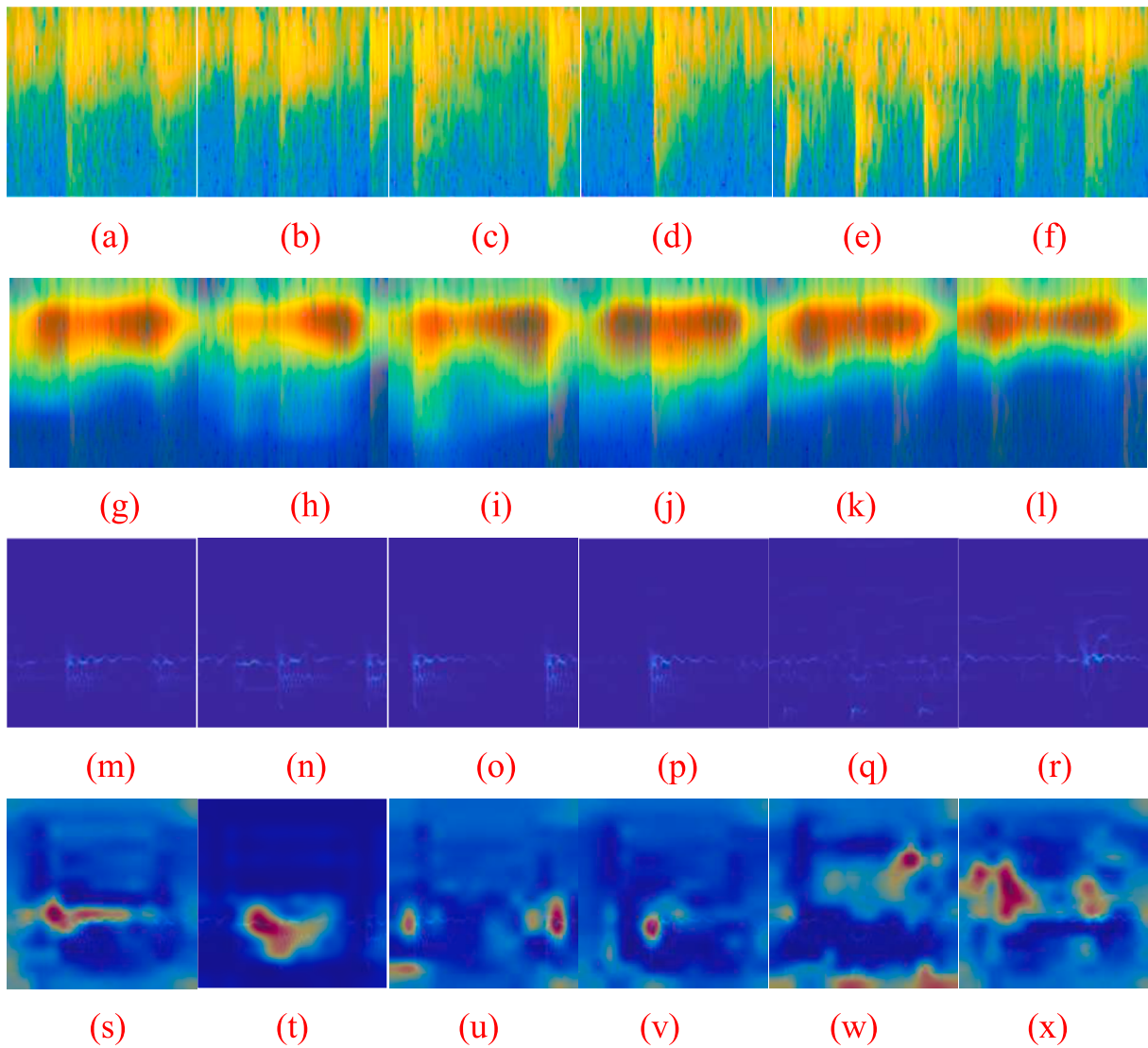


Fig. 21. Grad-CAM feature visualization results: (a)(b)(c)(d)(e)(f) are the STFT time–frequency images corresponding to different fault modes; (g)(h)(i)(j)(k)(l) are the Grad-CAM visualization results corresponding to the STFT time–frequency images; (m)(n)(o)(p)(q)(r) are the SWT time–frequency images corresponding to different fault modes; (s)(t)(u)(v)(w)(x) are the Grad-CAM visualization results corresponding to the SWT time–frequency images.

diagnostic performance with only a slight increase in computational costs. The M4 model, incorporating the lightweight MSRFS network, reduces parameters by 33.44 % and computational complexity by 46.15 %, while improving diagnostic accuracy by 11 %. Replacing standard convolutions with MSRFS significantly boosts accuracy while saving computational resources. The M7 model, with its dual-branch structure, offers the highest diagnostic accuracy without adding extra computational cost. Although the training time for M7 is longer than that of M1 and M2, its testing time is only 0.47 s longer, which does not impact real-time diagnosis.

Fig. 19 illustrates the output responses produced by convolution kernels of various sizes.

Fig. 19 illustrates that as the size of the convolution kernel increases from 3×3 to 17×17 , the feature visualization results undergo significant changes. The 3×3 kernel captures rich details, effectively identifying local features like edges and corners, but has a limited receptive field and lacks global context. The 5×5 and 7×7 kernels progressively enlarge the receptive field, retaining some local information while incorporating more global context. The 9×9 to 13×13 kernels further blur details, shifting focus to global information, whereas the 15×15 and 17×17 kernels nearly completely obscure local details, resulting in smoother feature maps that emphasize overall structure over fine features. The multi-scale convolution structure's strength lies in its simultaneous use of kernels of various sizes, enabling the capture of local and global features at different levels. This approach effectively integrates multi-scale information within the model, enhancing feature representation and improving its ability to recognize complex patterns.

Fig. 20 shows the output responses without the attention mechanism and the output responses with the channel and spatial attention mechanisms from the MSCAM module designed in this paper.

Fig. 20(a) shows that the feature output response without the redundancy selection unit is relatively blurred, with uniformly distributed features across the output response map. This makes it challenging for the model to highlight key information. In contrast, Fig. 20(b) demonstrates that the feature map with the redundancy selection unit exhibits clearer local features, with certain regions significantly enhanced. This indicates that the model better captures critical features. The redundancy selection unit guides the model to focus on distinguishable areas while suppressing irrelevant information. As a result, the model equipped with the redundancy selection unit extracts more edge and high-frequency details, enhances feature representation, and avoids attention to irrelevant regions, thereby improving generalization performance. Overall, the redundancy selection unit enables the model to more accurately identify key features, demonstrating stronger classification and generalization capabilities.

In fault diagnosis tasks, high-dimensional features often contain a significant amount of redundant information, which not only reduces the model's generalization ability but also affects computational efficiency. To better analyze the issue of feature redundancy and gain deeper insights into the model's decision-making process under different operating conditions, we introduce gradient-weighted class activation mapping (Grad-CAM) (Li et al., 2023) for feature visualization. Grad-CAM is a gradient-based visualization technique that highlights key feature regions the model focuses on during decision-making, thereby identifying features that play a crucial role in classification while revealing potential redundancies. By computing the gradient of the target class concerning the feature maps of the last convolutional layer, Grad-CAM generates heatmaps that intuitively illustrate the feature regions of interest to the model. The Grad-CAM visualizations of different fault types used in this study are presented in Fig. 21.

As shown in Fig. 21, the proposed model exhibits superior performance in filtering redundant features in vibration signal fault diagnosis. By analyzing and comparing the time-frequency representations obtained using Short-Time Fourier Transform (STFT) and Stationary Wavelet Transform (SWT) with their corresponding Grad-CAM visualizations, the following observations can be made:

Table 13

Comparison results of diagnostic performance and efficiency of different models.

Method	ACC (%)	Train-Time (s)	Test-Time (s)	Parameters	FLOPS
CNN + MSRFS	89.14	69	0.07	22.96 M	1.25G
LSTM + MSRFS	92.62	95	0.15	24.18 M	1.44G
BiGRU + MSRFS	95.08	128	0.23	26.76 M	1.68G
ResNet + MSRFS	96.24	164	0.25	28.55 M	1.95G
DMRFS-Net	99.53	175	0.41	49.42 M	3.18G

- (1) **Selective Attention to Key Features:** In the STFT time-frequency representation Fig. 21 (a), high-energy regions are concentrated in the upper half, while the Grad-CAM visualization Fig. 21(g) indicates that the model primarily focuses on these high-energy areas and largely disregards the low-energy regions in the lower half, which correspond to redundant features. This indicates that the model effectively discriminates between informative and redundant features, ensuring its focus remains on signal components critical for fault diagnosis.
- (2) **Accurate Detection of Transient Impact Features:** In the SWT time-frequency representation Fig. 21 (m), transient impact characteristics of the vibration signal primarily appear in the middle and lower regions. The corresponding Grad-CAM visualization Fig. 21 (s) illustrates that the model selectively attends to these transient impact areas while disregarding irrelevant regions. This suggests that the model effectively captures short-duration transient events, which are often crucial for fault identification.
- (3) **Efficient Redundant Feature Filtering:** Experimental findings demonstrate that when processing STFT and SWT representations, the model autonomously extracts meaningful features while suppressing redundant information. This ability not only improves classification accuracy but also enhances the model's interpretability, making it highly applicable for real-world fault diagnosis.

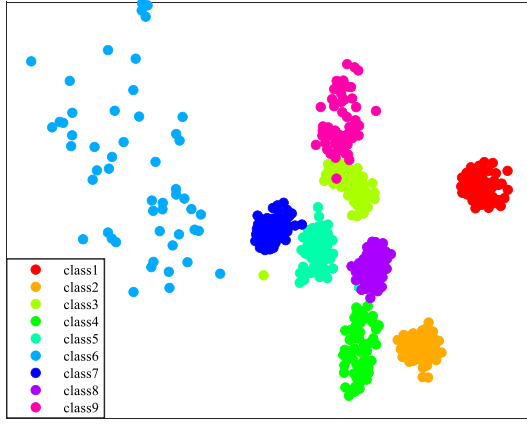
Overall, the integration of Grad-CAM visualization enables the proposed model to effectively prioritize critical vibration signal features while minimizing the influence of redundant components. This contributes to enhanced diagnostic performance and interpretability, offering a robust and efficient solution for vibration-based fault detection.

(2) Comparison of different algorithms

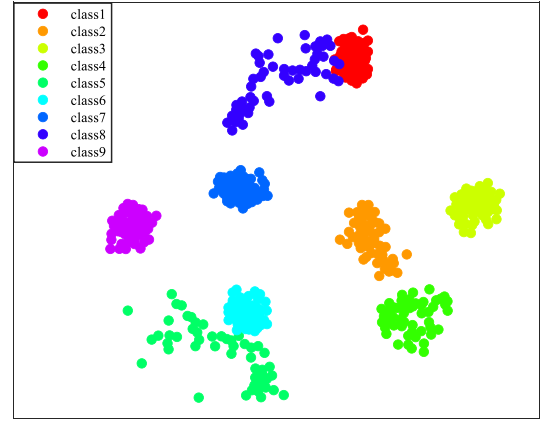
To validate the superiority of the proposed model's performance, comparisons were made with traditional models such as CNN (Wang et al., 2022), LSTM (Tian et al., 2023), BiGRU (Zhang et al., 2022), and ResNet (Chen et al., 2023).

According to Table 13, DMRFS-Net achieves improvements in average accuracy of 11.32 %, 7.46 %, 4.68 %, and 3.42 % over the benchmark models CNN, LSTM, BiGRU, and ResNet across three bearing datasets. This improvement is attributed to the dual-branch feature extraction structure of DMRFS-Net, which is designed to maximize the use of limited data. While DMRFS-Net exhibits a slightly higher parameter count and computational complexity compared to the other single-branch models, its diagnostic performance is superior. Furthermore, the training times of CNN, LSTM, BiGRU, and ResNet models are reduced by 60.57 %, 45.71 %, 36.72 %, and 6.29 %, respectively, in comparison to DMRFS-Net. In the testing phase, the testing time of DMRFS-Net is only 0.34 s longer than that of CNN. These findings suggest that DMRFS-Net not only ensures high diagnostic accuracy but also maintains a fast online testing speed.

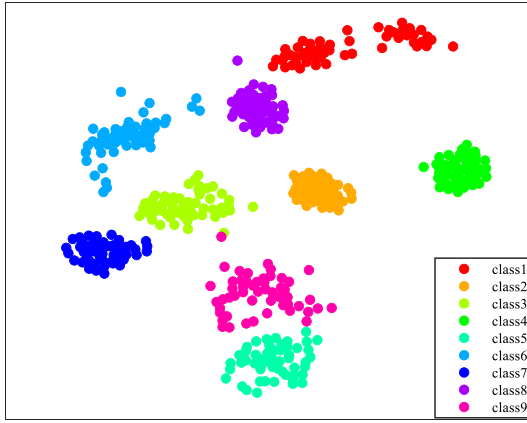
Fig. 22 illustrates the feature visualization results of different models on the HUST dataset. As shown in Fig. 22(a), the feature clustering



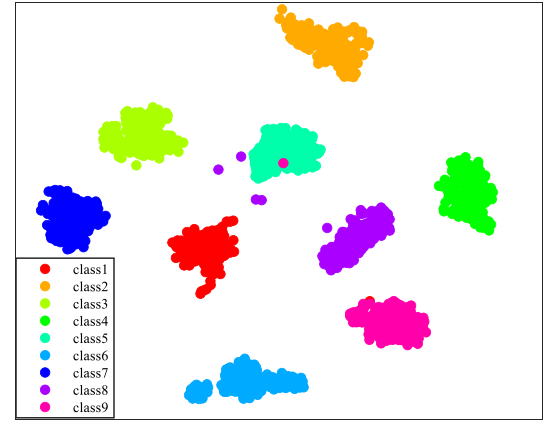
(a) CNN



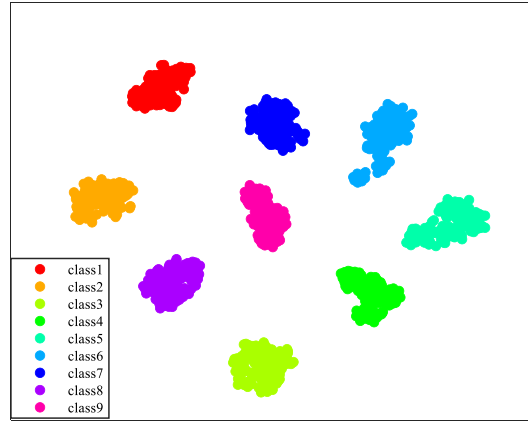
(b) LSTM



(b) BiGRU



(d) ResNet

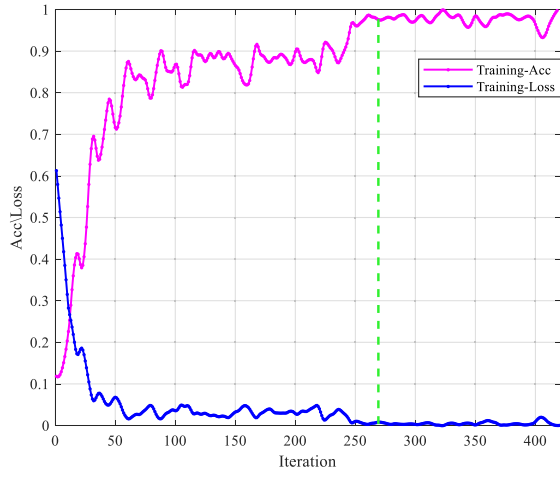


(e) DMRFS-Net

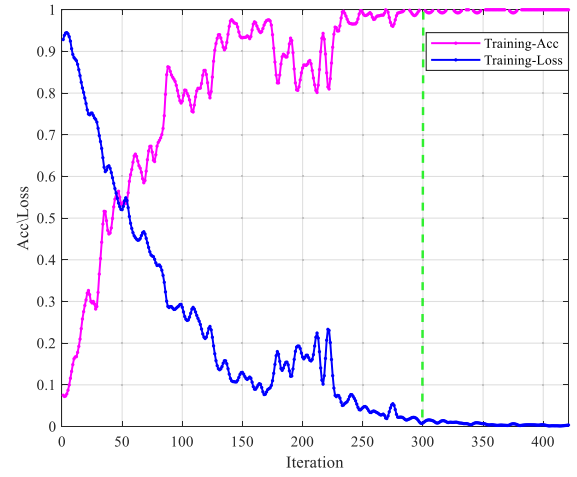
Fig. 22. Feature visualization results of different models at 20 Hz for the HUST dataset.

results of the CNN model exhibit significant overlap and unclear distribution between classes. The samples from different categories are not fully separated in the low-dimensional space, with overlapping observed in fault categories 3 and 9, as well as in fault categories 4, 5, and 8. The

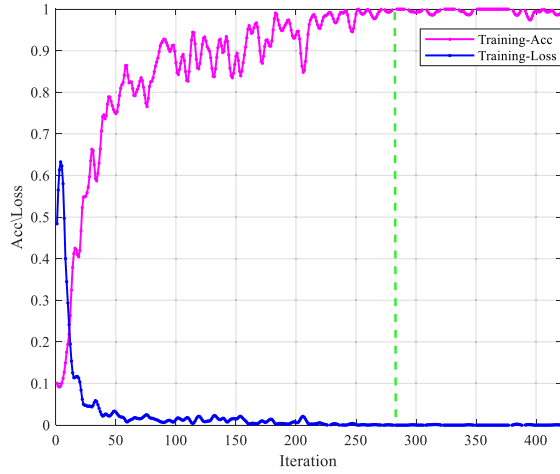
LSTM model shows some improvement in feature distribution compared to CNN, with certain categories (such as 1, 2, 3, 6, 7, and 9) demonstrating better separation in the low-dimensional space. However, some categories (such as 5 and 8) remain dispersed, and the boundaries



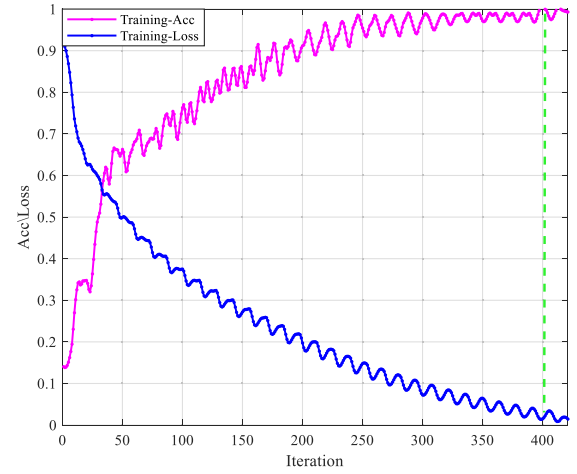
(a) CNN



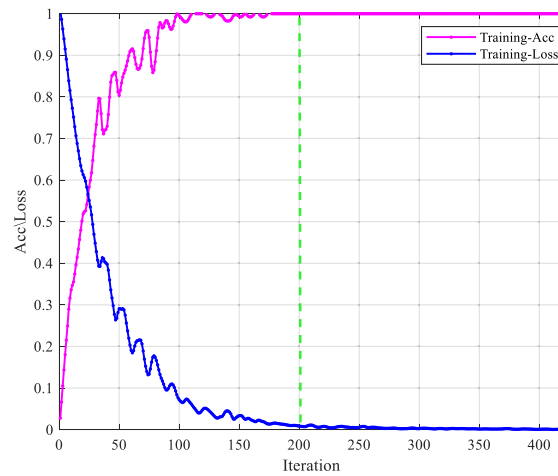
(b) LSTM



(c) BiGRU



(d) ResNet



(e) DMRFS-Net

Fig. 23. Training curves of different models on the HUST dataset.

Table 14

Summarizes the transfer tasks conducted on the HUST dataset.

Number	Scenarios	Transfer task
1	variable operating condition	$U_A \rightarrow U_B$
2		$U_B \rightarrow U_C$
3		$U_C \rightarrow U_E$
4		$U_D \rightarrow U_F$
5		$U_E \rightarrow U_D$
6	variable operating condition	$U_F \rightarrow U_A$
7		$U_A U_B \rightarrow U_C$
8		$U_C U_D \rightarrow U_F$
9		$U_E U_F U_D \rightarrow U_A$
10		$U_A U_C U_D \rightarrow U_B$
11		$U_B U_C U_E U_F \rightarrow U_D$
12		$U_A U_B U_C U_D \rightarrow U_E$

Table 15

Summarizes the transfer tasks conducted on the DIRG dataset.

Number	Scenarios	Transfer task
1	variable operating condition	$V_{A1} \rightarrow V_{B1}$
2		$V_{A2} \rightarrow V_{B2}$
3		$V_{B4} \rightarrow V_{C4}$
4		$V_{C2} \rightarrow V_{D2}$
5		$V_{C3} \rightarrow V_{D3}$
6	variable operating condition	$V_{D1} \rightarrow V_{E1}$
7		$V_{A1} V_{B2} \rightarrow V_{C3}$
8		$V_{C2} V_{D3} \rightarrow V_{E1}$
9		$V_{B1} V_{B2} V_{C3} \rightarrow V_{A3}$
10		$V_{A1} V_{B3} V_{C1} \rightarrow V_{C2}$
11		$V_{A2} V_{B2} V_{C3} V_{D1} \rightarrow V_{D3}$
12		$V_{B1} V_{C3} V_{D1} V_{E2} \rightarrow V_{C1}$

between classes are not well defined. In contrast, the clustering performance of the BiGRU and ResNet models shows significant enhancement, achieving effective clustering of the nine distinct fault categories. The proposed DMRFS-Net model exhibits the most pronounced advantages, achieving complete separation of the categories in low-dimensional space. There is almost no overlap between the different colored categories, and the intra-class clustering is exceptionally tight, demonstrating a high degree of inter-class distinction. The experimental results indicate that the DMRFS-Net model is better able to learn the key feature information of different fault modes, resulting in superior classification performance compared to other models.

Fig. 23 illustrates the training curves for various models. From Fig. 23(a), it can be observed that the CNN model's training accuracy increases rapidly in the initial stages, peaking around the 260th iteration, at which point both the training accuracy and loss stabilize. Fig. 23(b) and 23(c) reveal that the LSTM model exhibits fluctuations during training, with the LSTM and BiGRU models stabilizing their losses and achieving optimal performance after 300 and 280 iterations, respectively. The accuracy of ResNet shows a stable upward trend, while the loss function decreases significantly in the initial phase and then declines gradually. However, there are slight oscillations in the iteration curve of ResNet, which can be attributed to the inherent structure of ResNet that may cause certain layers' gradients to remain unstable, resulting in irregular jumps during updates. The training performance of DMRFS-Net is relatively stable, with accuracy and loss trends similar to those of CNN and BiGRU, stabilizing after approximately 200 iterations. Experimental results indicate that, compared to other benchmark models, the proposed model rapidly converges within a few iterations while maintaining very low loss values and high accuracy, thus ensuring its efficiency and robustness in practical applications.

(3) Sparse sample fault diagnosis verification across working conditions

To rigorously assess the transfer diagnostic performance of the

Table 16

Summarizes the transfer tasks conducted on the WTEP dataset.

Number	Scenarios	Transfer task
1	5 %	$W_A \rightarrow W_B$
2		$W_B \rightarrow W_A$
3	10 %	$W_A \rightarrow W_B$
4		$W_B \rightarrow W_A$
5	15 %	$W_A \rightarrow W_B$
6		$W_B \rightarrow W_A$
7	20 %	$W_A \rightarrow W_B$
8		$W_B \rightarrow W_A$

proposed model under cross-condition small-sample scenarios, comparative analyses were conducted against conventional models (CNN, LSTM, BiGRU, and ResNet) as well as advanced domain adaptation approaches (MSCNN-CBAM (Du et al., 2022), SimAM-CDAN (Tang et al., 2024), and AmaCNN-CORAL (Shao and Kim, 2024)). The experiments encompassed three datasets and four controlled training sample proportions (5 %, 10 %, 15 %, and 20 %) through systematic ablation studies. Within the HUST and DIRG bearing datasets, 12 different transfer tasks were formulated for each experimental condition, covering both single-source and multi-source domain adaptation, as illustrated in Tables 14 and 15. The corresponding transfer tasks for the WTEP dataset are summarized in Table 16. Each experiment was conducted 10 times, and the final comparative analysis was based on the averaged diagnostic accuracy.

The values of λ in the transfer tasks were as follows:

$$\lambda = 2 / (1 + e^{10i/n}) - 1 \quad (44)$$

where i indicates the current iteration of training, and n signifies the total number of iterations. Formula (44) guarantees that the value λ is constrained between 0 and 1, adjusting dynamically as the training progresses.

Figs. 24 to 27 illustrate the diagnostic comparison results of different models when applied to the HUST dataset under conditions of cross-condition small sample size.

Fig. 24 to Fig. 27 illustrate that the diagnostic accuracy of all models improves as the proportion of training samples increases. Traditional models, such as CNN and LSTM, exhibit relatively lower performance across all tasks, particularly under low training sample conditions (5 % and 10 %). Specifically, CNN achieves an average diagnostic accuracy of 58.54 % and 68.67 % at 5 % and 10 % training samples, respectively, while LSTM attains 64.00 % and 74.82 %. These results indicate that CNN and LSTM have limited generalization ability in small-sample scenarios. Compared to CNN and LSTM models, BiGRU and ResNet achieve an average diagnostic accuracy improvement of 22.26 % and 24.01 %, respectively, when trained with only 5 % of the available samples. Among traditional models, ResNet yields the highest diagnostic accuracy, indicating that its residual connections effectively enhance feature utilization. Furthermore, at 5 % training samples, attention-based models such as MSCNN-CBAM, SimAM-CDAN, AmaCNN-CORAL, and the proposed DMRFS-Net improve diagnostic accuracy by 2.75 %, 4.28 %, 6.80 %, and 10.10 % over ResNet, respectively. These findings suggest that even under extremely limited data conditions (5 %), models with feature selection mechanisms can still extract a sufficient number of informative features, maintaining an average diagnostic accuracy of approximately 80 %. As shown in Fig. 25, when the training sample ratio is 10 %, the proposed DMRFS-Net outperforms MSCNN-CBAM, SimAM-CDAN, and AmaCNN-CORAL by 6.63 %, 4.93 %, and 1.83 % in terms of average diagnostic accuracy, respectively. The experimental results indicate that DMRFS-Net consistently surpasses all comparative models across various transfer tasks when trained with a limited number of samples (5 % and 10 %), demonstrating strong generalization capabilities.

As illustrated in Fig. 27, when the training sample ratio reaches 20 %,

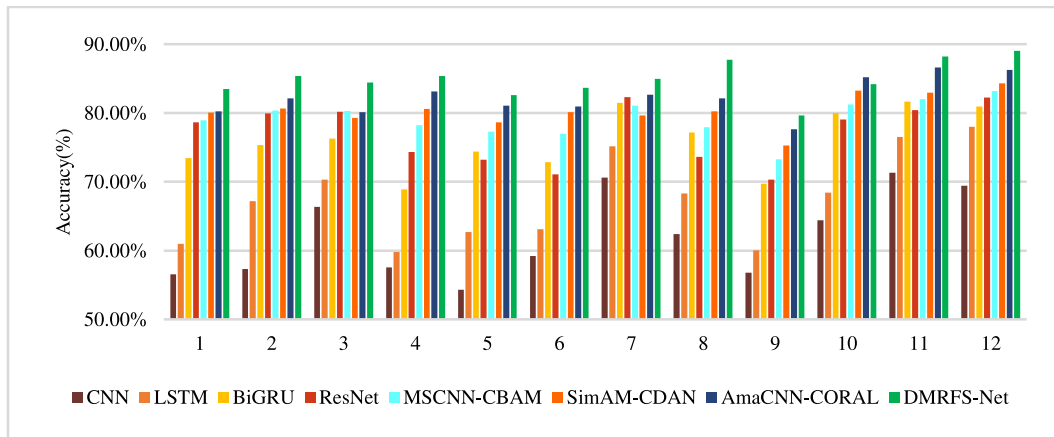


Fig. 24. The diagnostic accuracy under 5% training samples from the HUST dataset.

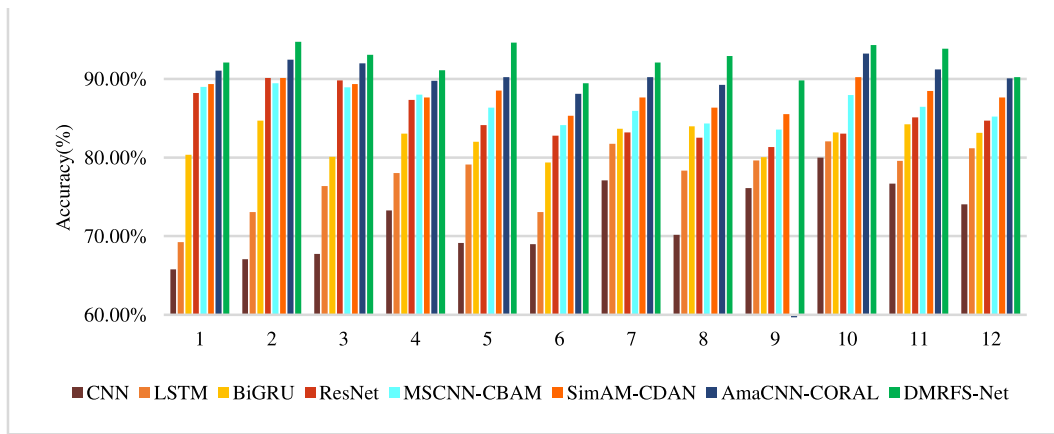


Fig. 25. The diagnostic accuracy under 10% training samples from the HUST dataset.

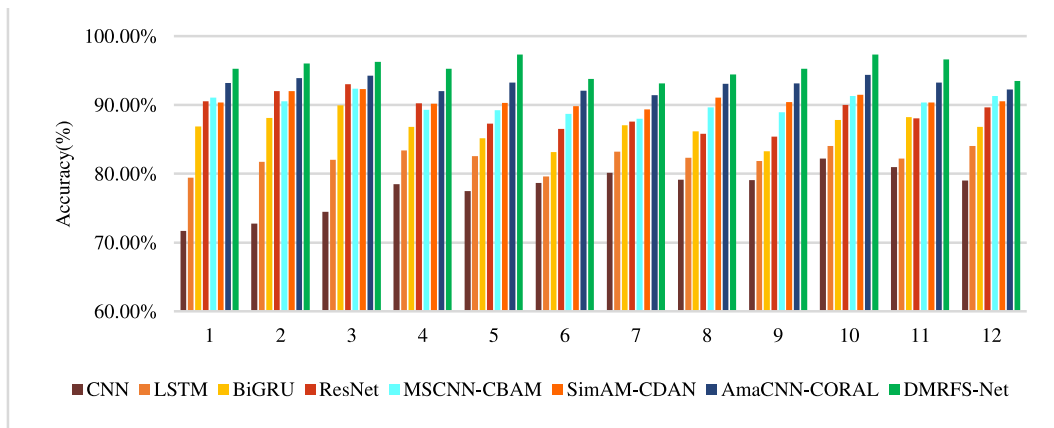


Fig. 26. The diagnostic accuracy under 15% training samples from the HUST dataset.

the proposed DMRFS-Net achieves an average diagnostic accuracy of 98.63 %, surpassing all baseline models. Specifically, compared to CNN, LSTM, BiGRU, and ResNet, DMRFS-Net exhibits accuracy improvements of 20.31 %, 14.30 %, 9.70 %, and 5.99 %, respectively. Additionally, relative to advanced domain adaptation models such as MSCNN-CBAM, SimAM-CDAN, and AmaCNN-CORAL, DMRFS-Net demonstrates accuracy gains of 5.34 %, 4.63 %, and 2.30 %, respectively. These results indicate that while all models achieve an accuracy exceeding 80 % with 20 % training samples, DMRFS-Net consistently delivers superior

diagnostic performance across varying operating conditions. This highlights its capacity to extract meaningful information from limited labeled data, ensuring high accuracy and reliability in fault mode recognition. Furthermore, DMRFS-Net not only maintains stable performance in small-sample scenarios but also consistently outperforms competing models across different working conditions, reinforcing its robust adaptability and generalization in cross-condition fault diagnosis.

Figs. 28 to 31 present the diagnostic comparison results of various models operating under cross-domain small sample conditions on the

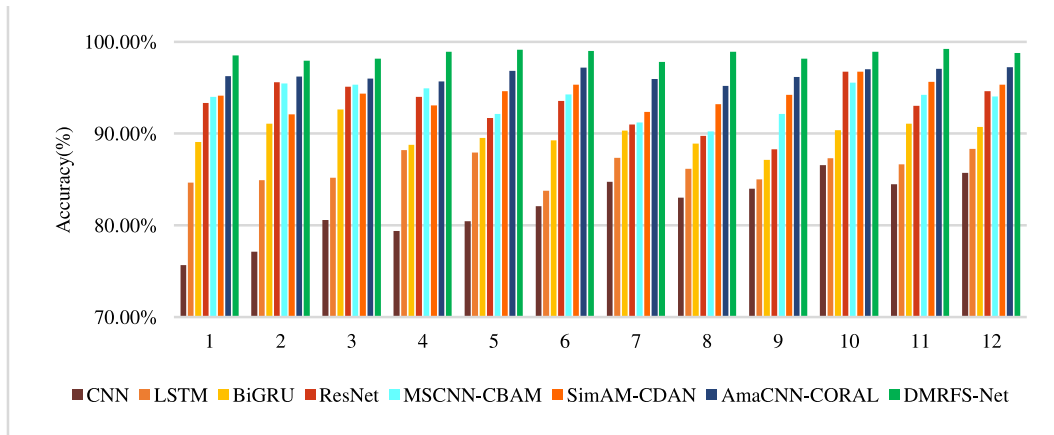


Fig. 27. The diagnostic accuracy under 20% training samples from the HUST dataset.

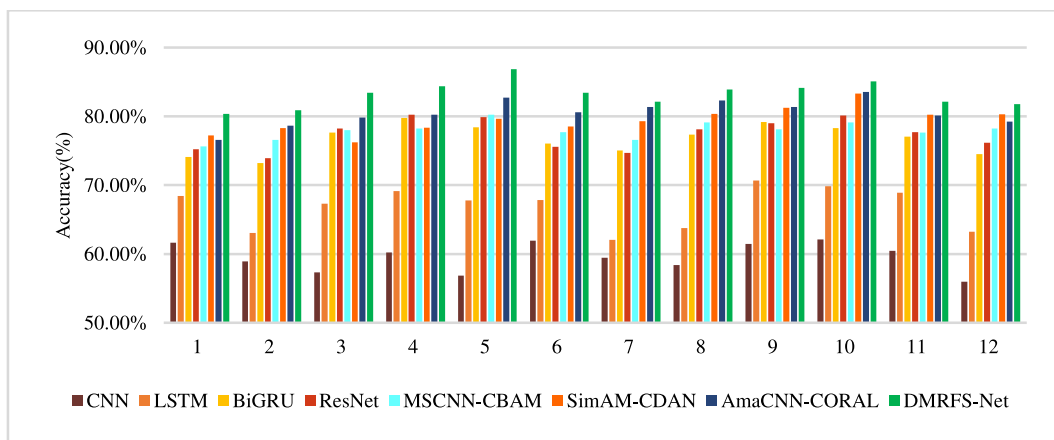


Fig. 28. The diagnostic accuracy under 5% training samples from the DIRG dataset.

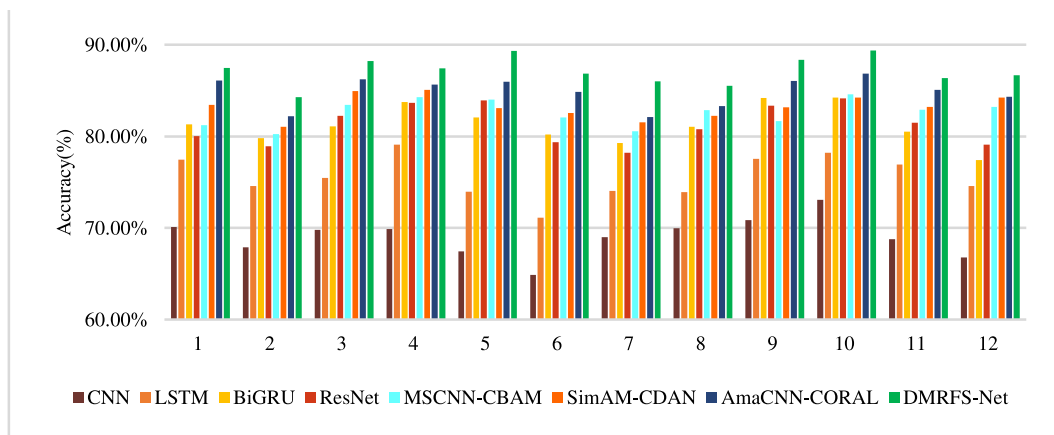


Fig. 29. The diagnostic accuracy under 10% training samples from the DIRG dataset.

DIRG dataset.

As shown in Fig. 28, when the training sample ratio is only 5 % on the DIRG dataset, the proposed DMRFS-Net achieves an impressive 83.21 % average diagnostic accuracy, significantly outperforming all compared models. Specifically, compared to CNN, LSTM, BiGRU, and ResNet, DMRFS-Net improves diagnostic accuracy by 39.71 %, 24.51 %, 8.46 %, and 7.48 %, respectively. Additionally, compared to MSCNN-CBAM, SimAM-CDAN, and AmaCNN-CORAL, it achieves accuracy gains of 6.79 %, 4.77 %, and 3.32 %, respectively. These results demonstrate that

even under extremely limited training data conditions, DMRFS-Net effectively learns fault features, exhibiting exceptional diagnostic capability and adaptability. Furthermore, as shown in Figs. 29 to 31, increasing the training sample ratio to 10 %, 15 %, and 20 % further enhances DMRFS-Net's diagnostic accuracy to 87.15 %, 96.03 %, and 98.83 %, consistently surpassing all baseline models. This trend reinforces the robustness and effectiveness of the proposed approach in cross-condition small-sample fault diagnosis. Notably, DMRFS-Net not only maximizes the utility of limited labeled data but also maintains

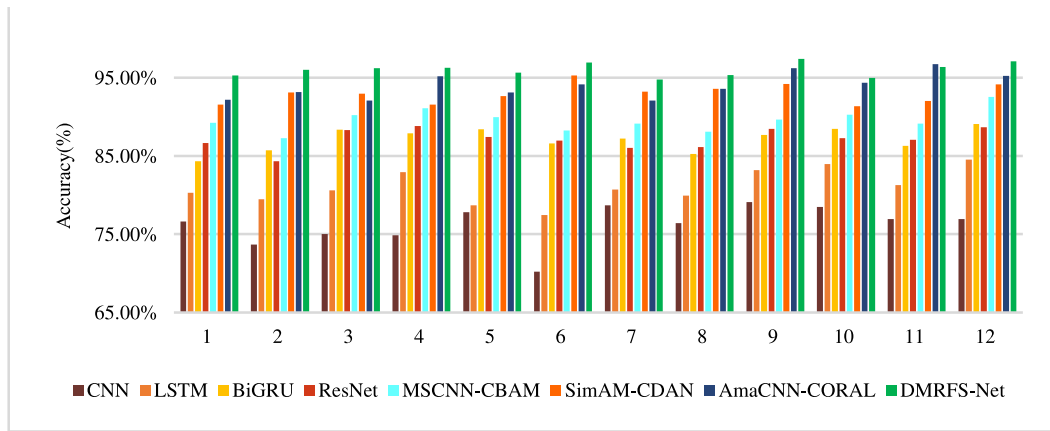


Fig. 30. The diagnostic accuracy under 15% training samples from the DIRG dataset.

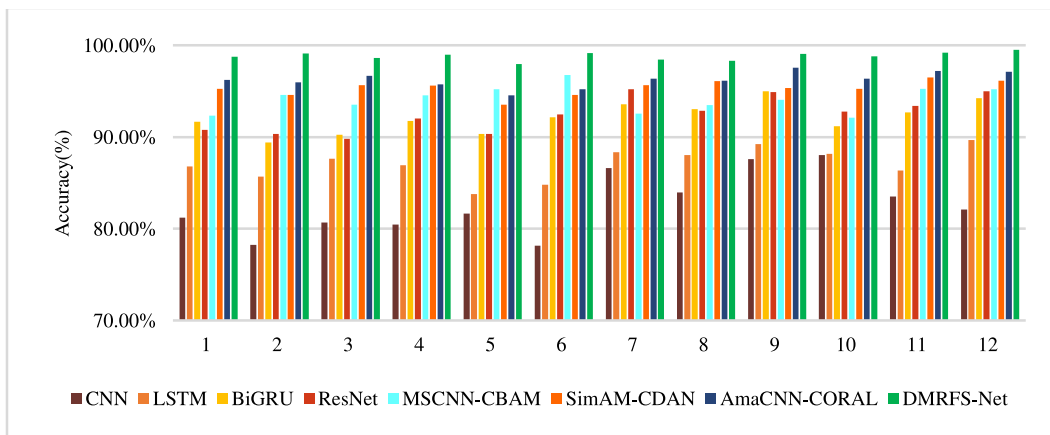


Fig. 31. The diagnostic accuracy under 20% training samples from the DIRG dataset.

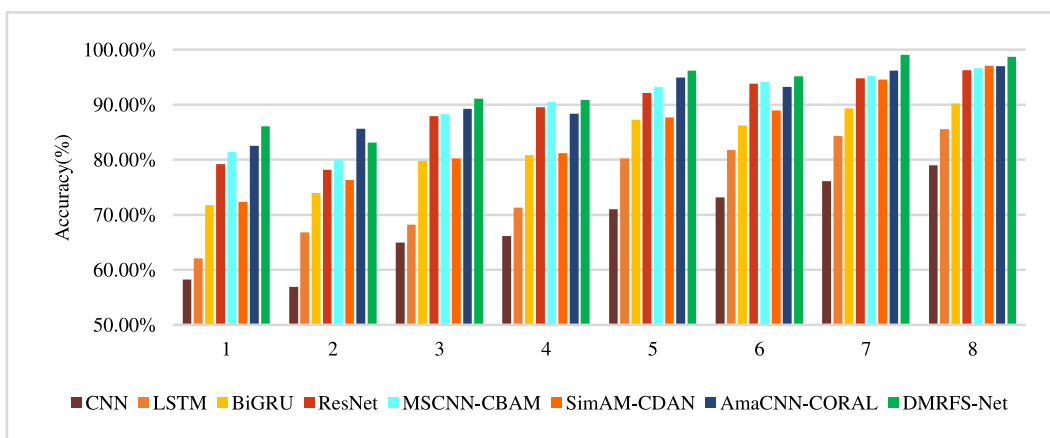


Fig. 32. Diagnostic accuracy of different models under varying sample conditions on the WTEP dataset.

strong generalization across different operating conditions and datasets, solidifying its superiority in small-sample cross-condition fault diagnosis.

The diagnostic comparison results of different models under cross-operating conditions in small-sample scenarios on the WTEP dataset are shown in Fig. 32.

The experimental results presented in Fig. 32 further validate the superior performance of DMRFS-Net across different training sample proportions. As the training sample ratio increases, the testing accuracy

of all models on the WTEP dataset exhibits an upward trend. However, under limited training data conditions, traditional models such as CNN and LSTM demonstrate significant performance limitations. Notably, at 5 % of training samples, their accuracy reached only 57.58 % and 64.44 %, respectively, indicating weak generalization capability. In contrast, BiGRU and ResNet show improved performance, achieving 89.79 % and 95.53 % accuracy at 20 % training samples, yet still falling short of optimal results. Models incorporating attention mechanisms and multi-scale feature fusion (e.g., MSCNN-CBAM, SimAM-CDAN) further

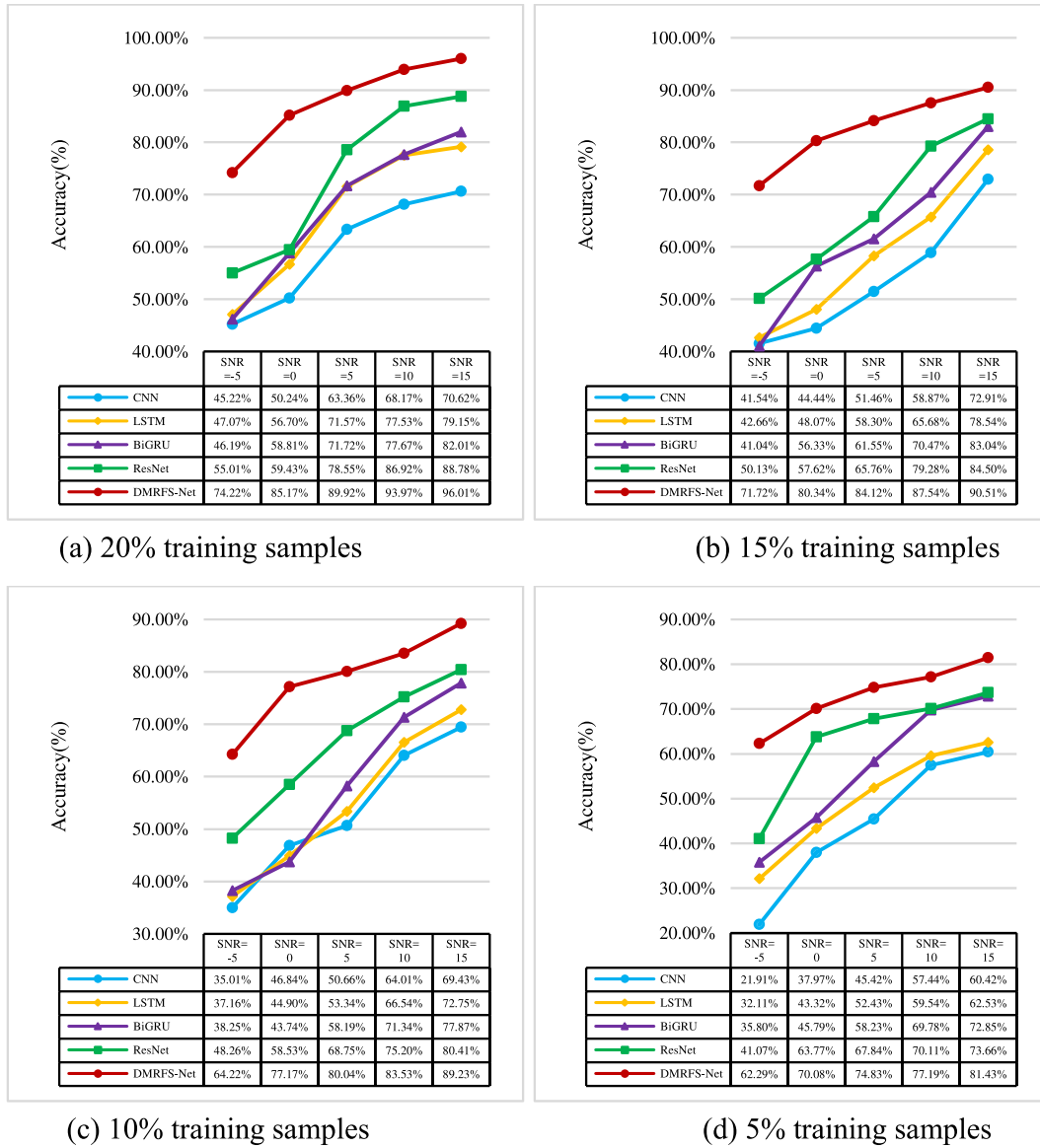


Fig. 33. Comparison of the classification accuracy of different models on the HUST data set under different noise conditions.

enhance diagnostic accuracy. Compared to CNN, the accuracy of MSCNN-CBAM, SimAM-CDAN, AmaCNN-CORAL, and DMRFS-Net improves by 66.62 %, 66.41 %, 67.78 %, and 71.76 %, respectively, at 20 % of training samples, demonstrating the effectiveness of feature extraction enhancements. Among them, DMRFS-Net achieves an average diagnostic accuracy of 98.90 %, surpassing all baseline models. Specifically, compared to MSCNN-CBAM, SimAM-CDAN, and AmaCNN-CORAL, DMRFS-Net achieves an additional accuracy gain of 3.09 %, 3.21 %, and 2.05 %, respectively, further confirming its superiority.

Moreover, even with only 5 % training samples, DMRFS-Net maintains an accuracy of 84.60 %, outperforming all other models, highlighting its strong feature learning ability and generalization capacity in low-data scenarios. Overall, DMRFS-Net not only maintains high accuracy under small-sample conditions but also continues to excel as more training data becomes available, demonstrating its robust adaptability and effectiveness in cross-condition small-sample fault diagnosis tasks.

3.4.3. Model anti-noise performance verification

To validate the noise resistance of the proposed model, Gaussian white noise is employed to simulate noise interference in actual industrial processes. Gaussian white noise with signal-to-noise ratios of -5

dB, 0 dB, 5 dB, 10 dB, and 15 dB is injected into the original datasets under various operating conditions. The noise resistance of different models is then assessed through various transfer tasks. Each transfer task is conducted five times at the same signal-to-noise ratio (SNR), with the training sample size set to 20 %. The average classification accuracy is taken as the final evaluation metric, and the results are presented in Figs. 33 and 34.

Figs. 33 and 34 illustrate that the predictive performance of all models in cross-condition tasks declines to varying extents with increasing noise intensity and decreasing training data volume. In Fig. 33(a), with the training set at 20 % in the HUST dataset, the average diagnostic accuracy of the models CNN, LSTM, BiGRU, and ResNet declines by 56.17 %, 68.15 %, 77.55 %, and 61.39 %, respectively, when the injected signal-to-noise ratio is -5 dB compared to 15 dB. Conversely, the proposed model DMRFS-Net experiences a more modest decline of 27.78 % under the same conditions, achieving an average diagnostic accuracy of 74.22 %, which is still considered acceptable. As the training sample size reduces from 20 % to 15 %, 10 %, and 5 %, DMRFS-Net's average diagnostic accuracy decreases by 3.49 %, 15.57 %, and 19.15 %, respectively, with each decline being less significant than those observed in the other four models. Fig. 34(d) illustrates that

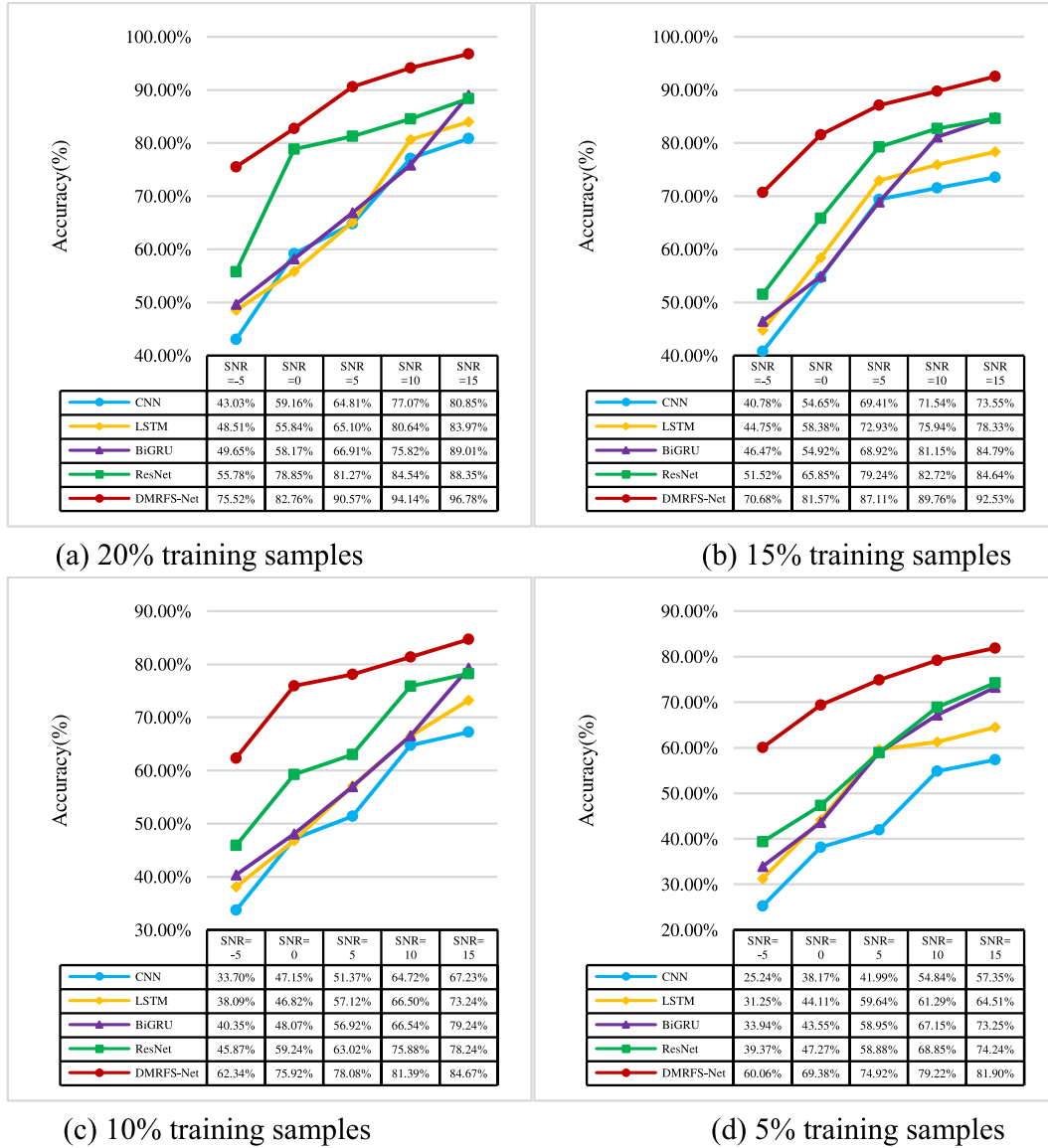


Fig. 34. Comparison of the classification accuracy of different models on the DIRG data set under different noise conditions.

in the DIRG dataset, with a training sample size set at 5 % and the introduction of noise with a signal-to-noise ratio of -5 dB, the proposed DMRFS-Net model achieves average diagnostic accuracy improvements of 57.98 %, 47.97 %, 43.49 %, and 34.45 % compared to the models CNN, LSTM, BiGRU, and ResNet, respectively. The experimental findings reveal that the proposed model maintains superior diagnostic accuracy across all SNR levels from -5 dB to 15 dB, and under the most challenging noise condition of -5 dB, it still outperforms the comparative models. This is primarily because the other three comparative models only accept vibration signals as input, while the proposed model utilizes dual-branch STFT and SWT image data, offering greater diversity and richness in input features compared to the other models. Additionally, the MCSAM and TFE modules in the proposed model effectively enhance both local and global critical feature information while suppressing unnecessary features and noise. Furthermore, the DCAFF module captures the interaction information between STFT and SWT image features more effectively, and combined with the DRF module, it facilitates thorough fusion and reusability. Moreover, the proposed model employs the multi-strategy enhanced MS-HHO algorithm for parameter optimization, whereas the comparative models may have suboptimal parameters, leading to lower diagnostic accuracy.

4. Conclusion

This paper introduces an efficient transfer fault diagnosis algorithm based on a parameter-optimized multi-scale redundant feature selection network. It aims to overcome challenges faced by traditional deep learning models, including excessive computational resource consumption, limited generalization capacity, and inadequate diagnostic accuracy resulting from feature redundancy. Furthermore, to address the practical limitations of conventional deep learning models in scenarios involving scarce samples and insufficient adaptability across various operating conditions, corresponding optimization strategies are proposed to improve model performance. The primary conclusions are as follows:

- (1) This paper utilizes STFT-SWT to encode one-dimensional signals into two-dimensional image data, which serves as the input for the dual-branch DMRFS-Net structure. This significantly enhances the richness of the features and the model's ability to extract key information. Experimental results show that the dual-branch DMRFS-Net model outperforms the single-branch DMRFS-Net model in terms of macro-average precision, with

- improvements of 7.98 %, 6.14 %, and 7.48 % across three datasets, respectively.
- (2) To reduce the impact of feature redundancy on model performance, this paper introduces a lightweight redundant feature selection module, which effectively reduces the model's parameter count and computational complexity while maintaining diagnostic accuracy. Additionally, the integration of the Transformer module enables precise capture of global contextual information, further enhancing the model's adaptability under complex operating conditions.
 - (3) In small-sample transfer tasks (with 20 % training samples), the proposed model achieved diagnostic accuracies of 98.63 %, 98.83 %, and 98.9 % on the three datasets, significantly outperforming other advanced comparison methods. Even under conditions with only 5 % of the training samples and the injection of -5dB noise, the model maintained high diagnostic accuracy on the three datasets (62.09 %, 60.06 %, and 61.22 %), demonstrating excellent generalization and noise resilience.
 - (4) To address the issues of local optima and slow convergence in the HHO algorithm, a chaotic backward learning strategy, adaptive dynamic exploration, and time-varying Cauchy strategy are employed to optimize the population search path, enhancing both global exploration and local search performance. Experimental results demonstrate that the improved algorithm significantly outperforms classical algorithms on standard benchmark functions.

While the proposed method performs well in feature selection, parameter optimization, and small-sample transfer diagnosis, it has certain limitations. (1) MSRFS may still preserve non-essential features in high-noise environments, affecting diagnostic precision. (2) The model focuses on varying conditions within a single device but does not account for cross-device transfer. Future research should investigate inter-device feature differences and develop effective transfer strategies to enhance adaptability. (3) The computational complexity remains a challenge in resource-limited settings, such as edge computing, necessitating further optimization. Moving forward, research will emphasize the practical deployment of this method in wind power generation and industrial applications. By integrating edge computing and real-time diagnostics, future efforts aim to improve deployment efficiency and facilitate intelligent maintenance in wind energy and smart manufacturing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61563032, 61963025, 62203196, 62463018) and the Gansu Provincial Science and Technology Program (22YF7GA164, 22CX8GA131, 22CX3GD005).

Data availability

Data will be made available on request.

References

An, Y., Zhang, K., Chai, Y., et al. (2023). Domain adaptation network base on contrastive learning for bearings fault diagnosis under variable working conditions. *Expert Systems with Applications*, 212, Article 118802. <https://doi.org/10.1016/j.eswa.2022.118802>

- Buchaiah, S., & Shakya, P. (2022). Bearing fault diagnosis and prognosis using data fusion based feature extraction and feature selection. *Measurement*, 188, Article 110506. <https://doi.org/10.1016/j.measurement.2021.110506>
- Chen, W., Sun, K., Li, X., et al. (2023). Adaptive multi-channel residual shrinkage networks for the diagnosis of multi-fault gearbox. *Applied Sciences*, 13(3), 1714. <https://doi.org/10.3390/app13031714>
- Chen, X., Yang, R., Xue, Y., et al. (2023). Deep transfer learning for bearing fault diagnosis: A systematic review since 2016. *IEEE Transactions on Instrumentation and Measurement*, 72, 1–21. <https://doi.org/10.1109/TIM.2023.3244237>
- Cui, L., Tian, X., Wei, Q., et al. (2024). A self-attention based contrastive learning method for bearing fault diagnosis. *Expert Systems with Applications*, 238, Article 121645. <https://doi.org/10.1016/j.eswa.2023.121645>
- Daga, A. P., Fasana, A., Marchesiello, S., et al. (2019). The Politecnico di Torino rolling bearing test rig: Description and analysis of open access data. *Mechanical Systems and Signal Processing*, 120, 252–273. <https://doi.org/10.1016/j.ymssp.2018.10.010>
- Dong, Y., Li, Y., Zheng, H., et al. (2022). A new dynamic model and transfer learning based intelligent fault diagnosis framework for rolling element bearings race faults: Solving the small sample problem. *ISA transactions*, 121, 327–348. <https://doi.org/10.1016/j.isatra.2021.03.042>
- Du, X., Jia, W., Yu, P., et al. (2022). A remaining useful life prediction method based on time-frequency images of the mechanical vibration signals. *Measurement*, 202, Article 111782. <https://doi.org/10.1016/j.measurement.2022.111782>
- Gao, W., Hu, J., Li, Y., et al. (2020). Feature redundancy based on interaction information for multi-label feature selection. *IEEE Access*, 8, 146050–146064. <https://doi.org/10.1109/ACCESS.2020.3015755>
- Gao, W., Hu, L., Li, Y., et al. (2021). Preserving similarity and staring decisio for feature selection. *IEEE Transactions on Artificial Intelligence*, 2(6), 584–593. <https://doi.org/10.1109/TAI.2021.3105084>
- Gong, B., An, A., Shi, Y., et al. (2024a). An interpretable hybrid spatiotemporal fusion method for ultra-short-term photovoltaic power prediction. *Energy*, 308, Article 132969. <https://doi.org/10.1016/j.energy.2024.132969>
- Gong, B., An, A., Shi, Y., et al. (2024b). Fault diagnosis of photovoltaic array with multi-module fusion under hyperparameter optimization. *Energy Conversion and Management*, 319, Article 118974. <https://doi.org/10.1016/j.enconman.2024.118974>
- Han, S., & Feng, Z. (2023). Deep residual joint transfer strategy for cross-condition fault diagnosis of rolling bearings. *Journal of Dynamics, Monitoring and Diagnostics*, 51–60. [Doi: 10.37965/jdmd.2023.147](https://doi.org/10.37965/jdmd.2023.147)
- He, Z., Shao, H., Ding, Z., et al. (2021). Modified deep autoencoder driven by multisource parameters for fault transfer prognosis of aeroengine. *IEEE Transactions on Industrial Electronics*, 69(1), 845–855. <https://doi.org/10.1109/TIE.2021.3050382>
- Heidari, A. A., Mirjalili, S., Faris, H., et al. (2019). Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems*, 97, 849–872. <https://doi.org/10.1016/j.future.2019.02.028>
- Hou, Y., Gao, H., Wang, Z., et al. (2022). Improved grey wolf optimization algorithm and application. *Sensors*, 22(10), 3810. <https://doi.org/10.3390/s22103810>
- Hu, Y., Li, J., Huang, Y., et al. (2019). Channel-wise and spatial feature modulation network for single image super-resolution. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(11), 3911–3927. <https://doi.org/10.1109/TCSVT.2019.2915238>
- Huang, Q., Liang, B., Dai, X., et al. (2024). Unmanned aerial vehicle fault diagnosis based on ensemble deep learning model. *Measurement Science and Technology*, 35(4), Article 046205. <https://doi.org/10.1088/1361-6501/ad2051>
- Jiang, Y., Shi, Z., Tang, C., et al. (2023). Cross-conditions fault diagnosis of rolling bearings based on dual domain adversarial network. *IEEE Transactions on Instrumentation and Measurement*. <https://doi.org/10.1109/TIM.2023.3322485>
- Jin, Y., Hou, L., & Chen, Y. (2022). A time series Transformer based method for the rotating machinery fault diagnosis. *Neurocomputing*, 494, 379–395. <https://doi.org/10.1016/j.neucom.2022.04.111>
- Kumar, R., & Anand, R. S. (2024). Bearing fault diagnosis using multiple feature selection algorithms with SVM. *Progress in Artificial Intelligence*, 13(2), 119–133. <https://doi.org/10.1007/s13748-024-00324-1>
- Kumbhar, S. G., Desavale, R. G., & Dharwadkar, N. V. (2021). Fault size diagnosis of rolling element bearing using artificial neural network and dimension theory. *Neural Computing and Applications*, 33(23), 16079–16093. <https://doi.org/10.1007/s00521-021-06228-8>
- Li, C., Li, S., Wang, H., et al. (2023). Attention-based deep meta-transfer learning for few-shot fine-grained fault diagnosis. *Knowledge-Based Systems*, 264, Article 110345. <https://doi.org/10.1016/j.knsys.2023.110345>
- Li J, Wen Y, He L. Sconv: Spatial and channel reconstruction convolution for feature redundancy[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 6153–6162. [Doi: 10.1109/CVPR52729.2023.00596](https://doi.org/10.1109/CVPR52729.2023.00596)
- Li, S., Li, T., Sun, C., et al. (2023). Multilayer Grad-CAM: An effective tool towards explainable deep neural networks for intelligent fault diagnosis. *Journal of Manufacturing Systems*, 69, 20–30. <https://doi.org/10.1016/j.jmsy.2023.05.027>
- Li, Y., Wang, T., Xie, J., et al. (2024). A simulation data-driven semi-supervised framework based on MK-KNN graph and ESSGAT for bearing fault diagnosis. *ISA transactions*. <https://doi.org/10.1016/j.isatra.2024.09.029>
- Liang, X., Yao, J., Zhang, W., et al. (2023). A novel fault diagnosis of a rolling bearing method based on variational mode decomposition and an artificial neural network. *Applied Sciences*, 13(6), 3413. <https://doi.org/10.3390/app13063413>
- Liao, J. X., Dong, H. C., Sun, Z. Q., et al. (2023). Attention-embedded quadratic network (qttnet) for effective and interpretable bearing fault diagnosis. *IEEE Transactions on Instrumentation and Measurement*, 72, 1–13. <https://doi.org/10.1109/TIM.2023.3259031>

- Liu, D., Cui, L., Wang, G., et al. (2024). Interpretable domain adaptation transformer: A transfer learning method for fault diagnosis of rotating machinery. *Structural Health Monitoring*, 14759217241249656. <https://doi.org/10.1177/14759217241249656>
- Liu, W., Zhang, Z., Zhang, J., et al. (2023). A novel fault diagnosis method of rolling bearings combining convolutional neural network and transformer. *Electronics*, 12 (8), 1838. <https://doi.org/10.3390/electronics12081838>
- Lu, J., Qian, W., Li, S., et al. (2021). Enhanced K-nearest neighbor for intelligent fault diagnosis of rotating machinery. *Applied Sciences*, 11(3), 919. <https://doi.org/10.3390/app11030919>
- Lv, H., Chen, J., Pan, T., et al. (2022). Attention mechanism in intelligent fault diagnosis of machinery: A review of technique and application. *Measurement*, 199, Article 111594. <https://doi.org/10.1016/j.measurement.2022.111594>
- Ni, Q., Ji, J. C., Feng, K., et al. (2022). A fault information-guided variational mode decomposition (FIVMD) method for rolling element bearings diagnosis. *Mechanical Systems and Signal Processing*, 164, Article 108216. <https://doi.org/10.1016/j.ymssp.2021.108216>
- Qin, S., Tao, J., & Zhao, Z. (2023). Fault diagnosis of wind turbine pitch system based on LSTM with multi-channel attention mechanism. *Energy Reports*, 10, 4087–4096. <https://doi.org/10.1016/j.egyr.2023.10.076>
- Shallue, C. J., Lee, J., Antognini, J., et al. (2019). Measuring the effects of data parallelism on neural network training. *Journal of Machine Learning Research*, 20 (112), 1–49. <https://doi.org/10.48550/arXiv.1811.03600>
- Shan, S., Liu, J., Wu, S., et al. (2023). A motor bearing fault voiceprint recognition method based on Mel-CNN model. *Measurement*, 207, Article 112408. <https://doi.org/10.1016/j.measurement.2022.112408>
- Shao, H., Li, W., Cai, B., Wan, J., Xiao, Y., & Yan, S. (2023). Dual-threshold attention-guided GAN and limited infrared thermal images for rotating machinery fault diagnosis under speed fluctuation. *IEEE Transactions on Industrial Informatics*, 19(9), 9933–9942. <https://doi.org/10.1109/TII.2022.3232766>
- Shao, K., Fu, W., Tan, J., et al. (2021). Coordinated approach fusing time-shift multiscale dispersion entropy and vibrational Harris hawks optimization-based SVM for fault diagnosis of rolling bearing. *Measurement*, 173, Article 108580. <https://doi.org/10.1016/j.measurement.2020.108580>
- Shao, X., & Kim, C. S. (2024). Adaptive multi-scale attention convolution neural network for cross-domain fault diagnosis. *Expert Systems with Applications*, 236, Article 121216. <https://doi.org/10.1016/j.eswa.2023.121216>
- Shen, J., Zhang, Y., Zhu, H., et al. (2023). A novel transfer learning network with adaptive input length selection and lightweight structure for bearing fault diagnosis. *Engineering Applications of Artificial Intelligence*, 123, Article 106395. <https://doi.org/10.1016/j.engappai.2023.106395>
- Takashima, T., Yamaguchi, J., & Ishida, M. (2010). Disconnection detection using earth capacitance measurement in photovoltaic module string. *Prog Photovolt Res Applications*, 16(8), 669–677. <https://doi.org/10.1002/pip.860>
- Tang, G., Yi, C., Liu, L., et al. (2023). A novel transfer learning network with adaptive input length selection and lightweight structure for bearing fault diagnosis. *Engineering Applications of Artificial Intelligence*, 123, Article 106395. <https://doi.org/10.1016/j.engappai.2023.106395>
- Tang, Z., Hou, X., Huang, X., et al. (2024). Domain adaptation for bearing fault diagnosis based on SimAM and adaptive weighting strategy. *Sensors*, 24(13), 4251. <https://doi.org/10.3390/s24134251>
- Tian, H., Fan, H., Feng, M., et al. (2023). Fault diagnosis of rolling bearing based on HSP0 algorithm optimized CNN-LSTM neural network. *Sensors*, 23(14), 6508. <https://doi.org/10.3390/s23146508>
- Umar, S. U., Rashid, T. A., Ahmed, A. M., et al. (2024). Modified Bat Algorithm: A newly proposed approach for solving complex and real-world problems. *Soft Computing*, 28 (13), 7983–7998. <https://doi.org/10.1007/s00500-024-09761-5>
- Wan, H., Liu, M., Zhu, H., et al. (2024). An interpretable bearing fault diagnosis method based on belief rule base with interval structured. *IEEE Access*, 12, 116939–116955. <https://doi.org/10.1109/ACCESS.2024.3435057>
- Wang, B., Qiu, W., Hu, X., et al. (2024). A rolling bearing fault diagnosis technique based on recurrence quantification analysis and Bayesian optimization SVM. *Applied Soft Computing*, 156, Article 111506. <https://doi.org/10.1016/j.asoc.2024.111506>
- Wang, H., Liu, Z., Peng, D., et al. (2023). Interpretable convolutional neural network with multilayer wavelet for Noise-Robust Machinery fault diagnosis. *Mechanical Systems and Signal Processing*, 195, Article 110314. <https://doi.org/10.1016/j.ymssp.2023.110314>
- Wang, H., Sun, W., He, L., et al. (2022). Rolling bearing fault diagnosis using multi-sensor data fusion based on 1d-cnn model. *Entropy*, 24(5), 573. <https://doi.org/10.3390/e24050573>
- Wang, J., Du, G., Zhu, Z., et al. (2020). Fault diagnosis of rotating machines based on the EMD manifold. *Mechanical Systems and Signal Processing*, 135, Article 106443. <https://doi.org/10.1016/j.ymssp.2019.106443>
- Wang, Y. Q., & Zhao, Y. P. (2023). A novel inter-domain attention-based adversarial network for aero-engine partial unsupervised cross-domain fault diagnosis. *Engineering Applications of Artificial Intelligence*, 123, Article 106486. <https://doi.org/10.1016/j.engappai.2023.106486>
- Wang, Z., Yao, L., Chen, G., et al. (2021). Modified multiscale weighted permutation entropy and optimized support vector machine method for rolling bearing fault diagnosis with complex signals. *ISA Transactions*, 114, 470–484. <https://doi.org/10.1016/j.isatra.2020.12.054>
- Wei, M., Yang, J., Yao, D., et al. (2022). Fault diagnosis of bearings in multiple working conditions based on adaptive time-varying parameters short-time Fourier synchronous squeeze transform. *Measurement Science and Technology*, 33(12), Article 124002. <https://doi.org/10.1088/1361-6501/ac8abf>
- Woo, S., Park, J., Lee, J. Y., et al. Cbam: Convolutional block attention module[C]// Proceedings of the European conference on computer vision (ECCV). 2018: 3-19. Doi: 10.1007/978-3-030-01234-2_1.
- Wu, Z., Jiang, H., Liu, S., et al. (2023). Conditional distribution-guided adversarial transfer learning network with multi-source domains for rolling bearing fault diagnosis. *Advanced Engineering Informatics*, 56, Article 101993. <https://doi.org/10.1016/j.aei.2023.101993>
- Yan, X., Xu, Y., She, D., et al. (2021). Reliable fault diagnosis of bearings using an optimized stacked variational denoising auto-encoder. *Entropy*, 24(1), 36. <https://doi.org/10.3390/e24010036>
- Yang, G., Wang, Y., Qin, D., et al. (2024). HMM-Based method for aircraft environmental control system turbofan rolling bearing fault diagnosis. *Shock and Vibration*, 2024(1), Article 5582169. <https://doi.org/10.1155/2024/5582169>
- Zhang, K., Deng, Y., Chen, P., et al. (2022). Quaternion empirical wavelet transform and its applications in rolling bearing fault diagnosis. *Measurement*, 195, Article 111179. <https://doi.org/10.1016/j.measurement.2022.111179>
- Zhang, L., Zhang, H., & Cai, G. (2022). The multiclass fault diagnosis of wind turbine bearing based on multisource signal fusion and deep learning generative model. *IEEE Transactions on Instrumentation and Measurement*, 71, 1–12. <https://doi.org/10.1109/TIM.2022.3178483>
- Zhang, P., Gao, W., Hu, J., et al. (2021). A conditional-weight joint relevance metric for feature relevancy term. *Engineering Applications of Artificial Intelligence*, 106, Article 104481. <https://doi.org/10.1016/j.engappai.2021.104481>
- Zhang, P., & Gao, W. (2021). Feature relevance term variation for multi-label feature selection. *Applied Intelligence*, 51, 5095–5110. <https://doi.org/10.1007/s10489-020-02129-w>
- Zhang, R., Zhao, Z., Tao, X., et al. (2024). Fault diagnosis method for bearings under variable working conditions based on transfer relation network. *Measurement Science and Technology*, 35(12), Article 126202. <https://doi.org/10.1088/1361-6501/ad7485>
- Zhang, X., He, C., Lu, Y., et al. (2022). Fault diagnosis for small samples based on attention mechanism. *Measurement*, 187, Article 110242. <https://doi.org/10.1016/j.measurement.2021.110242>
- Zhang, Y., Ji, J. C., Ren, Z., et al. (2023). Digital twin-driven partial domain adaptation network for intelligent fault diagnosis of rolling bearing. *Reliability Engineering & System Safety*, 234, Article 109186. <https://doi.org/10.1016/j.res.2023.109186>
- Zhao, B., Yang, D., Karimi, H. R., et al. (2023). Filter-wrapper combined feature selection and adaboost-weighted broad learning system for transformer fault diagnosis under imbalanced samples. *Neurocomputing*, 560, Article 126803. <https://doi.org/10.1016/j.neucom.2023.126803>
- Zhao, C., Zio, E., & Shen, W. (2024). Domain generalization for cross-domain fault diagnosis: An application-oriented perspective and a benchmark study. *Reliability Engineering & System Safety*, 109964. <https://doi.org/10.1016/j.res.2024.109964>
- Zhao, J., Zhang, D., He, Q., et al. (2023). A hybrid-strategy-improved dragonfly algorithm for the parameter identification of an SDM. *Sustainability*, 15(15), 11791. <https://doi.org/10.3390/su151511791>
- Zhao, Y., Song, Z., Li, D., et al. (2024). Wind Turbine Gearbox Fault Diagnosis Based on Multi-Sensor Signals Fusion. *Protection and Control of Modern Power Systems*, 9(4), 96–109. <https://doi.org/10.23919/PCMP.2023.000241>
- Zhong, H., Lv, Y., Yuan, R., et al. (2022). Bearing fault diagnosis using transfer learning and self-attention ensemble lightweight convolutional neural network. *Neurocomputing*, 501, 765–777. <https://doi.org/10.1016/j.neucom.2022.06.066>
- Zhong, H., Yu, S., Trinh, H., et al. (2023). Fine-tuning transfer learning based on DCGAN integrated with self-attention and spectral normalization for bearing fault diagnosis. *Measurement*, 210, Article 112421. <https://doi.org/10.1016/j.measurement.2022.112421>

Research on bearing remaining useful life anti-noise prediction based on fusion of color-grayscale time-frequency features

Wenchao Jia^{ID}, Aimin An, Xianjun Du^{*}, Yaoke Shi^{ID} and Bin Gong^{ID}

College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou 730050, People's Republic of China

E-mail: xdu@lut.edu.cn

Received 29 January 2024, revised 29 April 2024

Accepted for publication 23 May 2024

Published 3 June 2024



Abstract

In contemporary industrial processes, vibration signals collected from bearings often contain significant noise, challenging the efficacy of conventional predictive models in extracting critical degradation features and accurately predicting the remaining useful life (RUL) of bearings. Addressing these challenges, this paper introduces a novel method for predicting bearing RUL under noisy conditions, leveraging a dual-branch multi-scale convolutional attention network (DMCSA) integrated with a dense residual feature fusion network (DRF). Initially, the method applies continuous wavelet transform (CWT) to vibration signals to extract color time-frequency image data, followed by grayscale processing to construct a comprehensive color-grayscale time-frequency image dataset, thereby augmenting the model's input features. Enhanced channel and spatial attention mechanisms, combined with multi-scale convolutions, facilitate superior feature extraction and selection. The model's resilience to noise is fortified by incorporating noise into the training dataset. Subsequently, selected color-gray time-frequency features undergo fusion and relearning through the DRF framework at the model's backend. The crayfish optimization algorithm (COA) is deployed for the astute determination of the model's critical hyperparameters. The proposed DMCSA-DRF model is then applied to predict the health indicator (MSCA-DRF-HI) of the test dataset, culminating in the accurate prediction of the bearings' RUL. Validation experiments demonstrate that our method surpasses comparative models in prediction accuracy under diverse noise interferences, signifying a substantial advancement in predictive performance.

Keywords: remaining useful life, multi-scale convolutional, attention enhancement mechanism, dense residual feature fusion, color-grayscale features

1. Introduction

1.1. Status of research and challenges in RUL prediction

The prediction of RUL of equipment is at the core of prognostics and health management (PHM), serving as a

prerequisite for decision-making in equipment health management. Predicting the RUL of equipment enables the early estimation of potential failure occurrences, thereby mitigating catastrophic events resulting from equipment failures, reducing unnecessary maintenance costs, and enhancing equipment reliability, availability, and safety.

Currently, mainstream prediction methods are primarily divided into two categories: physics-based methods and data-driven methods, as illustrated in table 1. Physics-based

^{*} Author to whom any correspondence should be addressed.

Table 1. Research status of RUL prediction methods.

RUL prediction models	Research methods	Relevant literature
Physical models	Damage accumulation theory-based prediction methods	[9, 10]
	Material mechanics-based prediction methods	[11–13]
Data-driven models		
Statistical model-based prediction methods	Regression models	[14–16]
	Markov models	[17, 18]
	Proportional hazard models	[19, 20]
Stochastic process-based prediction methods	Wiener process	[21, 22]
	Gamma process	[23, 24]
Machine learning-based prediction methods	Artificial neural networks	[25, 26]
	Support vector machines	[27, 28]
	Bayesian	[29, 30]
	Deep learning	[31, 32]

methods rely on the degradation or failure mechanisms of the equipment to establish mathematical expressions describing the degradation process, thereby achieving RUL prediction through these expressions [1–3]. However, as systems continue to scale up, structures become increasingly complex, and inter-device coupling strengthens, establishing precise physical models tailored to specific critical equipment poses significant challenges.

Data-driven methodologies harness sensor monitoring data to gauge equipment health status [4–6], predominantly falling under the purview of statistical models, stochastic processes, and machine learning techniques. Statistical models and stochastic processes leverage prior experience and knowledge to fit sensor data and prognosticate equipment RUL [7, 8]. However, as system complexity escalates and precision becomes more critical, the applicability of methods grounded in statistical models and stochastic processes dwindles, constraining their utility in RUL prediction.

Machine learning approaches, on the other hand, directly delineate the underlying nonlinear mapping relationships between historical data without the constraints encountered in stochastic process methodologies. Zhang *et al* [33] employed artificial neural networks (ANN) to achieve precise RUL prediction for lithium-ion batteries. Ordonez *et al* [34] introduced a RUL prediction methodology for aircraft engines predicated on support vector regression (SVR) models, substantiating its efficacy through experimental validation. Similarly, Bai *et al* [35] proposed a bearing RUL prediction approach anchored in Bayesian neural networks. Despite the commendable results attained by shallow machine learning methods in RUL prediction, the burgeoning volume of equipment monitoring data poses challenges in capturing critical equipment state information from high-dimensional, intricate historical data.

Deep learning techniques have shown extensive prospects in the field of RUL prediction by uncovering complex nonlinear relationships from large-scale high-dimensional data. In the task of predicting the RUL based on time series, the long short-term memory (LSTM) and gated recurrent unit (GRU) networks have shown significant results due to their ability to extract temporal information [36–38]. Additionally, convolutional neural network (CNN) has emerged as a primary focus due to its formidable ability to extract features [39, 40]. Yang *et al* [41] introduced a novel bearing RUL prediction model featuring a one-dimensional dual convolutional structure, enabling both online training and offline testing for bearing RUL. Pursuing exhaustive feature extraction, the amalgamation of CNNs for spatial feature extraction with LSTM for temporal feature extraction has spawned CNN-LSTM models [42], adept at extracting spatio-temporal features and enhancing RUL prediction accuracy. Although the above-mentioned methods have achieved remarkable results in the field of RUL prediction based on time series data. However, the degradation signals of real bearings are non-stationary and transient and contain a large amount of noise, how to effectively extract key degradation features under a large amount of strong noise background is still a key challenge for the RUL prediction task. This also forces researchers to further explore models with better feature extraction performance and more effective signal analysis methods.

1.2. Summary of existing RUL methods and related work in this paper

In addressing the RUL prediction problem under noisy conditions, Wang and Wang [43] developed a CNN-based semi-supervised deep learning model for RUL prediction, utilizing a fully convolutional denoising autoencoder (FCDAE) to distill both global and local features from data. Considering the problem of uncertainty in the RUL prediction process caused by noise, Liu *et al* [44] proposed a semi-supervised learning-based method for uncertainty quantification and interval prediction of equipment RUL. Xiao *et al* [45, 46] introduced a noise-enhanced RUL prediction method, augmenting model robustness by integrating noise into training samples, with validation on the C-MAPSS aero-engine dataset. Zhao *et al* [47] proposed a gated attention mechanism capsule neural network (GAM-CapsNet), assigning weights to features to bolster noise resilience. Wang *et al* [48] proposed an RUL prediction method based on the convolutional attention mechanism and temporal convolution, which used an effective adaptive shrinkage model (EAS) to compute the thresholds corresponding to the input features, and screening out the redundant and noisy features by the threshold function.

Additionally, the construction of accurate health indicators (HI) remains a pivotal challenge in RUL prediction. Traditional methods employing first-order [41], second-order, and piecewise degradation functions [47] to represent HI have exhibited limitations, often failing to accurately reflect equipment's specific health status or degradation level at given times, potentially obscuring inherent degradation traits. Qin *et al* [49] devised a HI construction method using a

degradation-trend-constrained variational autoencoder (DTC-VAE), enabling the adaptive generation of HI under unsupervised conditions. Zhou *et al* [50] developed an HI construction method based on the distribution contact ratio metric (DCRHI), facilitating the standardized determination of HI thresholds across various operational conditions. The RUL prediction methods under noisy conditions and the HI construction methods mentioned above rely solely on the extraction of time-domain features from vibration signals, thus overlooking the rich potential of frequency-domain feature information embedded within the signals. Moreover, these models fail to address the inherent limitations associated with manually tuning model parameters.

In response to the challenges of extracting critical degradation features amidst noise and formulating suitable HI, this paper introduces a feature extraction model characterized by a dual-branch structure that incorporates grayscale-color time-frequency features, multi-scale channel spatial attention, and dense residual structures. To augment model robustness, noise is intentionally introduced during training. The HI construction process attentively considers the bearings' actual degradation trends and monotonicity, leading to a more accurate depiction of the bearing degradation process. The paper proposes the following major contributions:

- (1) An enhanced feature extraction framework employing multi-scale channel and spatial attention mechanisms to diminish redundant and noisy data while accentuating essential degradation features.
- (2) The employment of a monotonicity function for screening one-dimensional time-frequency vibration signal features during the model's training phase, facilitating the accurate construction of HI and the subsequent extraction of MCSA-DRF-HI for truncated test bearings to predict their RUL.
- (3) To enhance the diversity and richness of the input features, this study introduces a novel RUL prediction model, the dual-branch multi-scale convolutional attention network with dense residual feature fusion (DMCSA-DRF), which utilizes color-grey scale time-frequency images. This dual-branch structure is specifically designed to process and enrich the feature space. To increase the model's resilience against environmental noise, three distinct types of noise are systematically introduced during the training phase, thereby improving the model's robustness. Additionally, in response to the challenges posed by manual hyperparameter tuning—which can be both time-consuming and imprecise—the COA is implemented for optimizing the model's hyperparameters.

2. Method

2.1. CNN

The CNN conducts a weighted operation on the input data using a convolutional kernel, transmitting essential features of the data sequentially to the model's backend. The convolution operation is formulated as follows:

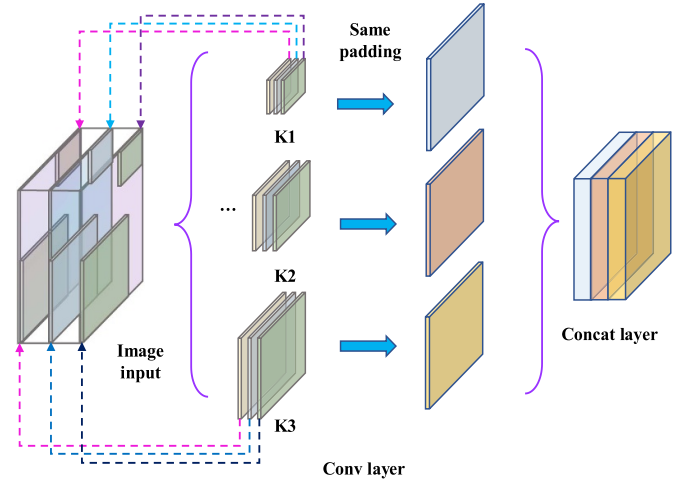


Figure 1. Multi-scale convolutional layer structure.

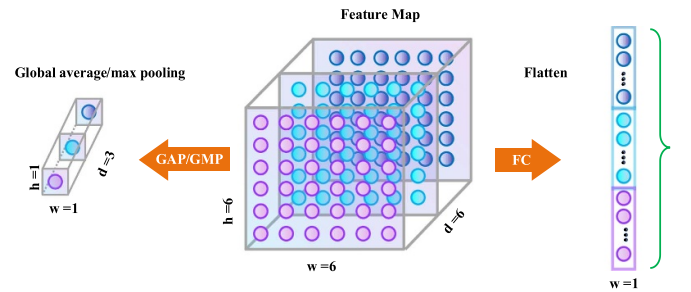


Figure 2. Comparison between FC layer and GAP/GMP layer.

$$y(k) = f \left(\sum_{i=1}^m \sum_{j=1}^m a_{i,j} \otimes w_{i,j}^k + b^k \right) \quad (1)$$

where f is the activation function, $a_{i,j}$ represents the value at position (i,j) in the feature map, $w_{i,j}^k$ represents the weight at (i,j) in the k th convolution kernel, and b^k represents the bias corresponding to the k th convolution kernel.

2.1.1. Multi-scale convolutional structures. The multi-scale convolutional feature extraction structure illustrated in figure 1, employs convolutional kernels of varying sizes $K1$, $K2$, and $K3$, denoted as $F_{K1}^{C_{K1} \times H \times W}$, $F_{K2}^{C_{K2} \times H \times W}$, and $F_{K3}^{C_{K3} \times H \times W}$, respectively. The output of the Concat layer is defined as:

$$F_{\text{Concat}} = \left[F_{K1}^{C_{K1} \times H \times W}, F_{K2}^{C_{K2} \times H \times W}, F_{K3}^{C_{K3} \times H \times W} \right] \\ = F^{(C_{K1} + C_{K2} + C_{K3}) \times H \times W} \quad (2)$$

where C_{K1} , C_{K2} , C_{K3} are the number of output channels corresponding to the convolution kernels $K1$, $K2$, and $K3$, respectively.

2.1.2. Global pooling. Figure 2 illustrates the comparison between the fully connected (FC) layer and the global average pooling (GAP)/global max pooling (GMP) layer.

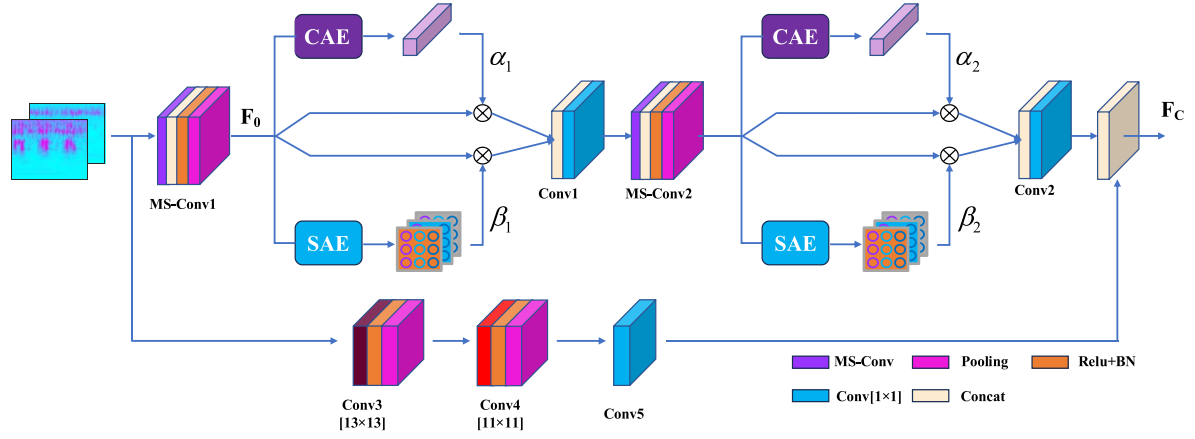


Figure 3. MCSA-Net architecture diagram.

GMP identifies the maximum value among all elements in each feature map, retaining locally significant features. Conversely, GAP calculates the average value of all elements in each feature map, capturing overall feature information. The global pooling layer reduces model parameters, mitigates overfitting, and extracts essential global features.

2.2. Crayfish optimization algorithm

The COA [51] emulates the heat avoidance, competitive, and foraging behaviors exhibited by crayfish. These three behaviors correspond to distinct phases in the algorithm, namely, heat avoidance, competitive, and foraging phases. The algorithm employs temperature as a control parameter, effectively balancing the exploratory and exploitative performance of the algorithm. The pseudo-code of COA is as follows:

Algorithm 1. Crayfish optimization algorithm pseudo-code.

```

Initialization population  $N$ , iterations  $T$ , dimension  $dim$ 
Randomly generate an initial population
Calculate the fitness value of the population to get  $X_G, X_L$ 
While  $t < T$ 
    Defining temperature  $temp$  by Eq:  $temp = rand \times 15 + 20$ 
    If  $temp > 30$ 
        (Summer resort stage)
        Define cave  $X_{shade}$  according to Eq:  $X_{shade} = (X_G + X_L) / 2$ 
        If  $rand < 0.5$ 
            Crayfish conducts the summer resort stage according to Eq:  $X_{i,j}^{t+1} = X_{i,j}^t + C_2 \times rand \times (X_{shade} - X_{i,j}^t)$ 
        (Competition stage)
        Else
            Crayfish compete for cave through Eq:  $X_{i,j}^{t+1} = X_{i,j}^t - X_{z,j}^t + X_{shade}$ 
        End
    Else
        (Foraging stage)
        The food intake  $p$  and food size  $Q$  are obtained by Eq:
         $p = C_1 \times \left( \frac{1}{\sqrt{2\pi}\sigma} \times \exp\left(-\frac{(temp-\mu)^2}{2\sigma^2}\right) \right)$  and  $Q = C_3 \times rand \times (fitness_i / fitness_{food})$ 
        If  $Q > 2$ 
            Crayfish shreds food by Eq:  $X_{food} = \exp\left(-\frac{1}{Q}\right) \times X_{food}$ 
            Crayfish foraging according to Eq:
             $X_{i,j}^{t+1} = X_{i,j}^t + X_{food} \times p \times (\cos(2\pi \times rand) - \sin(2\pi \times rand))$ 
        Else
            Crayfish foraging according to Eq:  $X_{i,j}^{t+1} = (X_{i,j}^t - X_{food}) \times p + p \times rand \times X_{i,j}^t$ 
        End
    End
    Update fitness values,  $X_G, X_L$ 
     $t = t + 1$ 
End

```

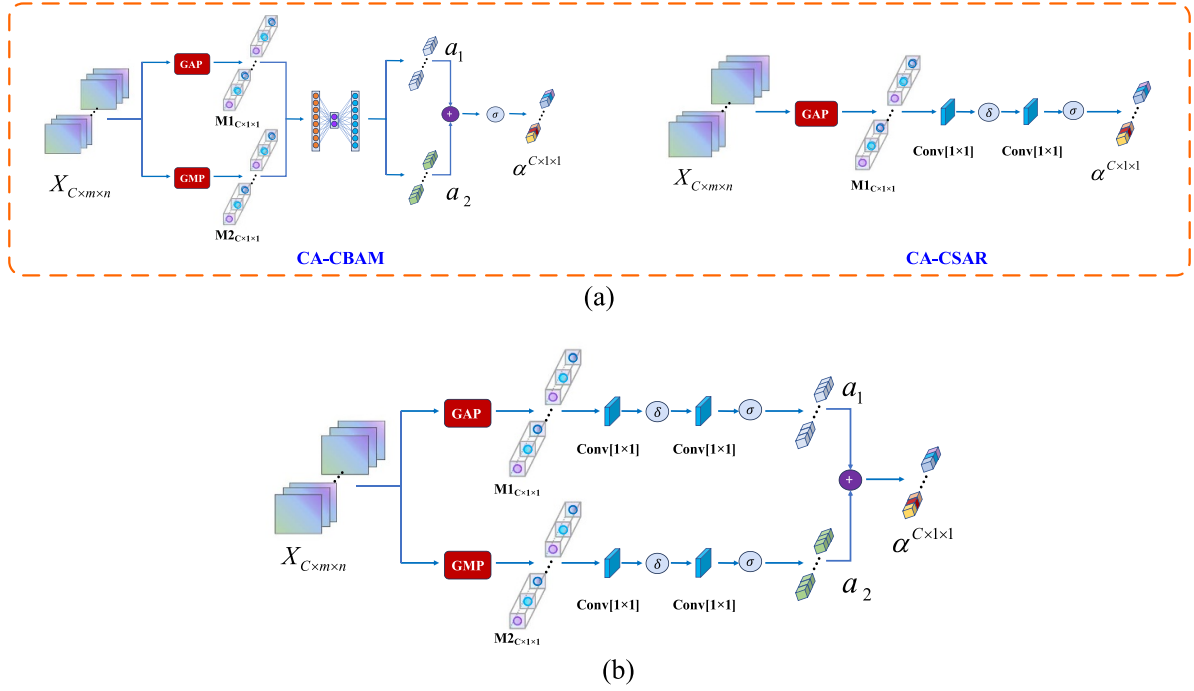


Figure 4. CA architecture diagram: (a) CA-CBAM and CA-CSAR structures; (b) improved CAE structure.

2.3. Multi-scale channels and spatial attention networks (MCSA)

The architecture of MCSA-Net, as illustrated in figure 3, integrates the concept of a residual network. Although the MSCNN module efficiently captures rich feature information, it may compromise the representational ability of critical features to some extent. Consequently, in this branch, a feature channel and spatial attention mechanism are introduced to further refine the extracted features. The attention mechanism selectively filters key feature information from the abundant multi-scale features obtained by MSCNN. The second branch comprises three convolutional layers, Conv3, Conv4, and Conv5. Conv3 and Conv4 employ $[13 \times 13]$ and $[11 \times 11]$ convolutional kernels, respectively, for fixed-scale feature extraction from the original input. Conv5 employs a $[1 \times 1]$ convolutional kernel, reducing model parameters by decreasing the output channels of this branch network.

2.3.1. Feature channel attention enhancement (CAE). In the domain of CNN, different channels correspond to distinct features, each contributing variably to the model's tasks. To address this, researchers have introduced convolutional attention mechanisms such as convolutional block attention module (CBAM) [52] and channel and spatial attention residual (CSAR) [53]. These mechanisms integrate both channel and spatial attention, offering a comprehensive focus that surpasses either attention type used in isolation. The CA attention mechanism in CBAM and CSAR, detailed in figure 4(a), serves as the foundation for an enhanced channel attention enhancement (CAE) and spatial attention enhancement (SAE)

mechanism proposed in this study. The original CA mechanism in CBAM, which utilizes fully connected layers for channel interaction, has been identified as inefficient and potentially detrimental to model performance [54]. The refined CAE mechanism amalgamates the strengths of CBAM and CSAR's CA mechanisms by substituting the fully connected layers with a more efficient one-dimensional convolution, thereby mitigating the drawbacks associated with fully connected layers. Additionally, while CSAR's CA only employs GAP, failing to account for global feature information, the enhanced CAE retains the GMP operation from CBAM, as illustrated in figure 4(b). This allows the model to dynamically emphasize features of significant contribution and diminish those of lesser relevance.

The utilization of GAP and GMP for feature CAE enables selective attention. The formulas for GAP and GMP are as follows:

$$F_{c1} = V_{\text{GAP}}(X) = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n X_c(i, j) \quad (3)$$

$$F_{c2} = U_{\text{GMP}}(X) = \text{Max}[X_c(i, j)] \quad (4)$$

where $X_c(i, j)$ represents the value of the input feature map X at the channel C in the feature map X_c at (i, j) , V_{GAP} is the global average pooling function, U_{GMP} is the global max pooling function, δ is the ReLU activation function, and σ is the sigmoid activation function. a_1 is the weight of channel enhancement of GAP on the feature map, the size is $c \times 1 \times 1$; a_2 is the weight of channel enhancement of GMP on the fea-

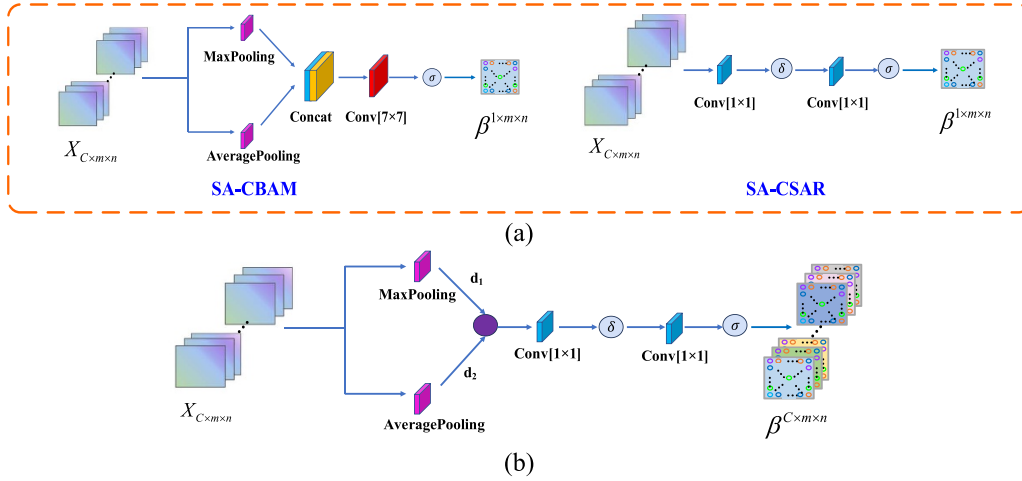


Figure 5. SA architecture diagram: (a) SA-CBAM and SA-CSAR structures; (b) improved SAE structure.

ture map, the size is the same as that of a_1 , and its calculation formula is as follows:

$$a_1 = \sigma(\text{Conv}_{1 \times 1}(\delta(\text{Conv}_{1 \times 1}(F_{c1})))) \quad (5)$$

$$a_2 = \sigma(\text{Conv}_{1 \times 1}(\delta(\text{Conv}_{1 \times 1}(F_{c2})))) \quad (6)$$

The CAE matrix α of the input feature map $X_{C \times m \times n}$ is calculated as:

$$\alpha = a_1 \oplus a_2 \quad (7)$$

where $\oplus$ represents the elements added point by point in order.

The weight matrix α of the output of the CAE is multiplied with the input features X in channel order to obtain the CAE feature map U_{CAE} , which is calculated as follows:

$$U_{\text{CAE}} = \alpha \otimes X_C. \quad (8)$$

2.3.2. Feature space attention enhancement (SAE). In the spatial domain, attention mechanisms are employed to either amplify or attenuate the significance of different positions within the feature map, thereby refining the model's focus on pertinent spatial features. The structures of spatial attention mechanisms in CBAM and CSAR, as depicted in figure 5(a), utilize a combination of pooling strategies and convolutions to extract comprehensive spatial feature information. Drawing inspiration from these mechanisms, an improved SAE mechanism is proposed, integrating the benefits of both CBAM's and CSAR's spatial attentions, with its architecture showcased in figure 5(b).

The structure of SAE is illustrated in figure 5(b), which extracts the importance of local and global feature information from different positions through max-pooling and average-pooling. Where the matrix extracted by max pooling is $d_1 = [d_1^1, d_1^2, \dots, d_1^c]$, d_1 has dimension $c \times m \times n$ and the matrix extracted by average pooling is $d_2 = [d_2^1, d_2^2, \dots, d_2^c]$, d_2 has dimension $c \times m \times n$, and $d_1 \oplus d_2$ is described as:

$$d_1^i = \begin{bmatrix} w_{d1}^{11} & w_{d1}^{12} & \dots & w_{d1}^{1n} \\ w_{d1}^{21} & w_{d1}^{22} & \dots & w_{d1}^{2n} \\ \dots & \dots & \dots & \dots \\ w_{d1}^{m1} & w_{d1}^{m2} & \dots & w_{d1}^{mn} \end{bmatrix}_{m \times n} \quad (9)$$

$$d_2^i = \begin{bmatrix} w_{d2}^{11} & w_{d2}^{12} & \dots & w_{d2}^{1n} \\ w_{d2}^{21} & w_{d2}^{22} & \dots & w_{d2}^{2n} \\ \dots & \dots & \dots & \dots \\ w_{d2}^{m1} & w_{d2}^{m2} & \dots & w_{d2}^{mn} \end{bmatrix}_{m \times n} \quad (10)$$

$$d_1^i \oplus d_2^i = \begin{bmatrix} w_{d1}^{11} + w_{d2}^{11} & \dots & w_{d1}^{1n} + w_{d2}^{1n} \\ \dots & \dots & \dots \\ w_{d1}^{m1} + w_{d2}^{m1} & \dots & w_{d1}^{mn} + w_{d2}^{mn} \end{bmatrix}_{m \times n} \quad (11)$$

where d_1^i and d_2^i are the weight matrices of the i th channel extracted by the max and average pooling operations, respectively, and w_{d1}^{mn} and w_{d2}^{mn} are the weights in the weight matrices extracted by the max and average pooling.

The SAE weight matrix of the features is formulated as:

$$\beta = \sigma(\text{Conv}_{1 \times 1}(\delta(\text{Conv}_{1 \times 1}(d_1 \oplus d_2)))) \quad (12)$$

where $\oplus$ represents the point-by-point summation of the elements in the spatial weight matrix of each feature map in the order of the feature map channels.

The formula for computing the SAE map U_{SAE} corresponding to feature map X is as follows:

$$U_{\text{SAE}} = \beta \otimes X \quad (13)$$

where $\otimes$ denotes the multiplication of the corresponding positional elements.

2.3.3. Dense residual feature fusion network. The DRF network structure, shown in figure 6, introduces dense residual connectivity to prevent network deepening beyond optimal levels, thereby avoiding potential performance degradation. This connectivity facilitates the re-utilization of

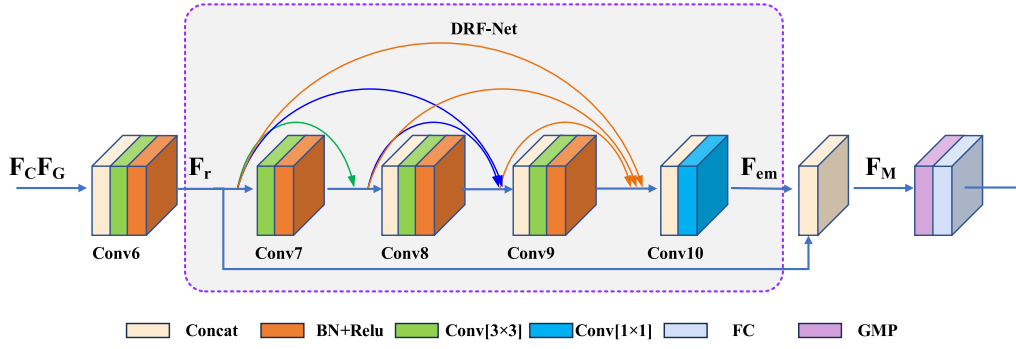


Figure 6. DRF architecture diagram.

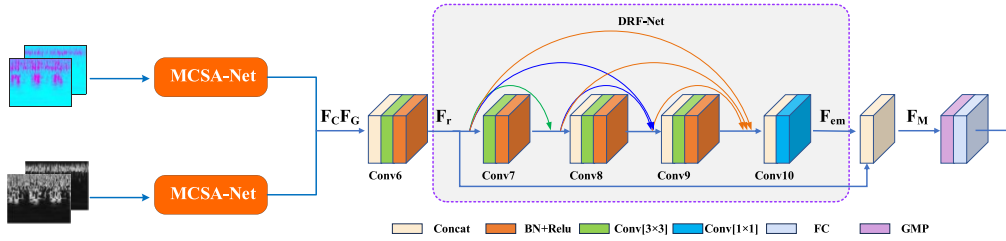


Figure 7. DMCSA-DRF architecture diagram.

color-grayscale time-frequency feature maps and addresses the gradient vanishing issue common in deep networks.

The DRF network contains three convolutional layers, whose input features F_r are described as:

$$F_r = \delta(\text{Conv}^6[F_C, F_G]) \quad (14)$$

where $[F_C, F_G]$ is the color-grayscale spliced feature map extracted by MCSA, Conv^6 is the sixth single-scale convolutional layer, and δ is the relu activation function. The output of the m th convolutional layer in the DRF is:

$$F_{d,m} = \delta(w_{d,m}[F_r, F_{d,0}, \dots, F_{d,m-1}]) \quad (15)$$

where $w_{d,m}$ denotes the weights of the m th convolutional layer in the DRF, and $[F_r, F_{d,0}, \dots, F_{d,m-1}]$ denotes the stitching of the residually connected feature maps. In the last convolutional layer, Conv10, of DRF, the 1×1 convolutional kernel is utilized for parameter and dimension reduction. The formula for calculating the final output F_{em} of the DRF is described as:

$$F_{em} = \text{Conv}^{10}[F_r, F_{d,0}, F_{d,1}, F_{d,2}]. \quad (16)$$

The final DRF output is: $F_M = [F_{em}, F_r]$.

2.4. Design of the DMCSA-DRF prediction model

2.4.1. DMCSA-DRF model. A two-branch MCSA network is implemented to extract features from both color and grayscale time-frequency image data, merging these features through a DRF network. This approach, illustrated in

figure 7, underscores the emphasis on texture feature information within the dataset.

The model parameters are shown in table 2.

2.4.2. DMCSA-DRF model prediction process. Figure 8 illustrates the prediction process of the DMCSA-DRF model, comprising the following key steps:

Step 1: Data preprocessing:

The time-frequency feature image of the original vibration signal undergoes extraction using CWT. Subsequently, the extracted time-frequency feature image is converted to greyscale to construct the color-greyscale time-frequency feature image dataset. Additionally, the one-dimensional time-frequency domain of the original signal is extracted, normalized, and smoothed using Gaussian filtering to formulate the HI of the training set.

Step 2: Noise injection:

Salt & pepper noise, Gaussian noise, and speckle noise are injected into the training data to simulate the interference signals in real industrial processes, respectively. The data containing noise is input into the DMCSA-DRF model for training.

Step 3: Model training

During model training, the COA optimization algorithm is employed to optimally search for key hyperparameters such as InitialLearnRate, MiniBatchsize, and MaxEpochs in the model.

Step 4: Model evaluation

Test data is input into the trained model for prediction, and the model's performance is assessed using metrics such as relative accuracy (RA), mean absolute error (MAE), root mean squared error (RMSE), and the coefficient of determination (R^2). Additionally, the complete full-life dataset is

Table 2. Model parameters.

MSCA-DRF					
Layer name		Kernel size	Stride	Number of channels	Output size
MS-Conv1	MS-CNN	$[9 \times 9], [7 \times 7], [5 \times 5]$	2	32	$[112 \times 112 \times 32]$
	Concat	—	—	96	$[112 \times 112 \times 96]$
	Pooling	$[2 \times 2]$	1	—	$[112 \times 112 \times 32]$
MS-Conv2	MS-CNN	$[7 \times 7], [5 \times 5], [3 \times 3]$	2	64	$[56 \times 56 \times 64]$
	Concat	—	—	192	$[56 \times 56 \times 192]$
	Pooling	$[2 \times 2]$	1	—	$[56 \times 56 \times 192]$
Conv1	CNN	$[1 \times 1]$	1	96	$[112 \times 112 \times 96]$
	Concat	—	1	—	$[112 \times 112 \times 192]$
Conv2	CNN	$[1 \times 1]$	1	96	$[56 \times 56 \times 96]$
	Concat	—	—	384	$[56 \times 56 \times 384]$
Conv3	CNN	$[13 \times 13]$	2	64	$[112 \times 112 \times 64]$
	Pooling	$[2 \times 2]$	1	—	$[112 \times 112 \times 64]$
Conv4	CNN	$[11 \times 11]$	1	128	$[112 \times 112 \times 128]$
	Pooling	$[2 \times 2]$	2	—	$[56 \times 56 \times 128]$
Conv5	CNN	$[1 \times 1]$	2	64	$[56 \times 56 \times 64]$
Conv6	CNN	$[3 \times 3]$	1	64	$[56 \times 56 \times 64]$
	Concat	—	—	160	$[56 \times 56 \times 160]$
Conv7	CNN	$[3 \times 3]$	1	128	$[56 \times 56 \times 128]$
Conv8	CNN	$[3 \times 3]$	1	256	$[56 \times 56 \times 256]$
	Concat	—	—	192	$[56 \times 56 \times 192]$
Conv9	CNN	$[3 \times 3]$	1	64	$[56 \times 56 \times 128]$
	Concat	—	—	448	$[56 \times 56 \times 384]$
Conv10	CNN	$[1 \times 1]$	2	64	$[28 \times 28 \times 64]$
	Concat	—	—	576	$[56 \times 56 \times 576]$

truncated, and the truncated HI for model prediction is combined with a regression model to facilitate RUL prediction post-truncation.

The formulas for calculating the aforementioned evaluation indices are as follows:

(1) Relative accuracy (RA)

$$RA = 1 - \frac{|y_i - p_i|}{y_i} \quad (17)$$

where y_i is the actual values, p_i is the predicted values.

(2) Mean absolute error (MAE)

$$MAE = \frac{1}{N} \sum_{i=1}^N (y_i - p_i) \quad (18)$$

where N is the number of samples evaluated.

(3) Root mean squared error (RMSE)

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - p_i)^2} \quad (19)$$

(4) Coefficient of determination (R^2)

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - p_i)^2}{\sum_{i=1}^N (\bar{y} - y_i)^2} \quad (20)$$

The closer R^2 is to 1, the better the model fitting performance.

3. Results

3.1. Experiments and analyses case I

3.1.1. Data sources. In Case I, the PHM2012 rolling bearing accelerated life experimental dataset [55] is selected for validation of the DMCSA-DRF model. The dataset was collected by the PRONISTIA experimental platform as shown in figure 9, which can complete the bearing accelerated degradation-to-failure experiment in a few hours. Two acceleration sensors are deployed in the vertical and horizontal directions of the test bearing, with a sampling frequency of

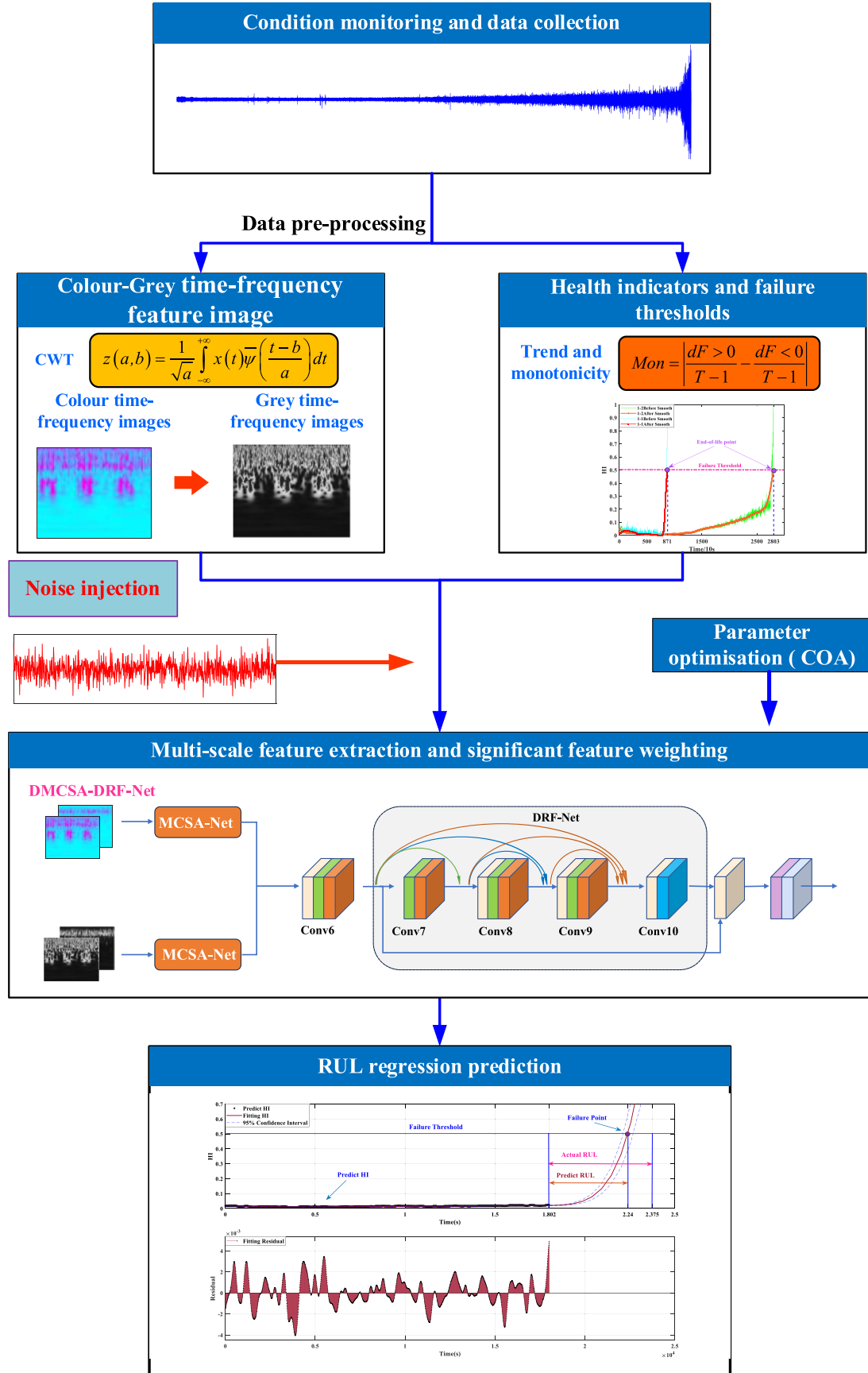


Figure 8. MCSA-DRF overall structure diagram.

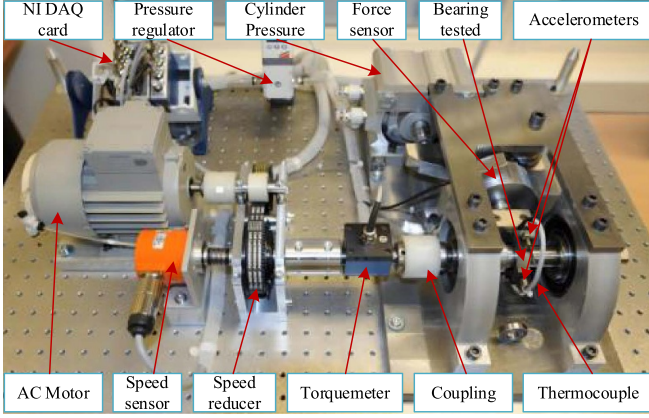


Figure 9. PRONOSTIA experimental platform.

Table 3. Description of PRONOSTIA dataset.

Condition	Speed (rpm)	Load (N)	Training set	Test set
I	1800	4000	Bearing1_1, Bearing1_2	Bearing1_3 to Bearing1_7
II	1650	4200	Bearing2_1, Bearing2_2	Bearing2_3 to Bearing2_7
III	1500	5000	Bearing3_1, Bearing3_2	Bearing3_3

25.6 kHz, a sampling interval of 10 s, and a sampling duration of 0.1 s each time to collect bearing vibration information. The dataset contains data under three different operating conditions, including seven datasets for each of operating conditions I and II, and three datasets for operating condition III, with a sampling frequency of 25.6 and a sampling interval of 10 s. The description of the dataset is shown in table 3.

3.1.2. Data pre-processing

3.1.2.1. Vibration signal feature extraction. This paper opts for the extraction of time-frequency domain features from the complete vibration signals of bearings. These features are evaluated through monotonicity constraints, with those demonstrating strong monotonicity utilized to construct HI. The calculation formula is represented as follows:

$$\text{Mon} = \left| \frac{dF > 0}{T-1} - \frac{dF < 0}{T-1} \right| \quad (21)$$

where T represents the length of the feature sequence, dF represents the differential of the feature sequence, and a higher monotonicity value approaching 1 indicates better feature monotonicity. The monotonicity corresponding to the 21 time-frequency features of the quantized PRONOSTIA bearing 1-1 is depicted in figure 10.

Upon analysis of figure 10, it is observed that among the 21 extracted features, the time domain feature ‘rootAmplitude’

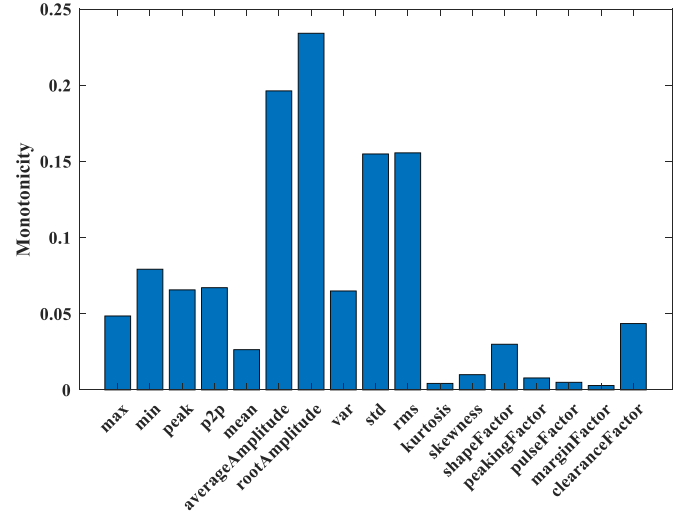


Figure 10. Monotonicity of feature metrics.

exhibits the highest monotonicity. Consequently, ‘rootAmplitude’ is utilized as the HI for bearings operating under the same conditions.

3.1.2.2. Health indicator and failure thresholds. The selected features with strong monotonicity were normalized and smoothed to be used as HI, with Gaussian filtering chosen for the smoothing process, employing a sliding window set to 15.

Figure 11 illustrates the smoothed HI for two training sets, bearing1_1 and bearing1_2, from Condition I of the PRONOSTIA dataset. The actual lifetimes of bearing1_1 and bearing1_2 are 28030s and 8710s, respectively. In this paper, the HI values corresponding to the end of life of the training set bearings are used as the fault thresholds for the same operating conditions [56]. The HI values corresponding to the end of life of the two training set bearings in figure 11 are 0.5. Therefore, the fault threshold for this operating condition is set to 0.5. The same method is applied to set the fault thresholds for other operating conditions. For the PRONOSTIA dataset, the fault threshold for condition II is $HI = 0.2$, and for condition III, it is $HI = 0.5$.

3.1.2.3. Color time-frequency image feature extraction.

Given the non-stationary and instantaneous nature of bearing vibration signals, CWT is employed to capture instantaneous characteristics and enhance the time-frequency features of the vibration signals. The formulation of CWT is as follows:

$$z(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t) \bar{\psi} \left(\frac{t-b}{a} \right) dt \quad (22)$$

where $x(t)$ is the input signal, a is the complex conjugate of the mother wavelet function, a is the scale parameter, which is used to capture the different frequency components; and b is

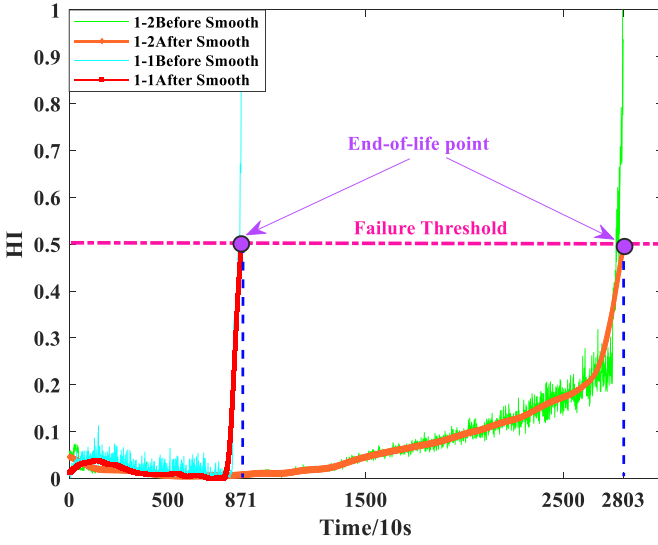


Figure 11. Feature degradation trends for the PRONOSTIA dataset for condition I.

the translation parameter, which is used to control the translation of the wavelet function on the time axis

$$\psi(t) = e^{-\frac{1}{2}t^2} e^{j\omega_0 t} \quad (23)$$

where ω_0 is the center frequency of the wavelet.

The Fourier transform of $\psi(t)$ is:

$$\varphi(\omega) = \sqrt{2\pi} e^{-\frac{1}{2}(\omega - \omega_0)^2}. \quad (24)$$

This yields a continuous wavelet transform of the signal in the frequency domain

$$z(a, b) = \sqrt{a} \int_{-\infty}^{+\infty} f(\omega) \cdot \varphi^*(a\omega) \cdot e^{j\omega b} dt \quad (25)$$

where $f(\omega)$ is the Fourier transform of $x(t)$, $\varphi^*(a\omega) \cdot e^{j\omega b}$ is the Fourier transform of $\bar{\psi}\left(\frac{t-b}{a}\right)$.

The above expression can be discretized as:

$$z(m, n) = \sqrt{m\Delta a} \sum_n f(f_n) \varphi^*(m\Delta a f_n) \cdot e^{j2\pi f_n n \Delta b} \quad (26)$$

where f_n is the discrete frequency.

3.1.3. Experiments and analysis.

3.1.3.1. Determination of training parameters. To identify the most advantageous hyperparameter combinations for model optimization, this study implements the COA to conduct a systematic exploration of key model parameters. A detailed comparison is carried out against several advanced optimization algorithms, including the dragonfly algorithm (DA), grey wolf optimization algorithm (GWO), whale optimization algorithm (WOA), and Bat optimization algorithm (BAT). The COA rigorously tests various hyperparameter

configurations, assessing model performance across diverse hyperparameter settings to pinpoint the optimal combinations that significantly enhance model training efficacy.

Figure 12 showcases the comparative analysis results between the COA and other optimization algorithms across 12 test functions. Notably, the population size, maximum number of iterations, and dimensionality are all standardized for this comparison. The historical search distribution plot of the COA distinctly reveals a concentration of search trajectories around the optimal solution, highlighting the algorithm's precision in navigating the hyperparameter space. Moreover, the convergence curve comparison underscores COA's superior convergence speed and accuracy over competing algorithms for test functions F1 to F12, as detailed in table 4.

To substantiate the effectiveness of the COA algorithm, it was benchmarked against the Grid Search (GS) algorithm, a conventional approach in machine learning for exhaustive hyperparameter search. The comparative validation, summarized in table 5, where 'Non' represents the training set without noise and the test set contains noise (including 'salt & pepper noise,' 'Gaussian noise,' and 'speckle noise'). The other cases are injected with noise for both the test set and the training set.

Table 5 indicates that under non-noisy conditions, hyperparameters optimized via COA display negligible differences in predictive accuracy compared to those determined by GS, yet with a notable reduction in training time. Specifically, in noisy environments, COA-optimized models demonstrate significantly reduced training times by 72.2%, compared to GS-optimized models, evidencing COA's enhanced efficiency and effectiveness in parameter optimization under adverse conditions. Given these findings, the COA emerges as the preferred algorithm for hyperparameter optimization within this study, attributed to its superior search efficiency. This decision is supported by COA's demonstrated capability to expedite the optimization process while ensuring the maintained accuracy of models, particularly in environments affected by noise.

3.1.3.2. Validation of features extracted by different-sized convolutional kernels. Figure 13 illustrates the results of feature extraction utilizing multi-scale convolutional kernels. The analysis indicates that smaller-sized convolution kernels primarily capture detailed low-frequency information within a local data range, while larger-sized kernels focus on extracting high-frequency overall information across a wider data spectrum. The utilization of various kernel sizes, as opposed to a single-scale approach, enhances the richness of the extracted feature information.

3.1.3.3. Validation of attention mechanism effectiveness. Figure 14 offers a comparative analysis of model outputs with and without the incorporation of attention mechanisms. The enhanced clarity and definition of feature outlines in models utilizing channel and spatial attention mechanisms signify a marked improvement in feature representation. This enhancement facilitates a more nuanced differentiation between essential and peripheral features, optimizing the model's focus and interpretability.

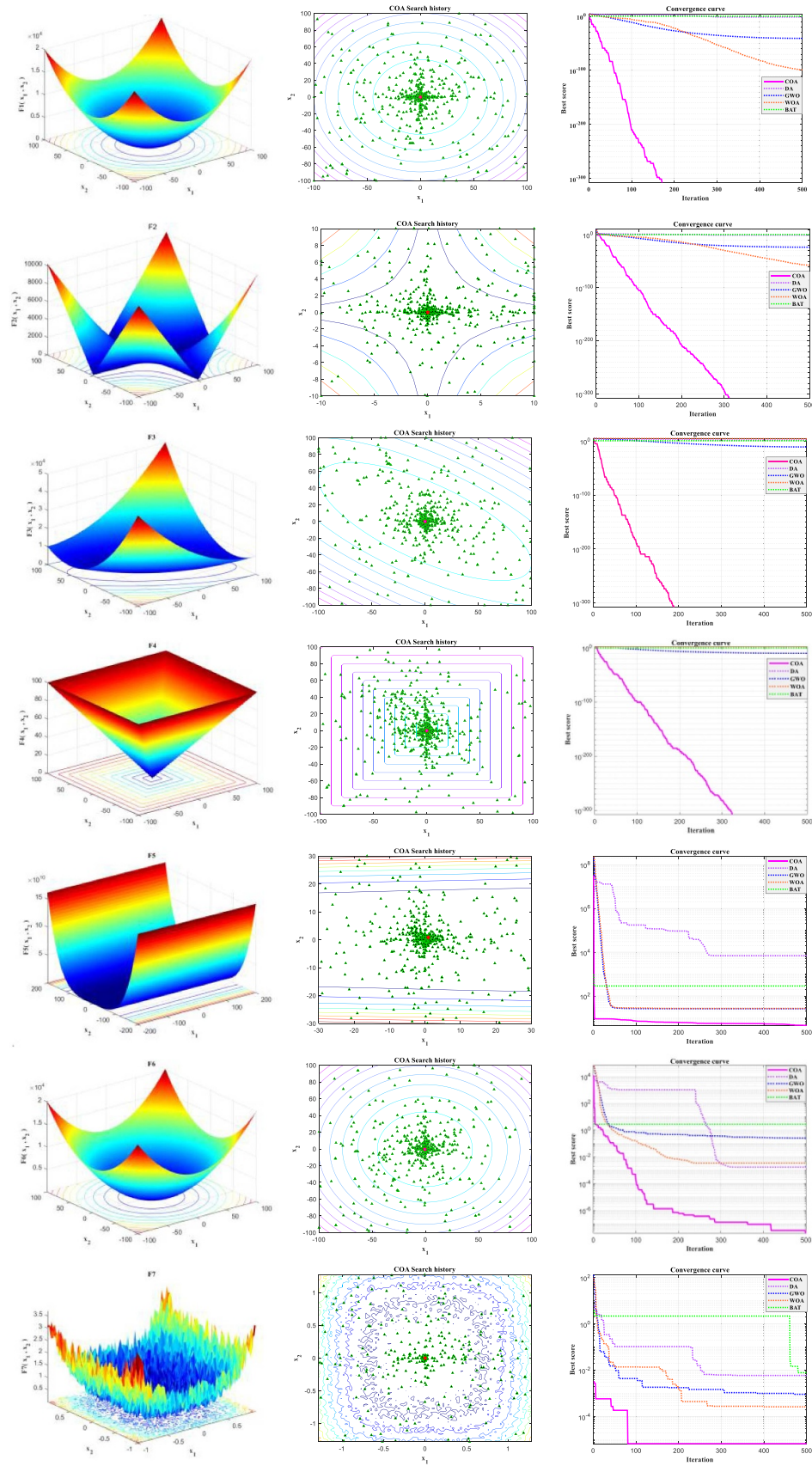


Figure 12. Comparative results of COA algorithm on test functions F1–F12 against other algorithms.

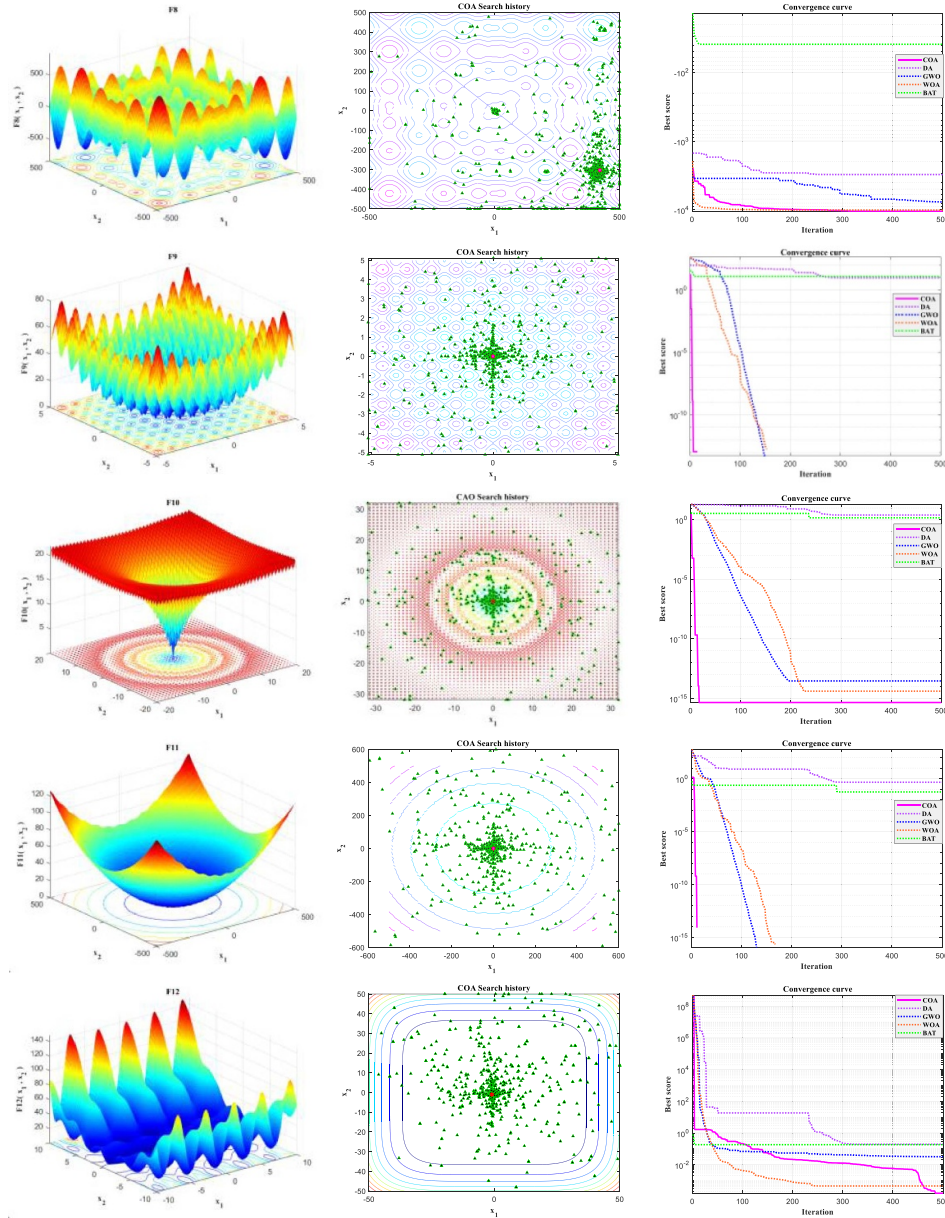


Figure 12. (Continued.)

Table 4. Hyperparameter search results.

Parameter index	Hyperparameter name	Search range	Best search results
h_1	InitialLearnRate	$[10^{-2}-10^{-6}]$	10^{-4}
h_2	MiniBatchsize	$[16, 32, \dots, 521, 1024]$	64
h_3	MaxEpochs	$[20-100]$	40

3.1.3.4. Validation of the model's noise resistance performance. To assess the model's resistance to noise, three distinct types of noise are introduced during the training process:

- Salt & pepper noise:** This type of noise, akin to impulse noise, randomly sets pixel values of some points in the image to 0 and 255.
- Gaussian noise:** Gaussian noise is generated by adding a noise signal obeying a Gaussian distribution to the original data.
- Speckle noise:** Speckle noise arises from the multiplication of the noise signal with the original signal and typically stems from the system's time-varying or non-linear nature.

The input feature map after adding noise is shown in figure 15.

Table 5. GS algorithm and COA algorithm search results under noise conditions.

Model	Noise (−5 dB–30 dB)	Average accuracy (RA%)	Time	h_1	h_2	h_3
GS	Non	89.03	133.6 min	10^{-4}	64	20
	Salt & pepper noise	96.25	157.1 min	10^{-5}	32	30
	Gaussian noise	94.33	166.3 min	10^{-3}	64	30
	Speckle noise	95.57	154.7 min	10^{-5}	32	20
COA	Non	90.56	30.2 min	10^{-4}	64	40
	Salt & pepper noise	96.25	45.6 min	10^{-4}	64	40
	Gaussian noise	95.91	42.4 min	10^{-4}	32	30
	Speckle noise	95.73	44.4 min	10^{-3}	64	30

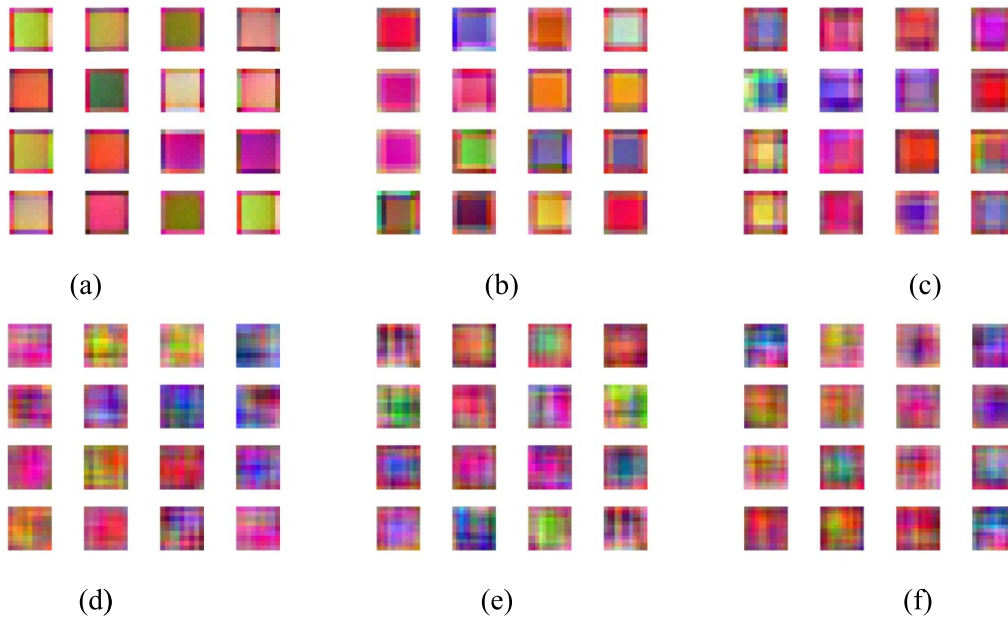
**Figure 13.** Output features of different-sized convolutional kernels: (a) 3×3 convolution output; (b) 5×5 convolution output; (c) 7×7 convolution output; (d) 9×9 convolution output; (e) 11×11 convolution output; (f) 13×13 convolution output.

Figure 16 delves into the predictive accuracy of the MSCA-DRF training under varied noise conditions, establishing the model's adeptness in maintaining accuracy across a spectrum of signal-to-noise ratios (SNRs) from −5 dB to 30 dB. Where 'without noise' means that no noise is injected in the training phase of the model, but the test set contains noise. Notably, the introduction of noise during training phases does not uniformly affect predictive accuracy; while lower SNR levels see effective mitigation of noise's impact on key features, higher SNR conditions demonstrate the model's intrinsic capacity for discerning critical degraded features amidst noise, affirming the model's resilience.

To validate the noise robustness of the proposed method, combinations of the MSCA network with the DRF network and the MSCNN network with a multi-scale dilated convolution MS-DCNN network structure were tested. The CNN-DRF model substitutes the MSCNN layer in MSCA with a 3×3 single-scale convolutional kernel, excluding CAE and SAE, while keeping the DRF component unchanged.

The MSCNN-DRF model simply omits CAE and SAE from the MSCA-DRF configuration. MS-DCNN-DRF replaces MSCNN-DRF's multiscale convolution with a multiscale dilated convolution structure, applying dilation rates of 2, 3, and 4, optimized via the COA algorithm. MDCSA-CNN introduces CAE and SAE into a multi-scale dilated convolution framework, diverging from MSCA-DRF by replacing DRF with a conventional convolution structure. The experimental evaluation, conducted across ten iterations per model against varying levels of salt & pepper noise, aimed to ascertain average accuracy, with results showcased in figure 17.

Figure 17 showcases the performance comparison of six distinct structural models when subjected to salt and pepper noise, with the signal-to-noise ratio (SNR) ranging from −5 dB to 30 dB. The MSCA-DRF model exhibited a notable enhancement in prediction accuracy, quantified as a 10.4% relative increase. Similarly, the MSCA-CNN, MS-DCNN-DRF, MSCNN-DRF, and CNN-DRF models also showed relative increases in prediction accuracy, each marked at 24.1%,

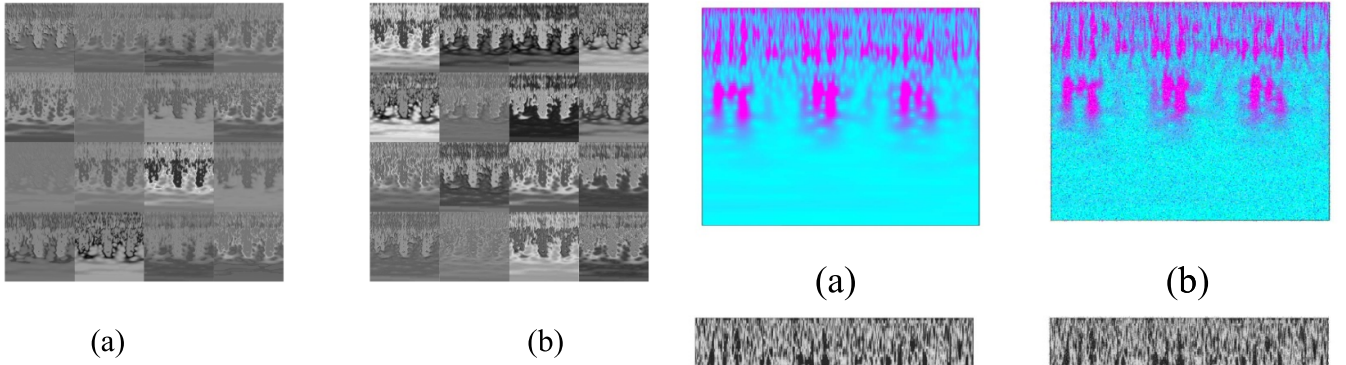


Figure 14. Output comparison between models with and without attention mechanism: (a) output of the model without attention mechanism; (b) output of the model with attention mechanism.

21.9%, 19.7%, and 25.5%. The magnitude of the relative increase in prediction accuracy serves as an indicator of the model's sensitivity to noise interference; higher increases suggest a lower noise resistance, while lower increases denote stronger resilience. Through this comparative analysis, it becomes clear that the MCSA-DRF model demonstrates superior noise resistance capabilities compared to the other evaluated structural prediction models.

Table 6 displays the average prediction accuracies for a test set subjected to salt & pepper noise, Gaussian noise, and speckle noise across SNR ranging from -5 dB to 30 dB. Compared to the CNN-DRF baseline, the utilization of a multi-scale convolutional structure in MS-DCNN-DRF and MSCNN-DRF resulted in reductions of 9.23% and 12.84% in the average MAE (AVE_{MAE}) and the average RMSE (AVE_{RMSE}), respectively. Additionally, the average RA (AVE_{RA}) and the average R^2 (AVE_{R^2}) are improved by 3.17% and 6.41%, respectively. This enhancement is attributed to the utilization of multi-scale features, providing a broader and more detailed spectrum of information, thereby significantly enhancing the models' feature extraction capabilities. Compared to MDSCA-CNN, MCSA-CNN resulted in reductions of 21.15% and 8.67% in the AVE_{MAE} and AVE_{RMSE} , respectively. The AVE_{RA} and the AVE_{R^2} are improved by 3.53% and 2.22%, respectively. Despite the multi-scale dilated convolution's capability for extensive feature extraction, it is noted that consecutive dilated convolution can introduce a checkerboard effect [57], potentially omitting crucial features. The proposed model, incorporating MCSA and DRF strategies, outperformed both the CNN-DRF and MCSA-CNN models in terms of average prediction accuracy by a significant margin. This affirms the effectiveness of combining multi-scale convolutional attention with dense residual fusion in enhancing model accuracy amidst noise disturbances.

Figure 18 shows the results of the comparison of the training time and testing time of each model on the PRONOSTIA dataset. In the analysis presented in figure 18, The MS-DCNN-DRF and MSCNN-DRF models exhibit an increase in training time compared to the CNN-DRF model, attributed to the computational demands of their multi-scale structures. Similarly, the MDSCA-CNN model's training duration

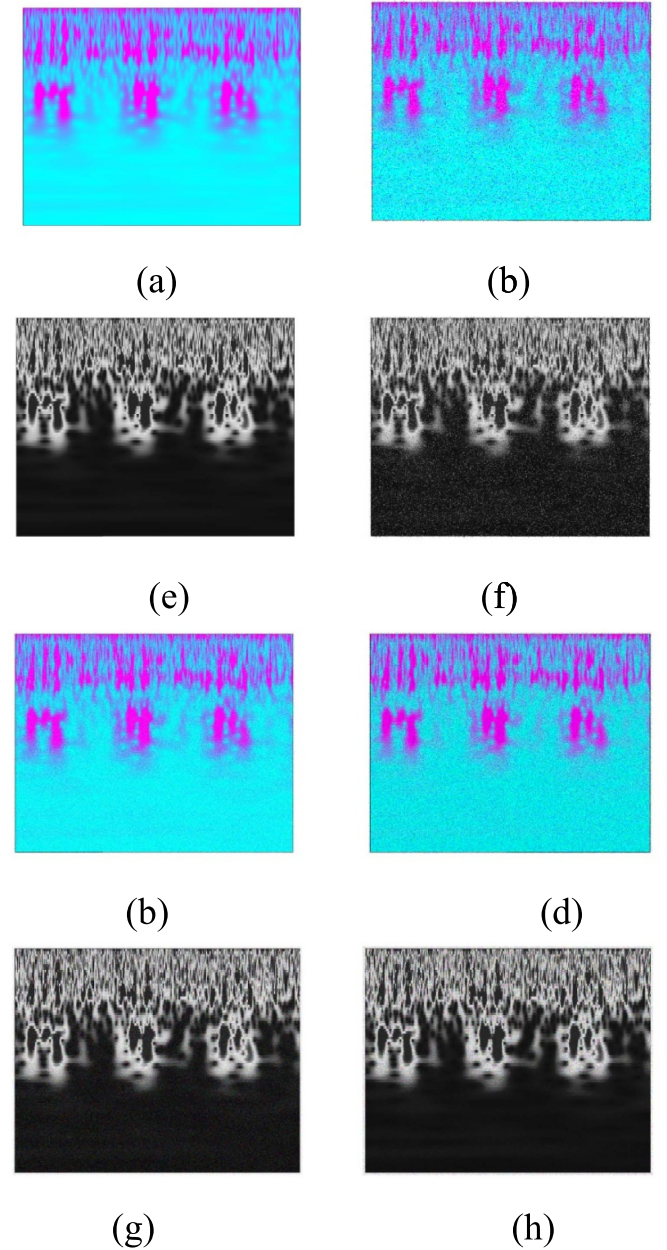


Figure 15. Color-grayscale time-frequency noisy images: (a), (e) original images; (b), (f) images with added salt & pepper noise; (c), (g) images with added Gaussian noise; (d), (h) images with added speckle noise.

extends beyond that of MCSA-CNN, underscoring the higher computational load of multi-scale dilated convolution versus standard multi-scale convolution. Among all models evaluated, the MCSA-DRF configuration requires the longest training period. However, during the online testing phase, the MCSA-DRF model's testing duration is remarkably efficient, with a minimal difference from the CNN-DRF model, which represents the benchmark for speed. This minimal discrepancy in testing time across models underscores a crucial consideration in real-world applications, where the emphasis is often placed on diagnostic accuracy and the ability to perform timely online testing rather than solely on training duration.

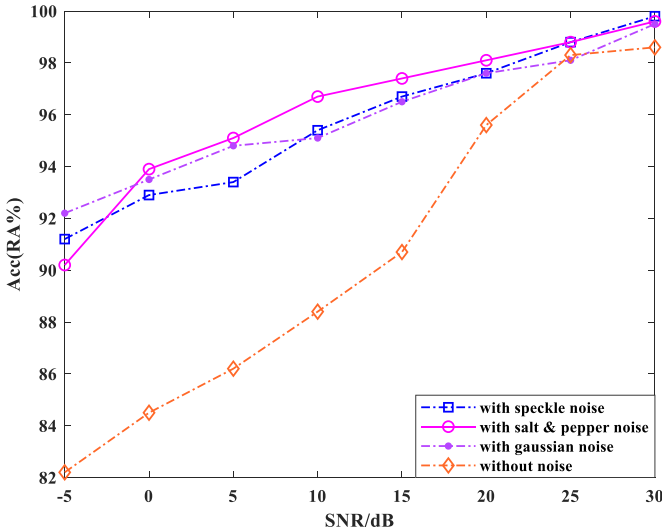


Figure 16. Prediction accuracy of the MCSA-DRF model on the test set under three types of noise during both noise and noise-free training.

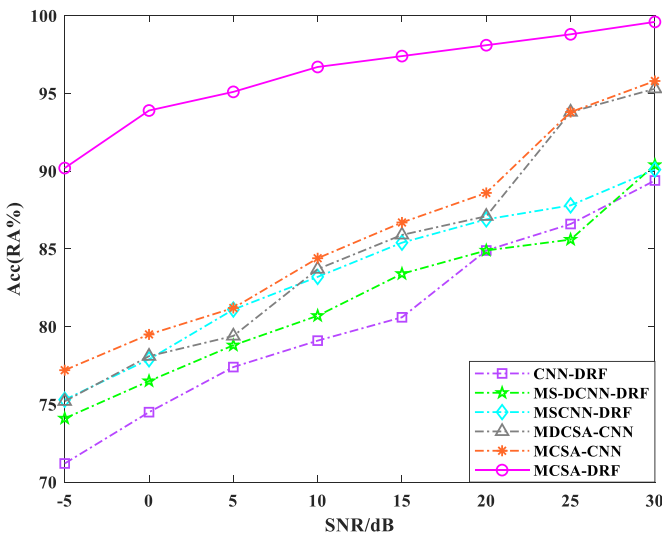


Figure 17. Comparative results of different models under salt & pepper noise.

Consequently, despite the extended training time, the MCSA-DRF model's superior accuracy and competitive testing speed render it a more advantageous choice compared to the alternative configurations evaluated, highlighting its potential for effective deployment in industrial settings.

3.1.3.5. Validation of the model in non-noisy conditions performance. Figure 19 displays the predictive outcomes for test sets 1_5 and 2_5 from the PRONOSTIA dataset, where the actual HI is marked by a red solid line, the model's predicted HI by a blue curve, and the prediction error by an orange line. To assess the predictive performance of the proposed model, we compare it with five other models. These include three different combinations of model configurations from the noise experiments, along with two additional convolutional

models featuring attention mechanisms, namely GAM-CNN [58] and CBAM-MSCNN [59]. The prediction errors for all test sets are summarized in table 7.

Table 7 provides a comprehensive overview of the performance metrics for all test set data under the evaluated models. It indicates that the proposed model has a 31.8% and 24.7% decrease in MAE and a 26% and 15% decrease in RMSE compared to the single branch structure CNN [58] and MSCNN [59] with an attention mechanism. Compared with MSCNN-DRF and CNN-DRF, the MAE of the proposed model decreases by 64.8% and 70%, and the RMSE decreases by 53.9% and 58%, respectively. Despite the increase in computational and training time necessitated by the MCSA-DRF's two-branch structure, the enhancements in MAE and RMSE underscore the efficacy of the model's advanced attentional mechanism in refining feature extraction capabilities.

Figure 20 and table 8 provide further insights into the quantitative evaluation of various models applied to the PRONOSTIA dataset. The proposed models, denoted as $MAE_{MCSA-DRF} = 0.374$, $RMSE_{MCSA-DRF} = 13.099$, $RA_{MCSA-DRF} = 0.903$, and $R^2_{MCSA-DRF} = 0.986$, demonstrate superior evaluation index values compared to existing single-branch CNN [58] and MSCNN models [59] with attention mechanisms. This enhancement confirms the dual-branch structure's ability to effectively capture and interpret the complex information inherent in degraded features. Moreover, the proposed method's notable improvements in prediction accuracy over models without an attention mechanism affirm the value of integrating attention mechanisms for heightened feature discernment. Comparatively, the MCSA model with the DRF structure exhibits marked accuracy improvements over the MCSA-CNN, validating the DRF's contribution to bolstering the model's overall predictive.

3.2. Experiments and analyses case II

3.2.1. Data sources. The XJTU-SY accelerated degradation dataset [60] acquisition platform is shown in figure 21, which allows accelerated life degradation experiments to be performed on various types of bearings under different operating conditions. The sensor deployment position is the same as that of PRONOSTIA, the sampling frequency is 25.6 kHz, the sampling interval is 1 min, and the duration of each sample is 1.28 s. The description of this data set is shown in table 9.

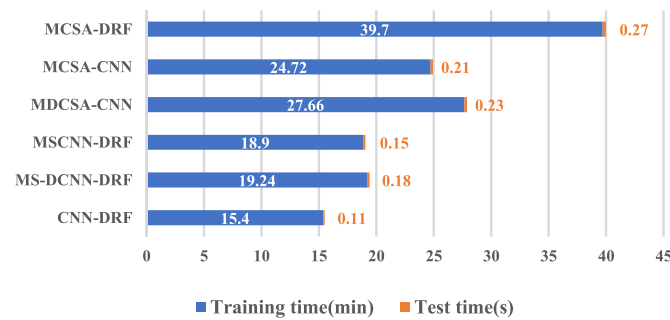
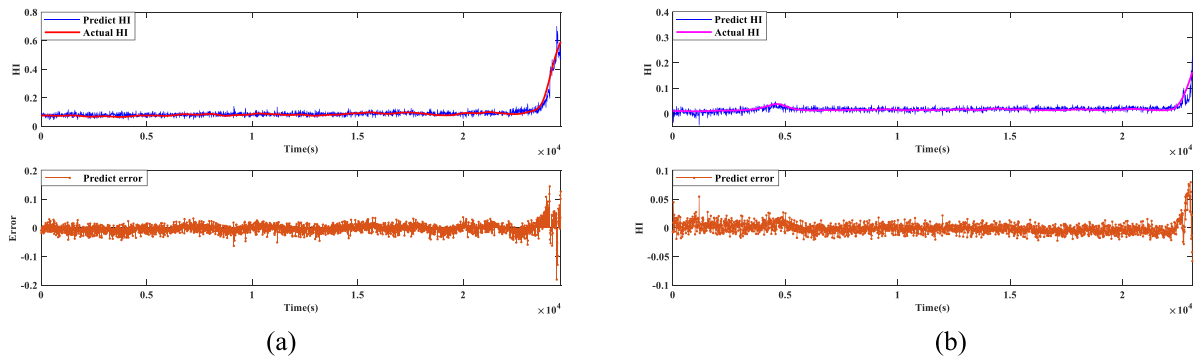
3.2.2. Data pre-processing

3.2.2.1. Health indicator and failure thresholds Figure 22 shows the smoothed HI of three training sets, bearing1_1, bearing1_4, and bearing1_5, in operational condition I of the XJTU-SY dataset.

The actual lifetimes of bearing1_1, bearing1_4, and bearing1_5 are 52 min, 122 min, and 123 min, respectively. The HI values corresponding to the end of life of the three training set bearings in figure 22 are 0.7. Therefore, the fault threshold for this operating condition is set to 0.7. For the XJTU-SY dataset, the fault threshold for condition II is $HI = 0.8$.

Table 6. Average prediction results of the test set under three types of noise ranging from -5 dB to 30 dB for different models.

Model	Salt & pepper noise				Gaussian Noise				Speckle noise			
	MAE	RMSE	RA	R^2	MAE	RMSE	RA	R^2	MAE	RMSE	RA	R^2
CNN-DRF	0.64	20.19	0.80	0.77	0.69	22.10	0.79	0.75	0.61	21.14	0.79	0.81
MS-DCNN-DRF	0.60	18.08	0.82	0.81	0.66	21.47	0.80	0.80	0.57	18.10	0.82	0.84
MSCNN-DRF	0.58	17.92	0.83	0.87	0.59	19.11	0.82	0.87	0.53	16.77	0.82	0.83
MDCSA-CNN	0.52	15.1	0.85	0.91	0.57	18.58	0.83	0.89	0.48	14.05	0.87	0.90
MCSA-CNN	0.44	13.94	0.86	0.93	0.38	17.73	0.87	0.96	0.41	11.92	0.94	0.96
MCSA-DRF	0.18	6.27	0.96	0.97	0.12	4.01	0.98	0.99	0.09	3.11	0.98	0.98

**Figure 18.** Training time and testing time of the model on the PRONOSTIA dataset.**Figure 19.** Prediction results: (a) prediction results for bearings 1-5; (b) prediction results for bearings 2-5.**Table 7.** Evaluation results of the test set.

Test set	MCSA-DRF		MCSA-CNN		[58]		[59]		MSCNN-DRF		CNN-DRF	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Bearing1-3	0.35	12.26	0.42	12.47	0.55	15.12	0.47	14.07	1.13	31.73	1.27	35.67
Bearing1-4	0.24	9.85	0.39	14.94	0.33	12.82	0.32	11.98	1.57	39.19	1.91	47.35
Bearing1-5	0.06	11.33	0.97	31.41	1.12	30.97	0.81	26.52	1.09	30.67	1.47	37.21
Bearing1-6	0.26	9.94	0.37	13.76	0.43	13.05	0.36	13.43	0.67	23.92	0.88	27.35
Bearing1-7	0.57	15.25	0.62	22.13	0.82	26.66	0.73	24.12	1.29	36.03	1.41	36.84
Bearing2-3	0.20	8.67	0.29	10.44	0.34	13.11	0.26	12.66	0.79	26.35	0.69	25.17
Bearing2-4	0.16	7.24	0.22	9.27	0.28	10.22	0.24	11.27	1.07	31.01	0.94	30.06
Bearing2-5	0.05	15.76	0.83	26.74	0.84	26.84	0.89	27.46	1.15	32.11	1.61	42.31
Bearing2-6	0.19	7.91	0.27	10.06	0.23	9.65	0.31	10.51	0.76	25.16	1.22	33.29
Bearing2-7	0.49	14.84	0.51	15.01	0.53	15.08	0.61	21.62	0.91	29.04	1.49	39.05
Bearing3-3	0.21	9.04	0.38	14.25	0.44	13.55	0.46	13.81	1.26	34.18	0.71	24.38

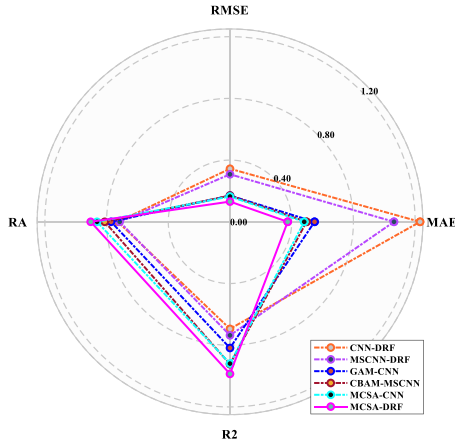


Figure 20. Model prediction performance evaluation chart.

Table 8. Comparison of prediction results of the PRONOSTIA dataset in different models.

Model	Evaluation indicators			
	MAE	RMSE	RA	R ²
CNN-DRF	1.236	34.425	0.727	0.694
MSCNN-CNN	1.063	30.763	0.715	0.736
GAM-CNN [58]	0.537	17.006	0.801	0.818
CBAM-MSCNN [59]	0.496	17.041	0.812	0.903
MCSA-CNN	0.479	16.407	0.862	0.919
MCSA-DRF	0.374	13.099	0.903	0.986

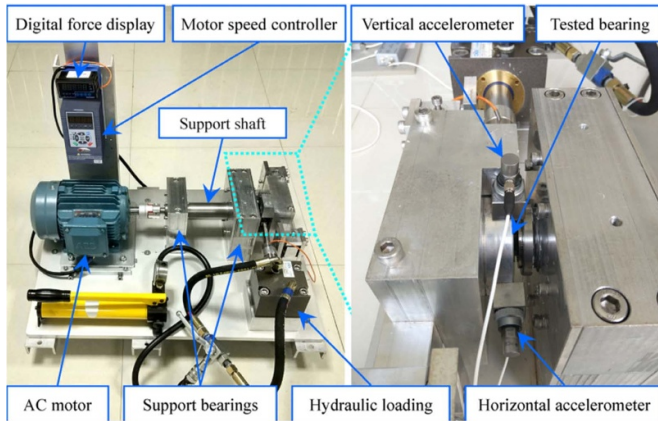


Figure 21. Data collection platform for XJTU-SY dataset.

3.2.3. Experiments and analysis

3.2.3.1. Validation of model's noise resistance performance.

To validate the comprehensive predictive performance of the proposed model under noise conditions, it is compared with other deep learning-based RUL prediction models, including CNN-LSTM, attention stacked bi-directional long short-term memory (A-SBiLSTM), and gated recurrent unit (GRU). Each experiment is conducted five times, and the mean values are taken as the final evaluation results, as shown in table 10. Figure 23 presents a comparative analysis of average prediction accuracy across different models on the XJTU-SY dataset

Table 9. Data partition for RUL prediction study.

Condition	Speed (rpm)	Load (N)	Training set	Test set
I	1650	11 000	Bearing1-1	Bearing1-2
			Bearing1-4	Bearing1-3
			Bearing1-5	
II	1500	5000	Bearing2-1	Bearing2-2
			Bearing2-4	Bearing2-3
			Bearing2-5	

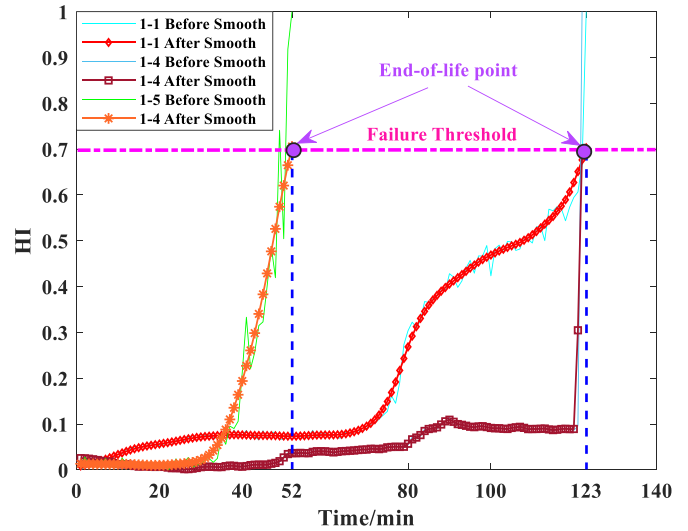


Figure 22. Feature degradation trends for the XJTU-SY dataset for condition I.

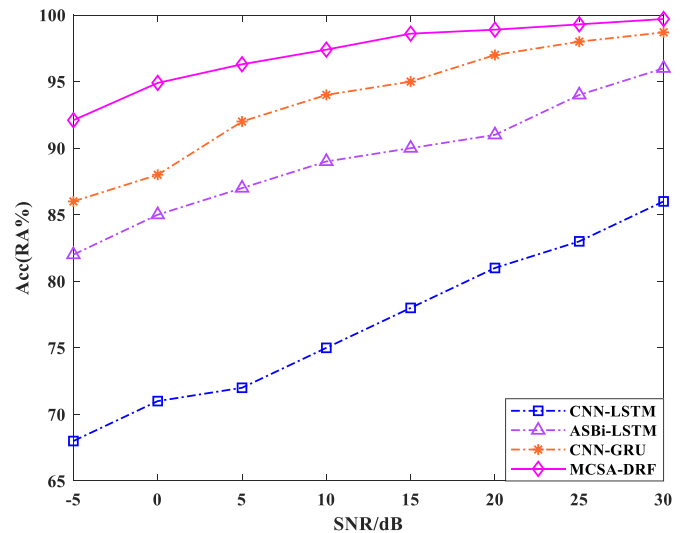


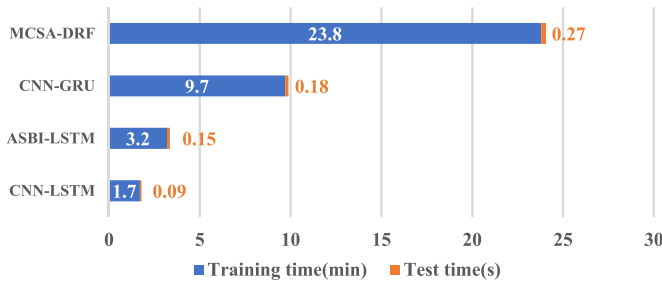
Figure 23. Comparative results of different models under salt & pepper noise.

under varying levels of salt-and-pepper noise, from signal-to-noise ratios of -5 dB to 30 dB.

It can be observed from figure 23 that the MCSA-DRF, CNN-GRU, A-SBiLSTM, and CNN-LSTM models demonstrate respective relative increases in prediction accuracy, all indicated as 8.3% , 14.78% , 17.07% , and 26.47% . Notably, the

Table 10. Average prediction results of the test set under three types of noise ranging from -5 dB to 30 dB for different models.

Model	Salt & pepper noise				Gaussian Noise				Speckle noise			
	MAE	RMSE	RA	R^2	MAE	RMSE	RA	R^2	MAE	RMSE	RA	R^2
CNN-LSTM [61]	0.75	23.61	0.7	0.71	0.68	22.27	0.78	0.81	0.71	22.01	0.75	0.78
A-SBiLSTM [62]	0.31	13.01	0.89	0.91	0.40	13.98	0.87	0.88	0.47	14.35	0.89	0.90
CNN-GRU [63]	0.29	11.21	0.93	0.94	0.31	12.76	0.91	0.94	0.25	12.90	0.91	0.93
MCSA-DRF	0.14	5.40	0.97	0.97	0.10	3.22	0.98	0.99	0.19	6.78	0.97	0.96

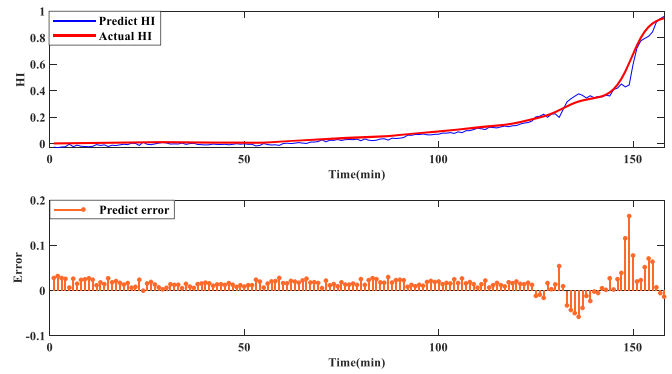
**Figure 24.** Training time and testing time of the model on the XJTU-SY dataset.

MCSA-DRF model showcases the least variation in prediction accuracy in response to noise changes, highlighting its robustness in noisy environments.

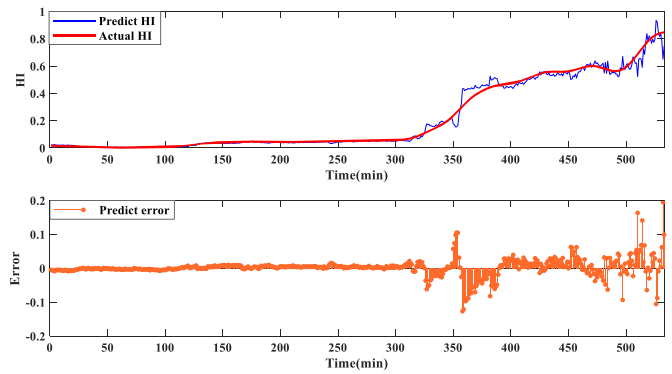
According to table 10, the CNN-LSTM model performs the worst in predicting results under three types of noise, with the highest average prediction errors $AVE_{MAE} = 0.71$ and $AVE_{RMSE} = 22.63$, and the lowest $AVE_{RA} = 0.74$ and goodness of fit $AVE_R^2 = 0.77$. Compared to CNN-LSTM, A-SBiLSTM and CNN-GRU exhibit reductions of 45.07% and 60.56% in the average prediction error AVE_{MAE} , reductions of 39.11% and 45.69% in AVE_{RMSE} , increases of 18.92% and 24.32% in average prediction accuracy AVE_{RA} , and increases of 16.88% and 22.08% in average goodness of fit AVE_R^2 . LSTM with attention mechanism and GRU with residual attention soft-threshold mechanism exhibit certain degree of noise robustness. The MCSA-DRF model achieves the best prediction metrics ($AVE_{MAE} = 0.14$, $AVE_{RMSE} = 5.13$, $AVE_{RA} = 0.97$, $AVE_R^2 = 0.97$), further elucidating that the enhanced channel and spatial attention augmentation mechanisms in the MCSA-DRF model effectively filter out redundant features and noise, enhancing the model's robustness.

The offline training time and online testing time of different models on the XJTU-SY dataset are illustrated in figure 24.

Figure 24 details the offline training and online testing times for these models on the XJTU-SY dataset. The CNN-LSTM model, owing to its simpler structure, requires the least time for both training and testing. Conversely, MCSA-DRF, with its more complex architecture, has the longest training and testing times, though the difference in online testing time between MCSA-DRF and the less time-consuming CNN-LSTM model is minimal, at just 0.18 s. This negligible difference, combined with the high accuracy of MCSA-DRF, indicates the model's efficiency in online testing scenarios across both datasets.



(a)



(b)

Figure 25. Prediction results: (a) prediction results for bearing 1-3; (b) prediction results for bearing 2-3.

3.2.3.2. Validation of the model in non-noisy conditions performance. Figure 25 showcases the results of non-noise experiments conducted on the XJTU-SY dataset, delineating predictions under diverse operating conditions. In figure 25(a), the model's predictions for the Condition I test set are presented, while figure 25(b) illustrates predictions for the Condition II test set. The model's predictions closely match actual health indicator trends, with prediction errors predominantly within the $[-0.2, +0.2]$ range, validating its capability to accurately forecast bearing RUL. Utilizing the comparative method from Case I for validation, table 11 elucidates the prediction errors for all test sets.

Table 11 provides a thorough evaluation of the performance across various models on the XJTU-SY dataset. Notably, the proposed model achieves significant reductions in MAE and RMSE, demonstrating a marked improvement over traditional

Table 11. Evaluation results of the test set.

Test set	MCSA-DRF		MCSA-CNN		GAM-CNN [58]		CBAM-MSCNN [59]		MSCNN-DRF		CNN-DRF	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Bearing1-1	0.19	12.58	0.48	14.62	0.62	17.25	0.58	15.88	0.83	22.92	1.01	32.25
Bearing1-3	0.09	8.27	0.36	12.18	0.44	13.28	0.54	15.47	0.95	28.35	0.96	28.92
Bearing2-1	0.17	9.75	0.85	23.71	0.97	30.76	0.93	27.86	1.21	35.83	1.66	41.24
Bearing2-3	0.15	10.68	0.31	11.42	0.51	15.29	0.55	15.61	0.71	19.12	0.94	28.16

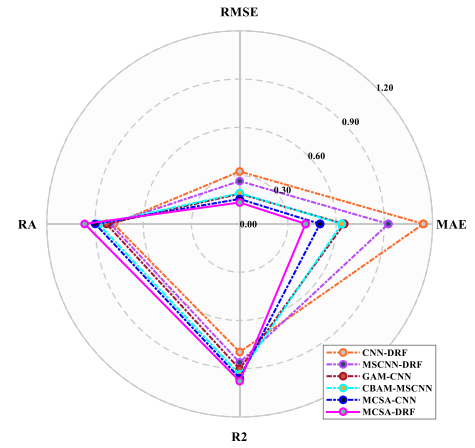
Table 12. Comparison of prediction results of XJTU-SY dataset in different models.

Model	Evaluation Indicators			
	MAE	RMSE	RA	R^2
CNN-DRF	1.143	32.643	0.784	0.796
MSCNN-CNN	0.925	26.578	0.805	0.861
GAM-CNN [58]	0.650	18.705	0.829	0.902
CBAM-MSCNN [59]	0.635	19.145	0.878	0.936
MCSA-CNN	0.500	15.483	0.899	0.958
MCSA-DRF	0.410	13.320	0.965	0.977

single-branch prediction methods without DRF, as referenced in the literature [56, 57]. These results underscore the advantage of integrating color-grayscale time-frequency features with the DRF structure, which significantly enhances the model's capability to learn from crucial degradation feature information. Additionally, when compared to MSCNN-DRF and CNN-DRF models, the proposed model shows a substantial improvement in both MAE and RMSE metrics, highlighting the effectiveness of the attention mechanism in bolstering feature extraction performance.

Table 12 delves into the specific comparative outcomes on the XJTU-SY dataset. Against the backdrop of CNN [58] and MSCNN [59] models that utilize a single-branch structure with an attention mechanism for processing color time-frequency images, the proposed approach records notable increases in prediction accuracy. This improvement evidences the enriched information content of the hybrid color-grayscale time-frequency features introduced by this study. The dual-branch structure employed for extracting these features affords a more detailed depiction of bearing degradation states. Furthermore, relative to CNN and MSCNN models lacking an attention mechanism, the proposed strategy registers significant gains in improved average prediction accuracy by 23.1% and 19.9%, validating the critical role of the attention mechanism in enhancing the model's feature discernment capabilities. Additionally, the integration of the DRF structure into MCSA not only yields an increase in prediction accuracy but also substantiates the DRF's contribution to the overall enhancement of the model's predictive accuracy.

Figure 26 shows the evaluation results of the proposed model on the XJTU-SY dataset, and the proposed model obtains the prediction indexes: $MAE_{MCSA-DRF} = 0.41$, $RMSE_{MCSA-DRF} = 13.32$, $RA_{MCSA-DRF} = 0.965$, and $R^2_{MCSA-DRF} = 0.977$. The evaluation indexes are better than

**Figure 26.** Model prediction performance evaluation chart.

the other comparative models, which further validates the effectiveness of the model.

3.3. Experimental study of RUL prediction

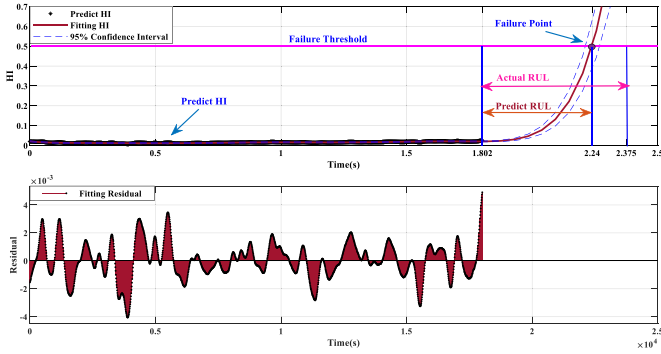
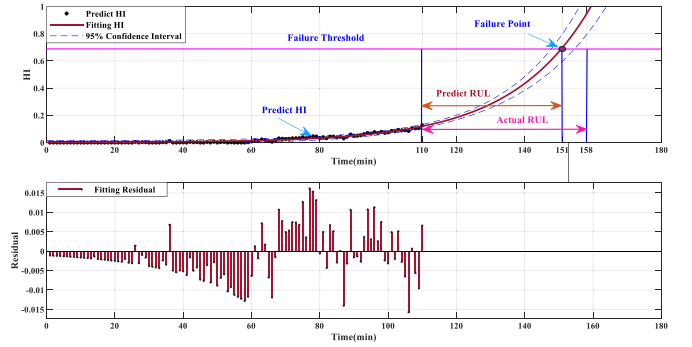
Figures 27 and 28 illustrate the RUL prediction results for the PORNOSTIA dataset and the XJTU-SY dataset for condition I, respectively. The RUL prediction methodology encompasses several stages: initially, the complete lifecycle signals from the test set bearings are truncated to represent early-stage degradation; subsequent to this truncation, time-frequency images reflecting these early degradation phases serve as inputs to the trained model for MCSA-DRF-HI prediction relevant to the truncated signals. The next phase involves leveraging a regression model to forecast the subsequent HI values, facilitating the computation of the anticipated RUL. This research employs three key evaluation metrics to gauge the regression experiments' efficacy in fitting the model predictions to actual outcomes: the sum of squares of errors (SSE), R^2 , and RMSE, alongside the RUL predictions themselves. These metrics offer a comprehensive view of the model's predictive accuracy and its ability to fit the regression model effectively to the observed data. The results are as follows:

$$SSE = \sum_{i=1}^n w_i (y_i - \hat{y}_i)^2 \quad (27)$$

where y_i is the i th actual value, $\hat{y}_i$ is the corresponding i th fitted value, and w_i is the corresponding fitted weight. SEE

Table 13. Assessment of the performance of different model fits.

Fit model	Index					
	R^2	SSE	RMSE	Number of coefficients	Predict RUL	Actual RUL
Polynomial	0.9930	0.0002	0.0025	7	143	158
Fourier	0.0985	0.0016	0.0039	4	240	
Power	0.9893	0.0011	0.0033	2	186	
Rational	0.9906	0.0010	0.0031	8	199	
Exponential	0.9964	0.0001	0.0018	2	151	

**Figure 27.** Truncated prediction results for PONORSTIA bearing 1-3.**Figure 28.** Truncated prediction results for XJTU-SY bearing 1-3.

evaluates the model fitting performance by calculating the sum of squares of the errors between the actual and fitted values. The closer SSE is to 0, the better the model fits the input data. Taking the XJTU-SY dataset as an example, the evaluation metrics for different fitted models are shown in table 13.

Table 13 reveals that the Fourier, power, and rational number fitting models exhibit delays in their predictions, resulting in predicted RULs that surpass the actual service lives of the bearings. Such inaccuracies render these models unsuitable for practical RUL prediction, where the goal is to forecast potential failures in advance to facilitate timely maintenance interventions. Contrarily, the polynomial and exponential fitting models demonstrate the capability to accurately predict failure times ahead of occurrence. Among these, the exponential fitting model stands out due to its minimal fitting error and most precise RUL prediction, showcasing the lowest fitting error, the highest coefficient of determination R^2 , and the closest alignment of predicted RUL with actual RUL. Consequently, the exponential fitting model is selected as the regression prediction model for validating RUL predictions within this research.

Figures 27 and 28 illustrate the RUL prediction results for the PRONOSTIA and XJTU-SY datasets, respectively, under condition I. The visualizations feature an orange solid line, intersected by two blue solid lines, representing the model's future HI prediction. The discrepancy in the x -axis coordinates between these blue lines denotes the predicted RUL, while the x -axis coordinates at the point where the predicted HI crosses the failure threshold indicate the estimated bearing failure time. A subsequent graph within each figure highlights the fitting error of the predicted HI, maintaining the fitting

error range of the test set within a 95% confidence interval, thus affirming the regression model's precision in fitting the predicted values accurately.

The scoring function is constructed from the relative error between the actual RUL and the predicted RUL and is defined as follows:

$$E_{ri} = 100 \times \frac{\text{ActRUL}_i - \text{PreRUL}_i}{\text{ActRUL}_i} \quad (28)$$

where ActRUL_i is the actual bearing life and PreRUL_i is the predicted RUL.

The scoring function A_i is defined as follows:

$$A_i = \begin{cases} \exp(-\ln(0.5) * (E_i/5)) & E_i \leq 0 \\ \exp(\ln(0.5) * (E_i/20)) & E_i \geq 0 \end{cases} \quad (29)$$

The predictive scoring metrics are defined below:

$$\text{Score} = \frac{1}{N} \sum_{i=1}^N A_i. \quad (30)$$

The results of the truncated RUL predictions for the PRONOSTIA and XJTU-SY datasets are presented in table 14.

During the experimental phase focused on RUL prediction, the proposed method was evaluated against four different research approaches. On the PRONOSTIA dataset, a study referenced as [59] offered a single-stream multi-scale convolution approach enhanced by an attention mechanism, achieving an average prediction error of 10.95 and a score of 0.5653. Another study [64], introduced a novel recurrent

Table 14. Evaluation of RUL prediction models.

Dataset	[64]		[65]		[57]		[59]		Proposed model	
	Mean	SCORE	Mean	SCORE	Mean	SCORE	Mean	SCORE	Mean	SCORE
PRONOSTIA	44.49	0.4384	28.18	0.4285	16.11	0.4522	10.95	0.5653	12.73	0.5761
XJTU-SY	—	—	—	—	27.84	0.4476	32.66	0.4766	9.74	0.4874

coding-decoding structure equipped with an attention mechanism that utilized five sub-band spectral energy values, also resulting in an unspecified average RUL prediction error of 44.49 and a score of 0.4384. For the XJTU-SY dataset, the research evaluated two models that incorporated both attention mechanisms and convolutional structures. Literature [59] resulted in a prediction error of 32.66 and a score of 0.4766. Literature [57], combining dilated convolution and adaptive threshold group convolution, exhibited an average prediction error of 27.84 and a score of 0.4476. The comparative analysis underscored that the method proposed in this study outperformed other models on both datasets in terms of prediction scores, thereby highlighting its superior effectiveness.

4. Conclusion

In this paper, we introduce a RUL prediction method named COA-DMCSA-DRF, aimed at bearings in noisy environments. The method begins with extracting color time-frequency feature images from vibration signals using CWT and converting these to grayscale to form a color-grey scale image dataset. It then employs enhanced CAE and SAE alongside multi-scale convolution for feature extraction from this dataset. A DRF structure is implemented at the model's backend for effective feature integration. The COA optimizes model parameters, which, combined with a regression model, enables RUL prediction. In the validation phase, incorporating improved CAE and SAE into MSCNN-DRF significantly boosted the MCSA-DRF model's prediction accuracy of 97.39% and 97.35% on both PRONOSTIA and XJTU-SY datasets under various noise conditions (−5 dB to 30 dB). This method outperformed others, showcasing superior prediction accuracy, noise resistance, and generalization performance. The advantages of the DMCSA-DRF-based RUL prediction method proposed in this paper are summarized as follows:

- (1) Given the prevalence of noise in real-world bearing data, this study proposes an enhanced channel and spatial attention mechanism aimed at amplifying essential features while reducing and filtering out unnecessary features and noise. Furthermore, a novel dual-branch multi-scale attention structure is introduced to tackle the problem of redundancy that often occurs in the multi-scale convolution process. The incorporation of a DRF structure at the model's backend significantly improves the representation of features, ensuring a more robust and effective feature extraction and integration.
- (2) In the data preprocessing phase, to mitigate the challenge of limited sample availability, we utilized CWT

and grayscale processing to create a color-grayscale time-frequency image dataset. This approach significantly enriches the input data's feature information. Additionally, during the (HI design phase, careful attention was given to the bearings' actual degradation trend and monotonicity. The resulting HI effectively captures the genuine degradation path while maintaining strong monotonicity, ensuring a reliable and accurate reflection of the bearings' condition.

- (3) To address the constraints of manually setting model parameters, we utilized the COA to conduct an optimal search for the model's hyperparameters, promoting a more scientific and logical parameter configuration. Testing revealed that the COA not only slightly surpasses the grid search algorithm in search accuracy but also significantly exceeds it in terms of search efficiency. Moreover, in the application of RUL prediction on truncated data from both the PRONOSTIA and XJTU-SY datasets, the MCSA-DRF model, utilizing the predicted MCSA-DRF-HI, achieved the highest prediction scores. This outcome serves as further evidence of the proposed model's capability and effectiveness in RUL prediction tasks.

In this study's approach to RUL prediction, the analysis was limited to data under uniform operating conditions, with the HI specifically tailored to these conditions. This approach did not address the potential applicability of the model and HI across varied operational settings. Consequently, future research endeavors will concentrate on developing methodologies to enhance the universality of models and HIs, aiming to achieve robust performance and predictive accuracy across a broader spectrum of operating conditions.

Data availability statement

The data cannot be made publicly available upon publication because they contain sensitive personal information. The data that support the findings of this study are available upon reasonable request from the authors.

Acknowledgments

This work is supported by the National Natural Science Foundation of P. R. China (61963025) (62241307), the Scientific and Technological Project of Gansu Province (22YF7FA166) (23CXGA0060), the Scientific and Technological Project of Lanzhou City (2022-RC-60) (2023-RC-26) and the Innovation Fund of the Education Department of Gansu Province(2021A-027).

Ethical compliance

All procedures performed in studies involving human participants were following the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki Declaration and its later amendments or comparable ethical standards.

Conflict of interest

All authors declare that they have NO affiliations with or involvement in any organization or entity with any financial interest in the subject matter or materials discussed in this manuscript.

ORCID iDs

Wenchao Jia  <https://orcid.org/0009-0008-6149-7635>

Yaoke Shi  <https://orcid.org/0000-0001-8533-3901>

Bin Gong  <https://orcid.org/0000-0002-5613-1419>

References

- [1] Paris P and Erdogan F 1963 A critical analysis of crack propagation laws *J. Basic Eng.* **84** 528–34
- [2] Liu X, Zhang W, Liu X, Dai W and Fu G 2019 A novel RLS-KS method for parameter estimation in particle filtering-based fatigue crack growth prognostics *IEEE Access* **7** 156764–78
- [3] Lee Y, Kim I, Choi S, Oh J and Kim N 2022 Remaining useful life prediction for PMSM under radial load using particle filter *Smart Struct. Syst.* **29** 799–805
- [4] Chen C, Liu Y, Wang S, Sun X, Di Cairano-Gilfedder C, Titmus S and Syntetos A A 2020 Predictive maintenance using cox proportional hazard deep learning *Adv. Eng. Inf.* **44** 101054
- [5] Aremu O O, Hyland-Wood D and McAree P R 2019 A relative entropy Weibull-SAX framework for health indices construction and health stage division in degradation modeling of multivariate time series asset data *Adv. Eng. Inf.* **40** 121–34
- [6] Lee S, Lee S, Lee K, Lee S, Chung J, Kim C-W and Yoon J 2021 Data-driven health condition and RUL prognosis for liquid filtration systems *J. Mech. Sci. Technol.* **35** 1597–607
- [7] Zhai Q and Ye Z S 2017 RUL prediction of deteriorating products using an adaptive Wiener process model *IEEE Trans. Ind. Inform.* **13** 2911–21
- [8] Xiahou T, Zeng Z and Liu Y 2020 Remaining useful life prediction by fusing expert knowledge and condition monitoring information *IEEE Trans. Ind. Inform.* **17** 2653–63
- [9] Hectors K, Van Hecke T and De Waele W 2022 Method for statistical evaluation of cumulative damage models applied to block loading *Fatigue Fracture Eng. Mater. Struct.* **45** 3319–32
- [10] Weber F, Wu H and Starke P 2023 A new short-time procedure for fatigue life evaluation based on the linear damage accumulation by Palmgren–Miner *Int. J. Fatigue* **172** 107653
- [11] Cerfontaine B and Collin F 2018 Cyclic and fatigue behaviour of rock materials: review, interpretation and research perspectives *Rock Mech. Rock Eng.* **51** 391–414
- [12] Zhu S P, Liu Q, Peng W and Zhang X-C 2018 Computational-experimental approaches for fatigue reliability assessment of turbine bladed disks *Int. J. Mech. Sci.* **142** 502–17
- [13] Sanhen N E and Braun M 2023 Fatigue life prediction of welded joints at sub-zero temperatures using modified Paris–Erdogan parameters *Theor. Appl. Fract. Mech.* **127** 104088
- [14] Dave H B, Singh D and Bansal H O 2021 Multiple linear regression-based impact analysis of impedance network design on life expectancy of DC-link capacitor in q-ZSI fed motor drive *Eng. Sci. Technol. Int. J.* **24** 171–82
- [15] Feng Y, Liu Q, Yao Q and Zhao G 2022 Model averaging for nonlinear regression models *J. Bus. Econ. Stat.* **40** 785–98
- [16] Wang F, Tang S, Sun X, Li L, Yu C and Si X 2023 Remaining useful life prediction based on nonlinear random coefficient regression model with fusing failure time data *J. Syst. Eng. Electron.* **34** 247–58
- [17] Diyin T, Jinrong C A O and Jinsong Y U 2019 Remaining useful life prediction for engineering systems under dynamic operational conditions: a semi-Markov decision process-based approach *Chin. J. Aeronaut.* **32** 627–38
- [18] Liu T, Zhu K and Zeng L 2018 Diagnosis and prognosis of degradation process via hidden semi-Markov model *IEEE/ASME Trans. Mechatronics* **23** 1456–66
- [19] Hu J and Chen P 2020 Predictive maintenance of systems subject to hard failure based on proportional hazards model *Reliab. Eng. Syst. Saf.* **196** 106707
- [20] Zhiyong G, Jiwu L and Rongxi W 2021 Prognostics uncertainty reduction by right-time prediction of remaining useful life based on hidden Markov model and proportional hazard model *Eksploracja i Niezawodnosc* **23** 154–64
- [21] Si X S, Wang W, Hu C H, Chen M-Y and Zhou D-H 2013 A Wiener-process-based degradation model with a recursive filter algorithm for remaining useful life estimation *Mech. Syst. Signal Process.* **35** 219–37
- [22] Zhang S, Zhai Q and Li Y 2023 Degradation modeling and RUL prediction with Wiener process considering measurable and unobservable external impacts *Reliab. Eng. Syst. Saf.* **231** 109021
- [23] Wang H, Liao H, Ma X and Bao R 2021 Remaining useful life prediction and optimal maintenance time determination for a single unit using isotonic regression and gamma process model *Reliab. Eng. Syst. Saf.* **210** 107504
- [24] Wang H, Xu T and Mi Q 2015 Lifetime prediction based on Gamma processes from accelerated degradation data *Chin. J. Aeronaut.* **28** 172–9
- [25] Saon S and Hiyama T 2010 Predicting remaining useful life of rotating machinery based artificial neural network *Comput. Math. Appl.* **60** 1078–87
- [26] Nagulapati V M, Lee H, Jung D W, Brigljevic B, Choi Y and Lim H 2021 Capacity estimation of batteries: influence of training dataset size and diversity on data driven prognostic models *Reliab. Eng. Syst. Saf.* **216** 108048
- [27] Shen F and Yan R 2021 A new intermediate-domain SVM-based transfer model for rolling bearing RUL prediction *IEEE/ASME Trans. Mechatronics* **27** 1357–69
- [28] Yao F, He W, Wu Y, Ding F and Meng D 2022 Remaining useful life prediction of lithium-ion batteries using a hybrid model *Energy* **248** 123622
- [29] Zhu R, Chen Y, Peng W and Ye Z-S 2022 Bayesian deep-learning for RUL prediction: an active learning perspective *Reliab. Eng. Syst. Saf.* **228** 108758
- [30] Jiang G J, Yang J S, Cheng T C and Sun H-H 2023 Remaining useful life prediction of rolling bearings based on Bayesian neural network and uncertainty quantification *Qual. Reliab. Eng. Int.* **39** 1756–74
- [31] Ellefsen A L, Bjørlykhaug E, Æsøy V, Ushakov S and Zhang H 2019 Remaining useful life predictions for

- turbofan engine degradation using semi-supervised deep architecture *Reliab. Eng. Syst. Saf.* **183** 240–51
- [32] Yang X, Zheng Y, Zhang Y, Ma Q and Fu J 2022 Bearing remaining useful life prediction based on regression shapalet and graph neural network *IEEE Trans. Instrum. Meas.* **71** 1–12
- [33] Zhang S, Zhai B, Guo X, Wang K, Peng N and Zhang X 2019 Synchronous estimation of state of health and remaining useful lifetime for lithium-ion battery using the incremental capacity and artificial neural networks *J. Energy Storage* **26** 100951
- [34] Ordóñez C, Lasheras F S, Roca-Pardiñas J and Juez F J D C 2019 A hybrid ARIMA–SVM model for the study of the remaining useful life of aircraft engines *J. Comput. Appl. Math.* **346** 184–91
- [35] Bai R, Noman K, Feng K, Peng Z and Li Y 2023 A two-phase-based deep neural network for simultaneous health monitoring and prediction of rolling bearings *Reliab. Eng. Syst. Saf.* **238** 109428
- [36] Xiang S, Zhou J, Luo J, Liu F and Qin Y 2023 Cocktail LSTM and its application into machine remaining useful life prediction *IEEE/ASME Trans. Mechatronics* **28** 2425–36
- [37] Zhou J, Qin Y, Luo J, Wang S and Zhu T 2022 Dual-thread gated recurrent unit for gear remaining useful life prediction *IEEE Trans. Ind. Inform.* **19** 8307–18
- [38] Zhou J, Qin Y, Chen D, Liu F and Qian Q 2022 Remaining useful life prediction of bearings by a new reinforced memory GRU network *Adv. Eng. Inf.* **53** 101682
- [39] Shaw J and Wu B 2021 Prediction of remaining useful life of wind turbine shaft bearings using machine learning *J. Mar. Sci. Technol.* **29** 631–7
- [40] Cheng C, Ma G, Zhang Y, Sun M, Teng F, Ding H and Yuan Y 2020 A deep learning-based remaining useful life prediction approach for bearings *IEEE/ASME Trans. Mechatronics* **25** 1243–54
- [41] Yang B, Liu R and Zio E 2019 Remaining useful life prediction based on a double-convolutional neural network architecture *IEEE Trans. Ind. Electron.* **66** 9521–30
- [42] Xie Q, Liu R, Huang J and Su J 2023 Residual life prediction of lithium-ion batteries based on data preprocessing and a priori knowledge-assisted CNN-LSTM *Energy* **281** 128232
- [43] Wang Y and Wang Y 2023 A denoising semi-supervised deep learning model for remaining useful life prediction of turbofan engine degradation *Appl. Intell.* **53** 1–18
- [44] Liu H, Liu Z, Zhang D, Jia W, Xin X and Tan J 2024 Uncertainty quantification and interval prediction of equipment remaining useful life based on semisupervised learning *IEEE Trans. Instrum. Meas.* **73** 3506315
- [45] Xiao L, Duan F, Tang J and Della Santina C C 2021 A noise-booster remaining useful life prediction method for rotating machines under different conditions *IEEE Trans. Instrum. Meas.* **70** 1–12
- [46] Xiao L, Tang J, Zhang X, Bechhoefer E and Ding S 2021 Remaining useful life prediction based on intentional noise injection and feature reconstruction *Reliab. Eng. Syst. Saf.* **215** 107871
- [47] Zhao C, Huang X, Li Y and Li S 2022 A novel remaining useful life prediction method based on gated attention mechanism capsule neural network *Measurement* **189** 110637
- [48] Wang H, Yang J, Wang R and Shi L 2023 Remaining useful life prediction of bearings based on convolution attention mechanism and temporal convolution network *IEEE Access* **11** 24407–19
- [49] Qin Y, Zhou J and Chen D 2021 Unsupervised health indicator construction by a novel degradation-trend-constrained variational autoencoder and its applications *IEEE/ASME Trans. Mechatronics* **27** 1447–56
- [50] Zhou J, Qin Y, Luo J and Zhu T 2022 Remaining useful life prediction by distribution contact ratio health indicator and consolidated memory GRU *IEEE Trans. Ind. Inform.* **19** 8472–83
- [51] Jia H, Rao H, Wen C and Mirjalili S 2023 Crayfish optimization algorithm *Artif. Intell. Rev.* **56** 1919–79
- [52] Woo S et al 2018 Cbam: convolutional block attention module *Proc. European Conf. on Computer Vision (ECCV)* pp 3–19
- [53] Hu Y, Li J, Huang Y and Gao X 2019 Channel-wise and spatial feature modulation network for single image super-resolution *IEEE Trans. Circuits Syst. Video Technol.* **30** 3911–27
- [54] Xu Q, Jiang H, Zhang X, Li J and Chen L 2023 Multiscale convolutional neural network based on channel space attention for gearbox compound fault diagnosis *Sensors* **23** 3827
- [55] Nectoux P, Gouriveau R, Medjaher K, Ramasso E, Chebel-Morello B, Zerhouni N and Varnier C 2012 PRONOSTIA: an experimental platform for bearings accelerated degradation tests *IEEE Int. Conf. on Prognostics and Health Management, PHM'12. IEEE Catalog Number: CPF12PHM-CDR* pp 1–8
- [56] Luo H, Bo L, Liu X and Zhang H 2021 A novel method for remaining useful life prediction of roller bearings involving the discrepancy and similarity of degradation trajectories *Comput. Intell. Neurosci.* **2021** 1–26
- [57] Gong B, An A and Shi Y 2023 Photovoltaic arrays fault diagnosis based on an improved dilated convolutional neural network with feature-enhancement *Meas. Sci. Technol.* **35** 015011
- [58] Du X, Jia W, Yu P, Shi Y and Gong B 2023 RUL prediction based on GAM–CNN for rotating machinery *J. Braz. Soc. Mech. Sci. Eng.* **45** 142
- [59] Du X, Jia W, Yu P, Shi Y and Cheng S 2022 A remaining useful life prediction method based on time–frequency images of the mechanical vibration signals *Measurement* **202** 111782
- [60] Lei Y, Han T and Wang B 2019 XJTU-SY rolling element bearing accelerated life test datasets: a tutorial *J. Mech. Eng.* **55** 1–6
- [61] Shaw J and Wu B J 2023 Artificial intelligence in prediction of the remaining useful life of wind turbine shaft bearings *J. Mar. Sci. Technol.* **31** 12
- [62] Rathore M S and Harsha S P 2022 An attention-based stacked BiLSTM framework for predicting remaining useful life of rolling bearings *Appl. Soft Comput.* **131** 109765
- [63] Zhang L, Wang B, Yuan X and Liang P 2022 Remaining useful life prediction via improved CNN, GRU and residual attention mechanism with soft thresholding *IEEE Sens. J.* **22** 15178–90
- [64] Hinch A Z and Tkouat M 2018 Rolling element bearing remaining useful life estimation based on a convolutional long-short-term memory network *Proc. Comput. Sci.* **127** 123–32
- [65] Lei Y, Li N, Gontarz S, Lin J, Radkowski S and Dybala J 2016 A model-based method for remaining useful life prediction of machinery *IEEE Trans. Reliab.* **65** 1314–26

PAPER

An efficient lightweight gearbox fault diagnosis method based on parameter optimization and adaptive attention enhancement across working conditions

To cite this article: Wenchao Jia *et al* 2025 *Meas. Sci. Technol.* **36** 056109

View the [article online](#) for updates and enhancements.

You may also like

- [Planetary gearbox fault diagnosis based on FDKNN-DGAT with few labeled data](#)
Hongfeng Tao, Haojin Shi, Jier Qiu *et al.*
- [A meta transfer learning fault diagnosis method for gearbox with few-shot data](#)
Zhichao Yang, Yudan Duan, Daoming She *et al.*
- [Deep subclass alignment transfer network based on time–frequency features for intelligent fault diagnosis of planetary gearboxes under time-varying speeds](#)
Songjun Han and Zhipeng Feng



ECS The Electrochemical Society
Advancing solid state & electrochemical science & technology

247th ECS Meeting

Montréal, Canada
May 18-22, 2025
Palais des Congrès de Montréal

ECS UNITED

Unite with the ECS Community

**Register to
save \$\$
before
May 17**

An efficient lightweight gearbox fault diagnosis method based on parameter optimization and adaptive attention enhancement across working conditions

Wenchao Jia¹ , Aimin an^{1,3,*}, Bin Gong¹ , Yaoke Shi²  and Zhipeng Luo¹

¹ College of Electrical and Information Engineering, Lanzhou University of Technology, Lan-zhou 730050, People's Republic of China

² School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, People's Republic of China

³ Key Laboratory of Gansu Advanced Control for Industrial Processes, Lanzhou University of Technology, Lanzhou 730050, People's Republic of China

E-mail: anaiminll@163.com

Received 9 December 2024, revised 30 March 2025

Accepted for publication 15 April 2025

Published 23 April 2025



Abstract

Currently, most mainstream gearbox fault diagnosis methods rely heavily on deep learning techniques. However, such deep learning-based diagnostic models often face challenges such as high computational resource requirements, poor real-time performance, and low diagnostic accuracy when applied to other operating conditions. To address the aforementioned issues, a novel lightweight cross-operating-condition gearbox transfer fault diagnosis framework is proposed, incorporating parameter optimization and adaptive feature attention enhancement. The framework utilizes two-dimensional Gramian angular field images to extract deep gearbox fault information. An improved lightweight mechanism integrating multi-scale channel and spatial adaptive attention enhancement is first adopted to allocate feature weights. This mechanism amplifies critical features while effectively mitigating the influence of redundant or noisy features. Subsequently, a local maximum mean discrepancy loss is employed to achieve cross-domain feature alignment across different operating conditions. Finally, a multi-strategy improved triangular topology optimization algorithm is applied to fine-tune the model's hyperparameters, boosting its cross-condition diagnostic capabilities and ensuring more accurate fault detection across different operational scenarios. Experimental results indicate that the proposed model achieved accuracy rates of 99.85% and 99.94% on two validation datasets, while maintaining diagnostic accuracies of 74.6% and 79.6% in the presence of significant noise. The model's diagnostic performance and robustness are superior to those of other models, demonstrating its effectiveness in challenging conditions.

Keywords: gearbox fault diagnosis, lightweight structure, feature attention enhancement, gram angle field images, parameter optimization, unsupervised domain adaptation

* Author to whom any correspondence should be addressed.

1. Introduction

As a renewable and clean energy source, wind power's contribution to the global energy landscape is progressively increasing [1, 2]. Nevertheless, the wind turbine gearbox, being a pivotal transmission component, is susceptible to failures that can cause severe economic consequences and a decline in power generation efficiency. Therefore, the establishment of robust and dependable gearbox fault diagnosis techniques is vital for enhancing the stability of wind turbine operations [3, 4].

Currently, with the increasing availability of equipment monitoring data, the mainstream research methods in fault diagnosis have transitioned from expert knowledge-based systems [5, 6] and physical models [7] to data-driven approaches. However, fault diagnosis methods relying on expert knowledge systems and physical models are highly dependent on domain-specific expertise, making them less adaptable to complex operating conditions.

Fault diagnosis methods based on data-driven approaches mainly depend on historical and real-time monitoring data, without requiring the system's physical mechanisms or expert knowledge. These techniques have increasingly become the dominant approach in fault diagnosis. Data-driven methods are generally divided into signal analysis techniques [8], shallow machine learning methods [9], and deep learning methods [10]. Signal analysis methods extract relevant features from the monitoring signals (e.g. vibration, current, and temperature signals) obtained from sensors through signal processing, facilitating fault diagnosis [11]. Miao *et al* [12] proposed a novel feature mode decomposition method for extracting mechanical equipment fault signal features, enabling accurate decomposition of fault characteristic information. Cheng *et al* [13] introduced a signal analysis technique based on maximum Ramanujan spectrum signal-to-noise ratio deconvolution, which overcomes the limitations of traditional deconvolution methods that struggle to achieve optimal denoising results. Fault diagnosis methods based on signal analysis are capable of extracting time-frequency features from signals. However, their generalization ability is constrained by factors such as sensitivity to noise, a strong reliance on manual feature selection, and challenges in setting parameters for decomposition algorithms. In contrast, shallow machine learning models can automatically learn intricate patterns and features from the data during training, eliminating the need for predefined fault classification rules and reducing the reliance on human experience [14]. Dhiman *et al* [15] proposed a fault detection method for wind turbine gearboxes based on adaptive thresholds combined with twin support vector machines (SVM) which accurately identified gearbox faults in a wind farm in the UK. Compared to signal analysis methods, shallow machine learning-based fault diagnosis methods further reduce the dependence on human expertise and improve diagnostic accuracy. However, shallow machine learning has certain limitations in fault diagnosis: (1) its accuracy heavily relies on the construction and extraction of feature sets; and (2) the nonlinear representation capacity of shallow machine

learning models is limited, making it challenging to fully uncover the complex mapping between sensitive features and fault categories, especially in conditions where vibration signals are weak and fault features are coupled.

Recently, deep learning has gained widespread adoption in the field of fault diagnosis, driven by its exceptional capacity for feature extraction and the ability to handle complex nonlinear mappings [16–18]. Yang *et al* [19] leveraged the strong feature learning abilities of the deep belief network to accurately identify gearbox faults. Zhang *et al* [20] introduced a gearbox fault diagnosis approach utilizing the discriminative sparse autoencoder, which is aimed at addressing complex vibration signals. The convolutional neural network (CNN) extracts features from signals autonomously via its local receptive field architecture, thus substantially minimizing the reliance on feature engineering [21]. Considering the irreversible information loss caused by excessive downsampling in traditional convolutional networks, Zhao *et al* [22] proposed a CNN model based on mixed information, which employs inter-channel connections to facilitate information exchange between different channels and utilizes residual connections to prevent information loss. Long short-term memory (LSTM) networks [23] and gated recurrent units (GRU) [24] are capable of capturing dynamic variations and long-term dependencies in data, offering advantages in time series fault signal analysis. Considering that gearboxes typically operate under variable speed conditions, Chen *et al* [25] proposed an LSTM model that integrates gearbox speed and vibration signals, enabling fault diagnosis of gearboxes under variable speed conditions. Additionally, some studies have employed Transformer models, which efficiently capture global dependencies and are particularly suited for modeling large-scale data, for fault feature extraction [26]. However, although deep learning methods have shown good performance in certain fault diagnosis tasks, they still face practical challenges, including high computational complexity [27–29], limited feature representation capability [30–32], and insufficient adaptability across varying operating conditions [33–35]. Specifically, CNN architectures primarily rely on local convolution operations, which hampers their ability to capture global features, thereby limiting their representation of complex vibration signals. As such, improving feature extraction capabilities while minimizing computational complexity remains a key challenge in current fault diagnosis research.

In response to the challenges of high computational complexity and limited feature expression capacity in deep learning models for fault diagnosis, researchers have incorporated the attention mechanism. This mechanism enables the model to automatically focus on critical feature regions associated with the fault during the input data processing, thereby effectively overcoming the limitations of traditional CNN models in capturing global features. The most representative models in channel attention mechanisms, squeeze-and-excitation network [36] and efficient channel attention [37], primarily enhance feature representation by learning the importance weights of different channels. The spatial attention mechanism, gather-excite networks [38], enhances the model's focus

on local key features by capturing the feature distribution information along the spatial dimension. Some studies have combined channel attention and spatial attention into a hybrid attention mechanism to simultaneously strengthen the representation of both the channel and spatial locations of key features, such as the convolutional block attention module (CBAM) [39] and channel-wise and spatial attention residual [40]. However, existing attention mechanisms still have the following limitations: (1) Single attention mechanisms focus solely on either channel or spatial factors, neglecting the distribution features of the other, which results in insufficient local feature representation; (2) Hybrid attention combining channel and spatial attention requires additional computations for both, thereby increasing computational complexity. Therefore, designing an efficient and lightweight spatial-channel fusion attention mechanism that improves feature representation while maintaining computational efficiency remains a pressing challenge.

The distribution of gearbox fault data is highly influenced by changes in operating conditions, which has led to the widespread use of transfer learning (TL) techniques to enhance diagnostic capabilities across varying operating conditions [41–43]. TL fundamentally aims to identify common feature representations between multiple related but distinct tasks, intending to apply these shared features to new tasks [44, 45]. Zhang *et al* [46] developed a dynamic conditional adversarial domain adaptation model for gearbox fault diagnosis. By aligning the feature distributions of source and target domain data, this model mitigates the performance loss associated with cross-condition transfer. Ma *et al* [47] proposed a fault diagnosis method for wind turbine gearboxes based on deep adversarial transfer neural networks. This approach employs a domain adversarial strategy to address data distribution differences, transferring features learned from auxiliary datasets to the target dataset, which reduces the reliance on labeled data and facilitates efficient cross-domain fault diagnosis in similar domains. Despite the advancements in TL, current methods continue to struggle with issues such as complex model structures, unstable distribution alignment, and high computational overhead, hindering their practical application in industrial settings. As a result, a critical challenge in the field of fault diagnosis is how to reduce model complexity while improving cross-condition generalization performance.

The choice of hyperparameters is critical to a model's diagnostic accuracy, robustness, and generalization ability during training. Nevertheless, hyperparameter optimization strategies aimed at improving model performance are often neglected. A carefully selected hyperparameter configuration can enhance diagnostic efficiency, boost accuracy, and improve the model's adaptability in practical applications. To tackle this issue, Huang *et al* [48] proposed a hyperparameter tuning approach for CNN-LSTM models based on grid search (GS), demonstrating substantial performance improvements. However, as the search space expands, GS becomes increasingly computationally expensive. To address this, Wang *et al* [49] introduced a Bayesian optimization-based fault diagnosis method for SVM, optimizing the penalty factor and kernel

function parameters to enhance model performance. Similarly, Zhao *et al* [50] developed a competitive group optimization algorithm aimed at refining hyperparameter selection in autoencoder-based models. Additionally, studies [51] indicate that heuristic optimization methods provide an effective means of balancing diagnostic accuracy and computational efficiency in hyperparameter tuning.

Although substantial progress has been made in gearbox fault diagnosis in current research, challenges such as high model complexity, restricted feature extraction capabilities, low efficiency in hyperparameter optimization, and poor adaptability across different operating conditions remain. To overcome these obstacles, this paper introduces an efficient, lightweight gearbox fault diagnosis approach that incorporates parameter optimization and adaptive attention enhancement. The key contributions of this paper are as follows:

- (1) The Gramian angular field (GAF) transformation is applied to convert vibration signals into Gramian angular sum/difference field (GASF/GADF) images, enabling the CNN to effectively extract time-series patterns and enhance the discriminability of features.
- (2) A lightweight dual-branch MCSA (DMCSA) feature extraction network is introduced, which strengthens the expression of fault features and mitigates issues of feature redundancy and noise. Additionally, a dense residual feature fusion module (DRF) is used to reduce the parameter count and effectively integrate and utilize GASF-GADF features.
- (3) A multi-strategy enhanced TTAO is developed to improve the efficiency and performance of model hyperparameter search, preventing the model from getting trapped in local optima.
- (4) An unsupervised domain adaptation-based TL strategy is introduced to strengthen the model's generalization ability in cross-condition scenarios, reducing reliance on labeled data from the target domain.

2. Theoretical background

2.1. Triangulation topology aggregation optimizer (TTAO)

The TTAO algorithm is a method that combines mathematical characteristics with stochastic search algorithms [52]. The algorithm iteratively generates new triangle vertices in the search space and forms similar triangles of varying sizes based on these vertices. This process aggregates and collects vertices with valuable information from different topology units, ultimately solving the optimization problem. TTAO consists of two key steps: general aggregation and local aggregation, which correspond to the development and exploration phases of heuristic algorithms. General aggregation produces new nodes by exchanging positive information between different triangular topology units, while local aggregation constructs new topological units at optimal positions based on the best vertex of each unit.

2.1.1. Improved triangular topology optimization algorithm (ITTAO). Although the TTAO algorithm demonstrates excellent optimization performance, it suffers from issues such as a tendency to get trapped in local optima and low convergence precision during the development and exploration stages. Therefore, improvements are made to these phases of the algorithm.

2.1.1.1. The adaptive collaborative group weight dynamic balancing strategy. In the original TTAO algorithm, the update of new triangle vertices is performed by applying random linear weights to the three existing vertices, without adequately considering factors such as coordination and dynamic adjustment between the vertices. This limitation restricts the algorithm's ability to explore the solution space effectively in the early stages. The original vertex update formula of the TTAO algorithm is as follows:

$$\vec{X}_{i,4} = r_1 * \vec{X}_{i,1} + r_2 * \vec{X}_{i,2} + (1 - r_1 - r_2) * \vec{X}_{i,3} \quad (1)$$

where r_1 and r_2 represent random numbers selected from the interval $[0,1]$.

The formula for the adaptive collaborative elite weight dynamic balancing strategy is described as follows:

$$v = 0.2 - 0.2 * (t/T) \quad (2)$$

$$A = 2 * v * r_1 - v \quad (3)$$

$$C = 2 * r_2 \quad (4)$$

$$\begin{cases} \vec{X}_{i,11} = \vec{X}_{i,1} - A * \left(C * \vec{X}_{i,1} - \vec{X}_{i,2} \right) \\ \vec{X}_{i,21} = \vec{X}_{i,2} - A * \left(C * \vec{X}_{i,2} - \vec{X}_{i,3} \right) \\ \vec{X}_{i,31} = \vec{X}_{i,3} - A * \left(C * \vec{X}_{i,3} - \vec{X}_{i,1} \right) \end{cases} \quad (5)$$

$$\begin{cases} w_1 = 0.5 * (1 - t/T) \\ w_2 = 0.3 * (1 - t/T) \\ w_3 = 1 - w_1 - w_2 \end{cases} \quad (6)$$

$$X_{i,4} = w_1 * \vec{X}_{i,11} + w_2 * \vec{X}_{i,21} + w_3 * \vec{X}_{i,31} \quad (7)$$

where A defines the degree of displacement relative to the domain points of the individuals, while parameter C serves to diminish the tendency for individuals to converge. The adaptive cooperative group weight dynamic balancing strategy effectively utilizes the information-sharing mechanism between individuals, preventing excessive reference to a single vertex and improving the exploration efficiency at the early stage of the algorithm.

2.1.1.2. A global search mechanism based on the Cauchy inverse cumulative distribution (CICD). To overcome the limitations of traditional development phases (general aggregation), where the algorithm often falls into local optima, this paper proposes an adaptive global exploration mechanism based on the CICD. Unlike the conventional Gaussian distribution, the Cauchy distribution exhibits heavy-tailed characteristics, allowing the generation of more sampling points that are far from the current solution within the search space. This enhances the global jumping ability of the search algorithm, particularly in complex multimodal functions. Additionally, by incorporating the inverse cumulative distribution function of the probability distribution, the algorithm can adaptively adjust the search scope, facilitating a progressive global exploration from coarse to fine granularity at different stages of iteration. This mechanism not only strengthens the diversity of the population but also significantly improves the algorithm's ability to escape local extrema, providing a more robust theoretical foundation for the search for the global optimal solution [53]. The Cauchy inverse cumulative distribution function is described as follows:

$$F^{-1}(p; a, b) = a + b \cdot \tan \left(\pi \cdot \left(p - \frac{1}{2} \right) \right) \quad (8)$$

where a represents the location parameter, b is the scale parameter, and p denotes the given probability.

The updated formula after improvement is as follows:

$$\begin{aligned} \vec{X}_{i,\text{new1}}^{t+1} &= r_4 * \vec{X}_{i,\text{best}}^t + (1 - r_4) * \vec{X}_{\text{rand,best}}^t \cdot \lambda_{\text{CICD}} \\ \lambda_{\text{CICD}} &= 0.01 \cdot \tan \left(\pi \cdot \left(p - \frac{1}{2} \right) \right). \end{aligned} \quad (9)$$

The parameters a and b are set according to [53]. Without the CICD operator, the update step size of the triangular topological unit remains relatively stable, with a limited range of step sizes. However, with the introduction of the CICD operator, the step size becomes more flexible and can undergo significant changes in certain dimensions, making the movement of the topological unit within the search space more random and diverse.

The step size variation of the CICD operator is shown in figure 1, with the dimensionality set to 500.

Figure 1 illustrates the step size of the CICD operator. The randomness and oscillation of the path arise from the step sizes generated by the CICD, with significant step size variations in certain areas. These large step sizes aid the optimization algorithm in escaping local optima, enabling the exploration of different regions within the search space.

2.1.1.3. Local aggregation strategy based on the nonlinear tangential flight operator. To address the issue of premature convergence during the exploration phase (local aggregation), this paper introduces an innovative Tangent-based nonlinear flight perturbation operator (TNFPO). This operator is inspired

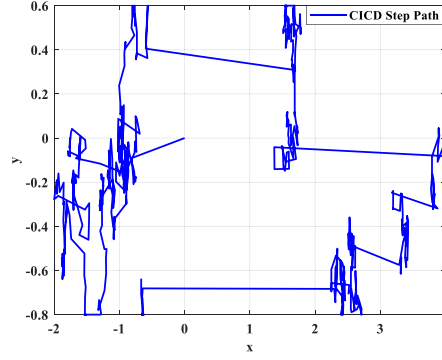


Figure 1. The step size of the CICD operator.

Algorithm 1. Improved triangulation topology aggregation optimizer pseudo-code.

Input: Initialization population N , iterations T , dimension D

Output: The optimal position $\vec{X}_{best}$ and its fitness value f_{best}

Initialize the triangle population

While $t < T$

(Formation of triangular topological units)

Generate three other positions using the Eq described below.

$$\vec{X}_{i,2} = \vec{X}_{i,1} + l * \vec{f}(\theta)$$

$$\vec{X}_{i,3} = \vec{X}_{i,1} + l * \vec{f}\left(\theta + \frac{\pi}{3}\right)$$

$$X_{i,4} = w_1 * \vec{X}_{i,1} + w_2 * \vec{X}_{i,2} + w_3 * \vec{X}_{i,3}$$

(Generic aggregation)

Update positions using Eq: $\vec{X}_{i,new1}^{t+1} = r_4 * \vec{X}_{i,best}^t + (1 - r_4) * \vec{X}_{rand,best}^t * \lambda_{CICD}$

If $f_{\vec{X}_{i,new1}^{t+1}} < f_{\vec{X}_{i,best}^t}$ **do**

$$\vec{X}_{i,best}^{t+1} = \vec{X}_{i,new1}^{t+1}$$

else if $f_{\vec{X}_{i,new1}^{t+1}} < f_{\vec{X}_{i,sbest}^t}$ **do**

$$\vec{X}_{i,sbest}^{t+1} = \vec{X}_{i,new1}^{t+1}$$

end if

(Local aggregation)

Update positions using Eq: $X_{i,new2}^{t+1} = X_{i,best}^{t+1} + \delta * (X_{i,best}^{t+1} - X_{i,sbest}^{t+1}) * \tan\left(\frac{\pi}{2} * rand(1, dim())\right)$

If $f_{X_{i,new2}^{t+1}} < f_{X_{i,best}^{t+1}}$ **do**

$$\vec{X}_{i,best}^{t+1} = X_{i,new2}^{t+1}$$

end if

Select the optimal population position X_{best} according to fitness values

end while

Return $\vec{X}_{best}$ and f_{best}

by the Levy flight model and leverages the nonlinear characteristics of the tangent function to simulate random transition behaviors. It enables individuals to perform both large and small-scale random perturbations near local optima, facilitating a non-uniform expansion of the local search area [54]. The formula for the forward flight operator is presented as follows:

$$f = \tan\left(\frac{\pi}{2} * rand(1, dim())\right) \quad (10)$$

where ‘dim’ represents the dimensionality of the problem.

The pseudocode of the TTAO algorithm, with improvements from multiple strategies, is given below:

2.1.2. Performance evaluation of improvement strategy 1.

To evaluate the impact of Improvement Strategy 1, which employs an adaptive collaborative group-weight dynamic balancing strategy, on the optimization performance of the TTAO algorithm, an ablation study was conducted. This study involved comparative analyzes using both an unimodal test function (F1) and a multimodal test function (F10). The performance of the original TTAO and the improved TTAO-Strategy 1 was assessed under identical conditions, with a population size of 100 and a maximum of 500 iterations. The experimental results are presented in figure 2.

As depicted in figure 2(b), TTAO-Strategy 1 (pink solid line) achieves a faster convergence rate in the early stage

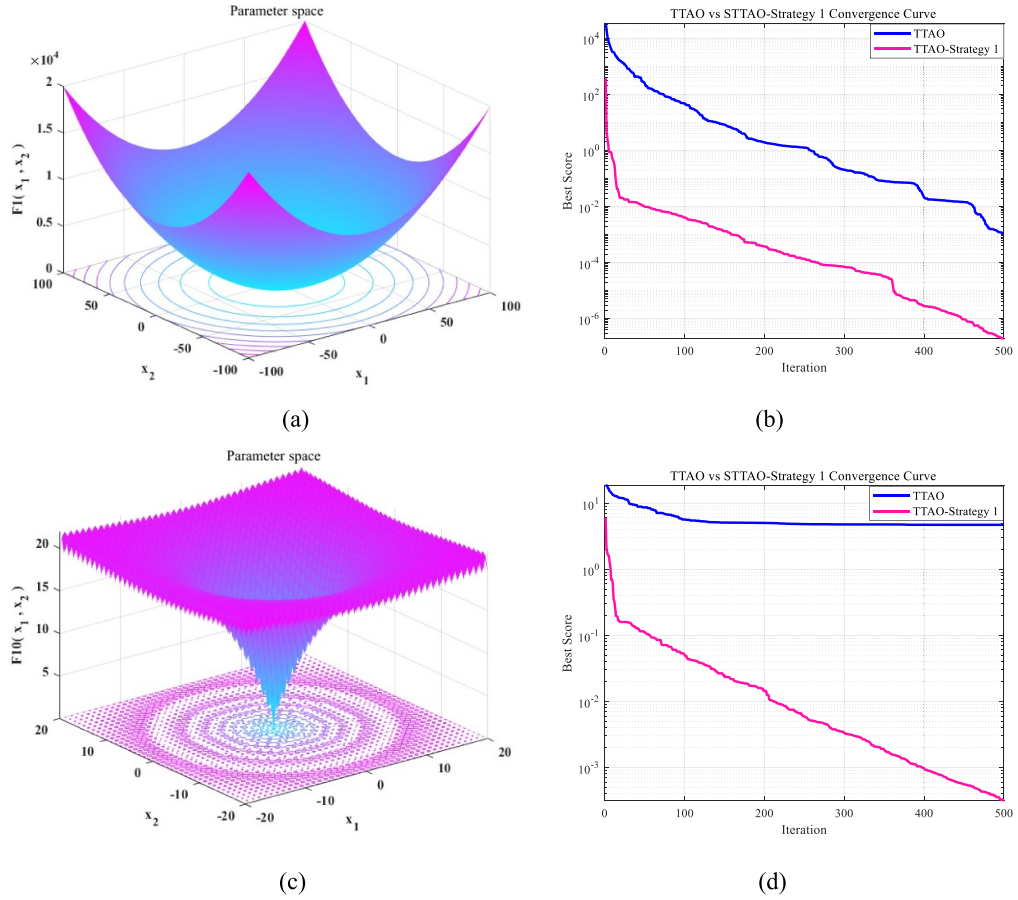


Figure 2. Performance verification of improvement strategy 1.

of optimizing the unimodal test function F1 than the original TTAO (blue solid line). This suggests that the proposed strategy strengthens early exploration, allowing the algorithm to approach the global optimum more efficiently. Additionally, once convergence is reached, the improved algorithm attains higher optimization accuracy than the original TTAO, confirming its superior performance in unimodal problems. For the multimodal test function F10 (figure 2(d)), TTAO-Strategy 1 consistently outperforms the original algorithm, particularly in later iterations, where it maintains a strong convergence trend and reduces the likelihood of premature local optima entrapment. These findings demonstrate that the adaptive collaborative group-weight dynamic balancing strategy enhances information exchange among individuals, improving search efficiency while maintaining global exploration capability.

2.1.3. Performance evaluation of improvement strategy 2.

To evaluate the impact of Improvement Strategy 2, which introduces an adaptive global exploration mechanism based on the CICD, on the optimization performance of the TTAO algorithm, comparative experiments were conducted on the unimodal test function F2 and the multimodal test function F9. These tests aimed to assess the effectiveness of the proposed strategy across different types of optimization problems. The

experiments were performed using both the original TTAO and the improved ITTAO, with experimental settings identical to those used in the validation of Improvement Strategy 1. The results are presented in figure 3.

As illustrated in figures 3(b) and (c), for the unimodal test function F2, the original TTAO algorithm exhibits a dispersed distribution of search points (green dots) across the solution space. This results in a significant amount of ineffective exploration, leading to suboptimal search efficiency, with some individuals failing to converge rapidly to the optimal solution (red pentagram). In contrast, the improved TTAO algorithm integrates the CICD for adaptive global exploration, enabling a more targeted early-stage search. This adjustment minimizes ineffective exploration, enhances search efficiency, and accelerates convergence toward the optimal solution. Figures 3(f) and (g) further demonstrate that for the multimodal test function F9, the original TTAO algorithm struggles with premature convergence, as indicated by search trajectories lingering near local optima. The enhanced TTAO algorithm, however, exhibits a more effective search capability, allowing for rapid early-stage convergence to the global optimum. These results confirm that the CICD-based adaptive global exploration mechanism significantly reduces ineffective searches and improves search directionality. By leveraging the heavy-tailed

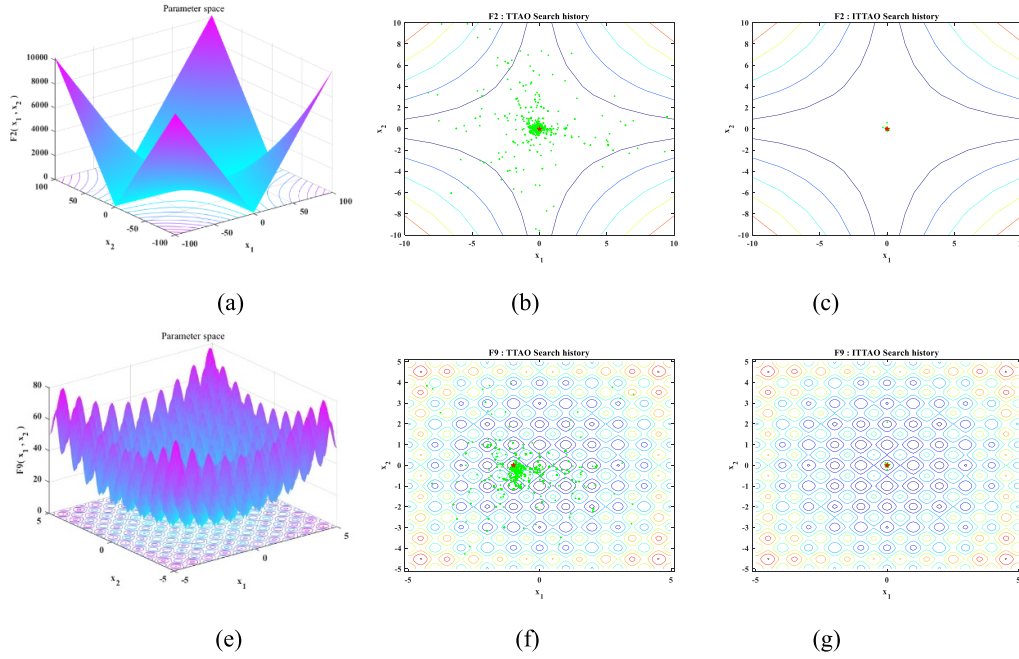


Figure 3. Performance verification of improvement strategy 2.

property of the Cauchy distribution, the mechanism enables broader early-stage exploration, preventing entrapment in local optima. Additionally, the inverse cumulative distribution function dynamically adjusts the search range, ensuring that the algorithm adapts effectively across different optimization phases. This enhancement not only significantly accelerates convergence for unimodal problems but also improves the algorithm's global search capability in multimodal optimization tasks, effectively mitigating the risk of premature convergence.

2.1.4. Performance evaluation of improvement strategy 3.

To assess the impact of the TNFPO on the optimization performance of the TTAO algorithm, this study conducts comparative experiments on the unimodal test function F3 and the multimodal test function F11. The objective is to evaluate the effectiveness of TNFPO in enhancing both convergence speed and global search capability.

As shown in figure 4(b), the original TTAO algorithm exhibits a relatively slow decline in the early convergence phase and later stagnates into a slow convergence state, indicating a risk of premature convergence during the local search phase. In contrast, the TTAO algorithm incorporating the TNFPO mechanism significantly reduces the objective function value in the early optimization stage and approaches the global optimum within a smaller number of iterations, demonstrating higher convergence efficiency. This suggests that TNFPO effectively enhances local search capability through nonlinear perturbations, thereby accelerating the convergence process for unimodal functions. As illustrated in figure 4(c), the original TTAO algorithm experiences a slow decline in the optimization process and stagnates in the mid-to-late

stages, implying that certain individuals may be trapped in local optima. In contrast, the improved ITTAO algorithm significantly enhances convergence speed and strengthens the algorithm's ability to escape local optima. This indicates that TNFPO, by simulating the random jump behavior of Lévy flight, introduces both large- and small-scale non-uniform perturbations during the search process. Consequently, it enhances the algorithm's global search capability and reduces the probability of premature convergence in multimodal optimization problems.

2.2. Dual-branch multi-scale attention feature extraction model

The proposed DMCSA-DRF model architecture, illustrated in figure 5, includes two primary components: the DMCSA module and the DRF module. The model uses Gram angle difference/sum field images as input for both branches. Each branch utilizes a multi-scale attention structure with residual connections for feature extraction. A dense residual structure is applied at the model's backend to efficiently integrate and repeatedly utilize the features from both branches, ensuring precise fault recognition across various fault types.

2.2.1. Multiscale channel and spatial attention feature extractor (MCSA).

Figure 6 illustrates the multi-scale feature extraction structure. The input image is first processed by the multi-scale convolution layer (MS-Conv) for feature extraction at multiple scales. Subsequently, parallel structures are employed to extract features using the Channel attention

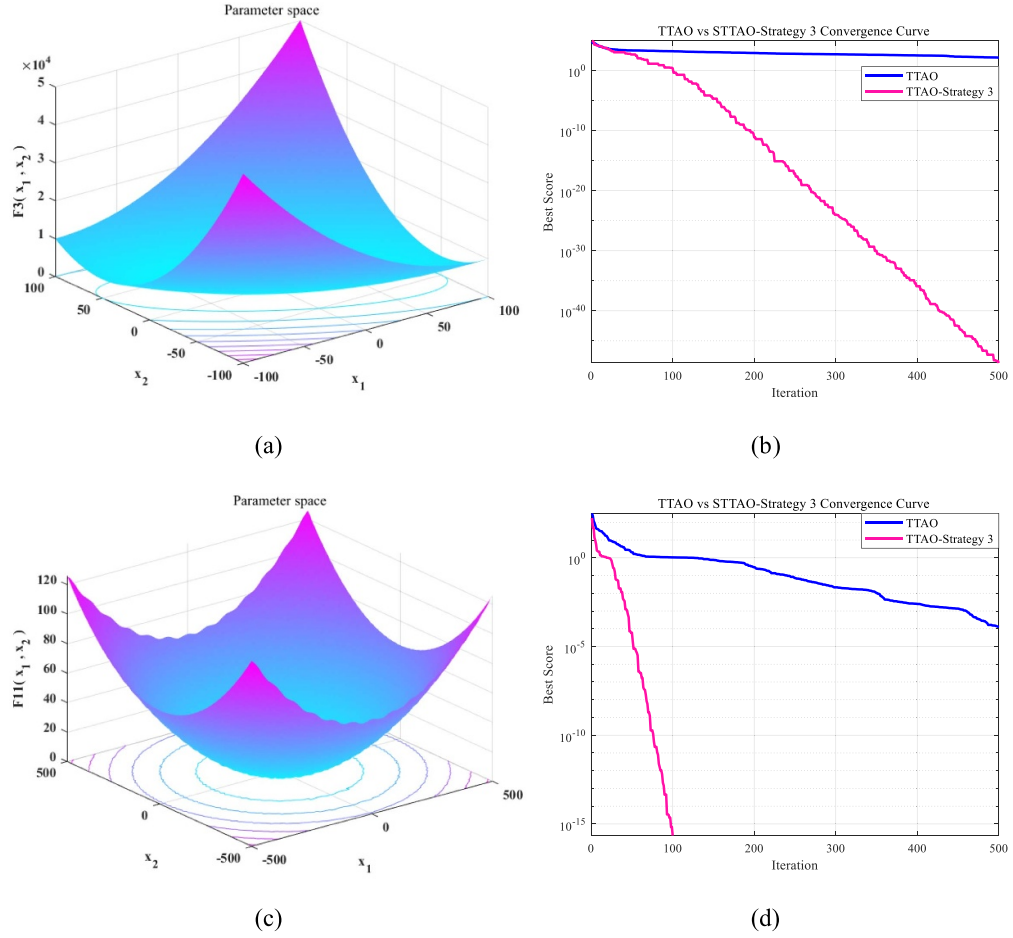


Figure 4. Performance verification of improvement strategy 3.

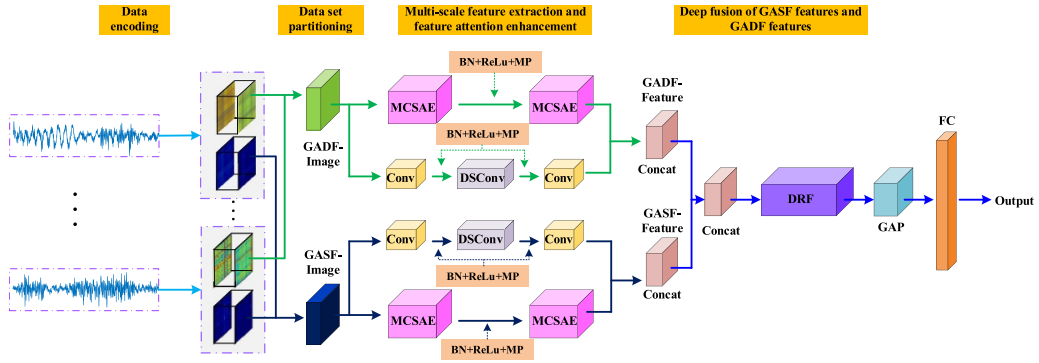


Figure 5. Structure of the DMCSA-DRF model.

enhancement mechanism (CAE) and SAE attention weight matrices, denoted as α and β , respectively.

The output of the MCSA network is described as follows:

$$F_{DS} = DSConv(F_{MS}) \quad (11)$$

$$F_{MCSA} = Concat[F_{MS} * \alpha, F_{DS}, F_{MS} * \beta]. \quad (12)$$

2.2.2. Multiscale feature extraction model. This paper utilizes convolution kernels of different sizes for multi-scale feature extraction. To reduce the number of model parameters, a depthwise separable convolution structure is employed as a substitute for the traditional convolution structure. The multi-scale feature extraction framework is shown in figure 7.

The multiscale feature extraction unit comprises three convolutional kernels of different sizes ($K1$, $K2$, and $K3$), each extracting features of different scales

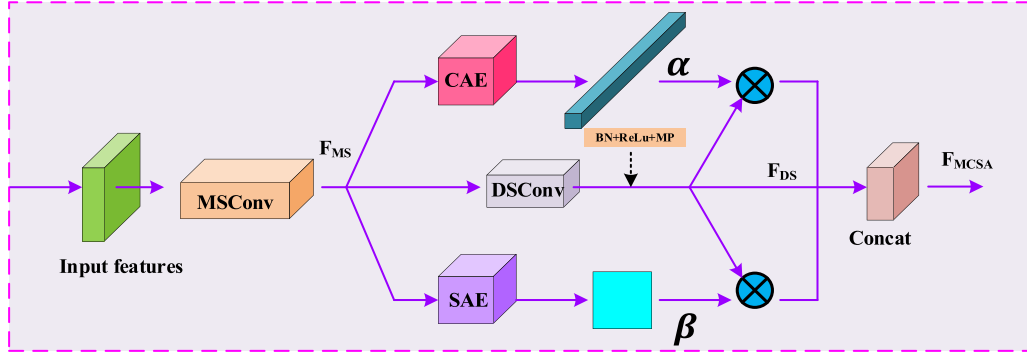


Figure 6. Multi-scale channel with spatial attention feature extractor.

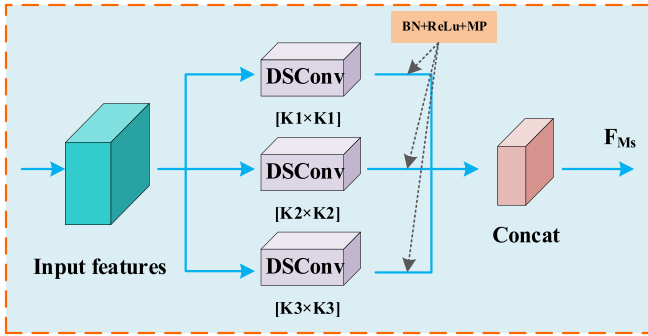


Figure 7. Multi-scale feature extraction model.

$(F_{K1}^{C_{K1} \times H \times W}, F_{K2}^{C_{K2} \times H \times W}, F_{K3}^{C_{K3} \times H \times W})$ from the input data. This enables the model to capture detailed information at various levels of the data. The different features extracted at individual scales are concatenated across channels to form a multiscale feature set F_{MS} , calculated as follows:

$$F_{MS} = [F_{K1}^{C_{K1} \times H \times W}, F_{K2}^{C_{K2} \times H \times W}, F_{K3}^{C_{K3} \times H \times W}] = F^{(C_{K1} + C_{K2} + C_{K3}) \times H \times W}. \quad (13)$$

2.2.3. Channel attention enhancement mechanism (CAE).

In CNN, different channels may represent distinct features, with varying levels of contribution to the task being processed. To address this issue, this paper proposes an CAE mechanism. CAE combines the strengths of the channel attention mechanisms in CBAM and CSAR by replacing fully connected layers with 1D convolutions, thereby avoiding the detrimental effects of FC layers. Furthermore, while CSAR's CA mechanism only uses global average pooling (GAP) and fails to consider the global information of the input features, the improved CAE preserves the global max pooling (GMP) operation from CBAM. The CAE structure is depicted in figure 8. CAE strengthens the channel attention for features that significantly contribute to the model, while diminishing attention for less significant features, allowing the model to dynamically learn critical feature representations.

Global maximum/mean pooling (GAP/GMP) operation is used in the CAE, the GMP and GAP computations are

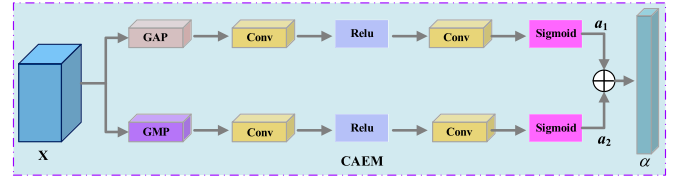


Figure 8. Improved channel attention enhancement mechanism.

described as follows:

$$M_1 = V_{GAP}(X) = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n X_c(i, j) \quad (14)$$

$$M_2 = U_{GMP}(X) = \text{Max}[X_c(i, j)] \quad (15)$$

where $X_c(i, j)$ is the eigenvalue of the input feature at (i, j) for the c th channel, δ is the ReLU activation function, and σ is the sigmoid activation function.

The feature information extracted from GMP and GAP was further filtered by two 1×1 convolution and activation functions to calculate the local and global attention weights for different feature channels, respectively, which were calculated as follows:

$$a_1 = \sigma(W_{CAE-conv2}^2 * \delta(W_{CAE-conv1}^1 * M_1 + b_{CAE-conv1}^1) + b_{CAE-conv2}^2) \quad (16)$$

$$a_2 = \sigma(W_{CAE-conv2}^2 * \delta(W_{CAE-conv1}^1 * M_2 + b_{CAE-conv1}^1) + b_{CAE-conv2}^2) \quad (17)$$

$$\alpha = a_1 \oplus a_2 \quad (18)$$

where $W_{CAE-conv1}^1 \in \mathbb{R}^{\frac{C}{r} \times C \times 1 \times 1}$ represents the weight matrix corresponding to the first convolutional layer after GMP or GAP, and $b_{CAE-conv1}^1 \in \mathbb{R}^{\frac{C}{r}}$ denotes the bias of the first convolutional layer. $W_{CAE-conv2}^2 \in \mathbb{R}^{C \times \frac{C}{r} \times 1 \times 1}$ represents the weight matrix of the second convolutional layer, $b_{CAE-conv2}^2 \in \mathbb{R}^C$ denotes the bias of the second convolutional layer, and r is the channel reduction factor.

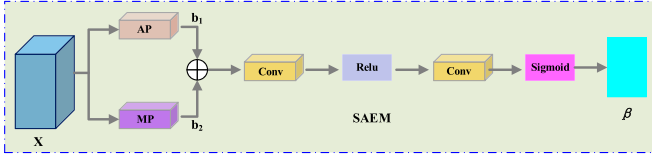


Figure 9. Improved spatial attention enhancement mechanism.

2.2.4. Spatial attention enhancement mechanism (SAE).

The spatial attention mechanism typically enhances or diminishes the representational ability of different locations in the feature map by learning the weights of each feature position, thereby improving the model's focus on features at different positions. This paper proposes an improved SAE mechanism, as shown in figure 9.

The matrix extracted by maximum pooling in SAE is denoted as $d_1 = [d_1^1, d_1^2, \dots, d_1^C]$, the dimension of d_1 is $c \times m \times n$, and the matrix extracted by average pooling in SAE is denoted as $d_2 = [d_2^1, d_2^2, \dots, d_2^C]$, the dimension of d_2 is $c \times m \times n$, $d_1 \oplus d_2$ is described as:

$$d_1^i = \begin{bmatrix} w_{d1}^{11} & w_{d1}^{12} & \dots & w_{d1}^{1n} \\ w_{d1}^{21} & w_{d1}^{22} & \dots & w_{d1}^{2n} \\ \dots & \dots & \dots & \dots \\ w_{d1}^{m1} & w_{d1}^{m2} & \dots & w_{d1}^{mn} \end{bmatrix}_{m \times n} \quad (19)$$

$$d_2^i = \begin{bmatrix} w_{d2}^{11} & w_{d2}^{12} & \dots & w_{d2}^{1n} \\ w_{d2}^{21} & w_{d2}^{22} & \dots & w_{d2}^{2n} \\ \dots & \dots & \dots & \dots \\ w_{d2}^{m1} & w_{d2}^{m2} & \dots & w_{d2}^{mn} \end{bmatrix}_{m \times n} \quad (20)$$

$$d_1^i \oplus d_2^i = \begin{bmatrix} w_{d1}^{11} + w_{d2}^{11} & \dots & \dots & w_{d1}^{1n} + w_{d2}^{1n} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ w_{d1}^{m1} + w_{d2}^{m1} & \dots & \dots & w_{d1}^{mn} + w_{d2}^{mn} \end{bmatrix}_{m \times n} \quad (21)$$

where d_1^i and d_2^i are the weight matrices of the i th channel extracted by the maximum and average pooling operations, respectively, and w_{d1}^{mn} and w_{d2}^{mn} are the weights in the weight matrices extracted by the maximum and average pooling.

The SAE weight matrix of the features is formulated as:

$$\beta = \sigma(W_{\text{SAE-conv2}}^2 * \delta(W_{\text{SAE-conv1}}^1 * (d_1 \oplus d_2) + b_{\text{SAE-conv1}}^1) + b_{\text{SAE-conv2}}^2) \quad (22)$$

where $W_{\text{SAE-conv1}}^1 \in \mathbb{R}^{\varepsilon C \times C \times 1 \times 1}$ is the weight matrix of the first convolutional layer, $b_{\text{SAE-conv1}}^1 \in \mathbb{R}^{\varepsilon C}$ is the bias of the first convolutional layer, ε is the reduction ratio, and the spatial attention weight matrix $\beta^{1 \times m \times n}$ is obtained by the second convolution.

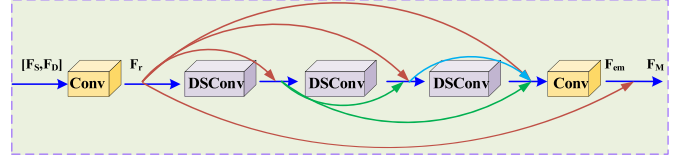


Figure 10. Dense residual feature integration module.

2.2.5. Feature residual dense fusion (DRF) module.

Figure 10 illustrates the DRF. Unlike traditional residual structures, this paper replaces the standard convolution in the residual connections with depthwise separable convolutions, further reducing the number of parameters while maintaining model performance.

In the DRF network, each layer's output serves as input to all subsequent layers, utilizing a dense residual connection mechanism. This not only alleviates the issue of vanishing gradients but also enables the reuse of feature maps, helping to prevent performance degradation as the network depth increases. The DRF network contains three convolutional layers, with the input feature F_r represented as:

$$F_r = \delta(\text{Conv}[F_S, F_D]) \quad (23)$$

where $[F_S, F_D]$ is the GAF feature splice map extracted by MCSA, and δ is the Relu activation function. The output of the m th convolutional layer in the DRF is:

$$F_{d,m} = \delta(w_{d,m}[F_r, F_{d,0}, \dots, F_{d,m-1}]) \quad (24)$$

where $w_{d,m}$ denotes the weights of the m -th convolutional layer in the DRF, and $[F_r, F_{d,0}, \dots, F_{d,m-1}]$ denotes the stitching of the residually connected feature maps. The calculation of output F_{em} is as follows:

$$F_{em} = \text{Conv}[F_r, F_{d,0}, F_{d,1}, F_{d,2}]. \quad (25)$$

The final output of the DRF is described as: $F_M = F_{em} \oplus F_r$.

2.3. The DMCSA-DRF unsupervised domain adaptation fault diagnosis framework

To diagnose faults across different operating conditions of the gearbox, this study adopts an unsupervised domain adaptation fault diagnosis framework based on DMCSA-DRF, as shown in figure 11. The framework utilizes DMCSA-DRF as the feature extractor and calculates the distribution difference of the features extracted from the source and target domains using the local maximum mean difference (LMMD) method, resulting in the domain loss (L_{LMMD}). Additionally, the classifier at the backend of the model computes the classification loss (L_c), and by combining L_{LMMD} and L_c , the total loss ($Loss$) for the unsupervised fault diagnosis framework is obtained. Through backpropagation, the model parameters are iteratively updated to enable accurate fault state identification.

The optimization objective of the DMCSA-DRF model includes the classifier loss L_c corresponding to the source

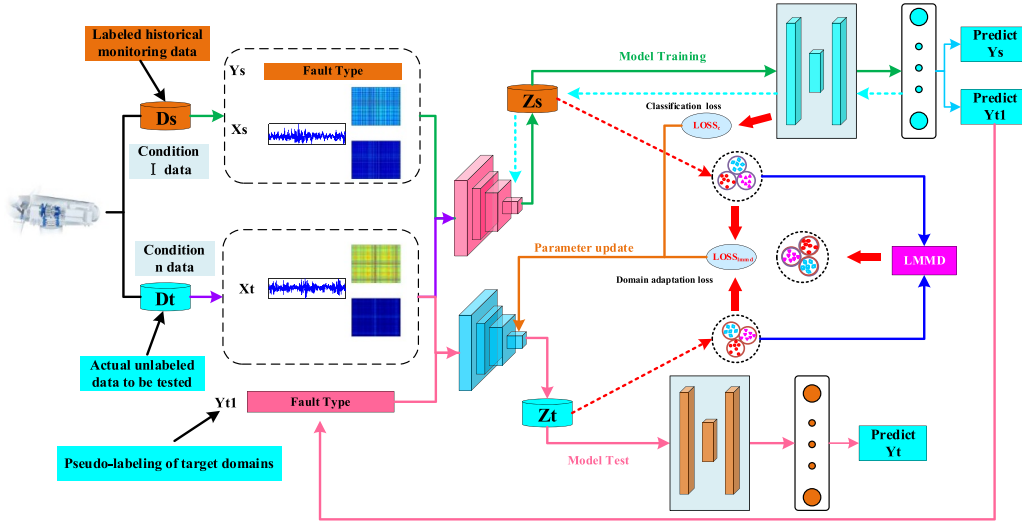


Figure 11. Unsupervised domain adaptation diagnostic framework.

domain data, as well as the domain loss L_{LMMD} between the source and target domains. The first optimization objective L_c is obtained by measuring the cross-entropy loss function of the error between the predicted labels Y'_s and the actual labels Y_s , calculated as follows:

$$L_c = -\frac{1}{n} \sum_{i=1}^n \sum_{c=1}^C y_{i,c}^s \lg f_c(z^s, i) \quad (26)$$

where n is the total number of samples in the source domain; C is the number of sample categories; z^s is the activation output generated by the fully connected layer; $f(z^s) = \text{softmax}(z^s)$; $f_c(z^s, i)$ is the c th element of output $f(z^s, i)$.

The second optimization objective is the domain loss L_{LMMD} . Given the source domain D_s and the target domain D_t , the samples from both domains follow different probability distributions p and q . By applying a kernel function, the data from both domains are mapped to a reproducing kernel Hilbert space, and the LMMD distance between the two domains is expressed as:

$$d_H(p, q) = \frac{1}{C} \sum_{c=1}^C \left\| \sum_{i=1}^n w_i^{sc} \phi(x_i^{(s)}) - \sum_{j=1}^m w_j^{tc} \phi(x_j^{(t)}) \right\|_H^2 \quad (27)$$

where C is the number of categories of fault samples; w_i^{sc} and w_j^{tc} are the weights of samples $x_i^{(s)}$ and $x_j^{(t)}$ belonging to a certain category of faults, respectively. The weight w_i^c is calculated as:

$$w_i^c = \frac{y_{ic}}{\sum_{(x_j, y_j) \in D} y_{jc}} \quad (28)$$

where y_{ic} is the c th element of the label vector Y_i . The source domain samples use their true labels $y_i^{(s)}$ to calculate the weight w_i^{sc} of each sample belonging to a particular class

of faults. The target domain samples use their pseudo-labels $\hat{y}_j = f(x_j)$ to compute the weight of each sample w_j^{tc} .

For a given source domain D_s and target domain D_t , the features extracted by the model are $\{Z_i^{sl}\}_{i=1}^n$ and $\{Z_j^{tl}\}_{j=1}^m$, respectively, and the LMMD calculation between the two domains is denoted as:

$$d_H(p, q) = \frac{1}{C} \sum_{c=1}^C \left[\sum_{i=1}^n \sum_{j=1}^m w_i^{sc} w_j^{tc} k(z_i^{sl}, z_j^{sl}) + \sum_{i=1}^m \sum_{j=1}^m w_i^{tc} w_j^{tc} k(z_i^{tl}, z_j^{tl}) - 2 \sum_{i=1}^n \sum_{j=1}^m w_i^{sc} w_j^{tc} k(z_i^{sl}, z_j^{tl}) \right] \quad (29)$$

where z^l is the activation output of the l -th fully connected layer ($l \in L = \{1, 2, \dots, |L|\}$).

The overall optimization objective of the model is:

$$\text{Loss} = L_c + \xi L_{LMMD} \quad (30)$$

where ξ is a hyperparameter.

The pseudocode for the DMCSA-DRF-based unsupervised domain adaptation algorithm is shown below:

2.4. System structure and diagnostic steps

The structure of the ITTAO-DMCSA-DRF model adopted in this paper is shown in figure 12.

The model training steps are described below:

Step 1: Sample interception and coding. All source and target domain data are intercepted, and the intercepted one-dimensional vibration signals are encoded into two-dimensional GADF and GASF image data using GAF.

Step 2: The model is pre-trained using labeled sample data $\{X_s, Y_s\}$ from the source domain. Then, the unlabeled target

Algorithm 2. Unsupervised domain adaptation fault diagnosis algorithm based on DMCSA-DRF pseudo-code.

Input: Labeled source domain data $D_s = \{x_i^s, y_i^s\}_{i=1}^N$, unlabeled target domain data $D_t = \{x_i^t\}_{i=1}^M$.

Output: Predicted labels for target domain samples $\hat{y}_t$.

1. **Initialize** model parameters

2. **while** not convergent **do**

(1) Randomly sample data $\{x_i^s, y_i^s\}_{i=1}^m$ from source domain D_s and data $\{x_i^t\}_{i=1}^m$ from target domain D_t .

(2) Input the source and target domain samples into the common feature extractor F to obtain shared features $z_i^s = F(x_i^s)$ and $z_i^t = F(x_i^t)$.

(3) Use the source classifier C to predict labels for source domain features: $\hat{y}_i^s = C(z_i^s)$.

(4) Compute the classification loss L_c according to equation (26).

(5) Calculate the domain adaptation loss L_{LMD} using the source domain features z^s and target domain features z^t according to equation (29).

(6) Compute the total loss L of the model according to equation (30).

(7) Update the model parameters through backpropagation.

3. **end while**

4. Input the target domain samples into the model to generate predicted labels: $\hat{y}_t = C(F(x^t))$.

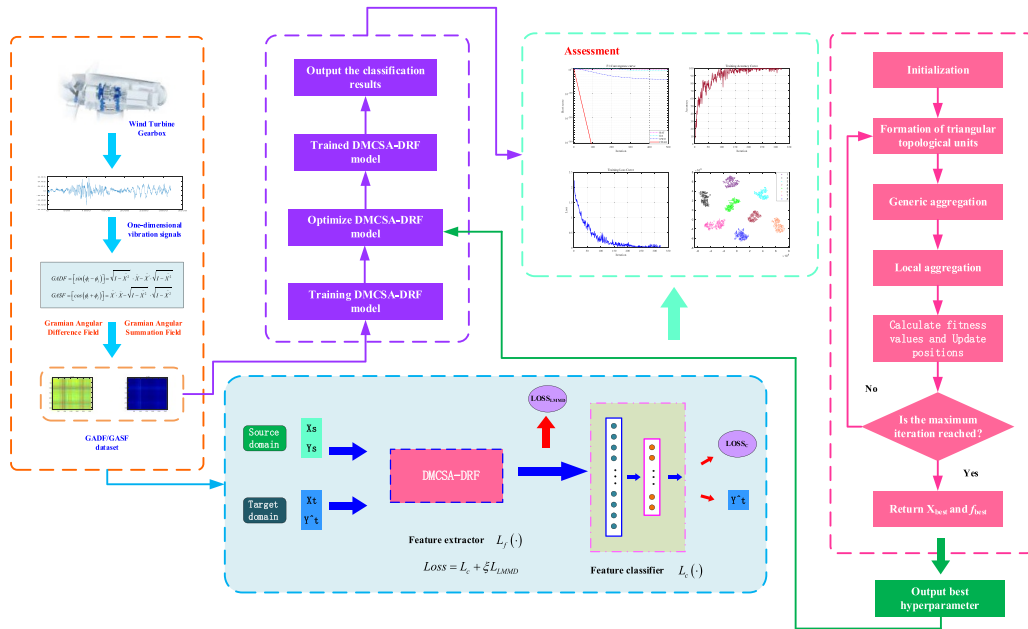


Figure 12. Overall system structure.

domain data $\{X_t\}$ is input into the pre-trained model to obtain the corresponding pseudo-labels $Y^{\wedge}_t$.

Step 3: Training the model. The source domain with labeled samples and the target domain with pseudo-labeled samples are input into the model using joint distribution alignment for feature distribution alignment, the total model $Loss$ is computed by classification loss and domain loss, and the gradient is computed by backpropagation and iterated according to the parameters optimized by ITTAO.

Step 4: Model Performance Evaluation. All the data in the target domain are used as inputs to the pre-trained model, and the performance of the model is evaluated by calculating the model's Accuracy, Precision, MacroRecall, and MacroF1-score based on the prediction results.

3. Experiments and analyzes

3.1. Data sources

3.1.1. Case study 1 dataset: PHM2009. The data was obtained using the PHM2009 gearbox data [55], the gearbox structure is shown in figure 13(a), with two accelerometers deployed near the input and output shafts on both sides of the gearbox. This gearbox dataset contains eight types of faults and the dataset description is shown in table 1. The data were collected at 30, 35, 40, 45, and 50 Hz with a sampling frequency of 66.7 kHz at high and low loads, respectively. In this paper, experimental validation is carried out using input data at high loads, with five speeds corresponding to five operating conditions, numbered A to E, as shown in table 2.

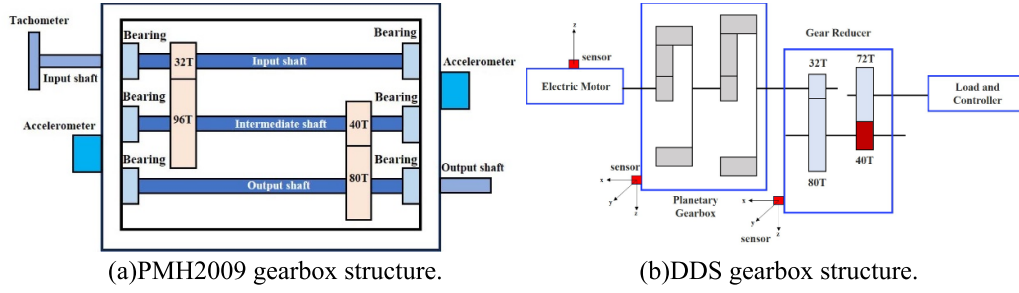


Figure 13. Gearbox structure.

Table 1. Description of fault categories in PHM2009 gearbox dataset.

Label	Gear				Bearing			Shaft	
	32 T	48 T	80 T	96 T	IS:IS	ID:IS	OS:IS	IS	OS
1	H	H	H	H	H	H	H	H	H
2	C	H	E	H	H	H	H	H	H
3	H	H	E	H	H	H	H	H	H
4	H	H	E	Br	B	H	H	H	H
5	C	H	E	Br	In	B	O	H	H
6	H	H	H	Br	In	B	O	Im	H
7	H	H	H	H	In	H	H	H	H
8	H	H	H	H	H	B	O	Im	KS

IS: Input shaft; ID: Intermediate shaft OS: Output shaft.

H= Health; E: Eccentric; Br: Broke; B: Ball; In: Inner; Im: Imbalance.

KS: Keyway shear.

Table 2. PHM2009 gearbox dataset working condition markers.

Condition (Hz)	30	35	40	45	50
Code	A	B	C	D	E

Table 3. Description of DDS gearbox data set.

	Type	Description
Gearboxes	Health	Gears in a healthy state
	Chipped	The crack occurs in the gear feet
	Miss	Missing one of the feet in the gear
	Root	Crack occurs in the root of gear feet
	Surface	Wear occurs on the surface of the gear

3.1.2. Case study 1 dataset: Dynamics Simulator DDS. The gearbox dataset used in the experiment is from Southeast University's Drivetrain DDS [56]. The gearbox structure is shown in figure 13(b), which consists of components such as the motor, planetary gearbox, reduction gearbox, and load. Vibration sensors of model 608A11 are deployed at three different axial positions (x , y , and z) on the gearbox to collect vibration signals from gears exhibiting different fault modes. The gearbox features five distinct fault states, as listed in table 3. The dataset includes two operational conditions: Condition A with a rotational speed of 20 Hz (1200 rpm) and no load (0 V, 0 Nm), and Condition B with a rotational speed of 30 Hz (1800 rpm) under load (2 V, 7.32 Nm).

In the experiment, data from five categories of the DDS gearbox were resampled at equal intervals, resulting in a dataset containing 1,000 samples, each with a length of 1,024. The resampled data were then encoded using GAF and used as input to the model for performance validation.

In the experiment, data from five categories of the DDS gearbox were resampled at equal intervals, resulting in a dataset containing 1,000 samples, each with a length of 1,024. The resampled data were then encoded using GAF and used as input to the model for performance validation.

3.2. Indicators for model assessment

To evaluate the model's effectiveness, we employ commonly used evaluation metrics in fault diagnosis multiclass classification to assess its predictive performance. These metrics include accuracy, precision, recall, macro-recall, F1 score, and macro F1-score. Below are the formulas for calculating these evaluation metrics:

Accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}. \quad (31)$$

Precision:

$$\text{Precision} = \frac{TP}{TP + FN}. \quad (32)$$

Recall:

$$\text{Recall} = \frac{TP}{TP + FN}. \quad (33)$$

Macro-recall:

$$\text{Macro Recall} = \frac{\sum_i \text{Recall}_i}{N}. \quad (34)$$

F1-score:

$$F_1 - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (35)$$

Macro F1-score:

$$\text{Macro } F_1 - \text{score} = \frac{\sum_i F_1 - \text{score}_i}{N}. \quad (36)$$

where TP stands for true positives, TN denotes true negatives, FP indicates false positives, FN refers to false negatives, and N represents the total number of sample classes.

3.3. Data pre-processing

The GAF [57] is a novel approach for representing time series data. GAF transforms one-dimensional signals into polar coordinates and performs inner product operations, ensuring angular differences between different modalities at the same time point, thus achieving maximum differentiation between different features. The definition of the GAF is as follows:

Given an input one-dimensional sequence of signals $X = \{x_1, x_2, \dots, x_n\}$, the signals are first normalized to the $[-1, 1]$ interval by equation (37) and the $[0, 1]$ interval by equation (38):

$$\tilde{x}_{-1}^i = \frac{(x_i - \max(X) + (x_i - \min(X)))}{\max(X) - \min(X)} \quad (37)$$

$$\tilde{x}_0^i = \frac{x_i - \min(X)}{\max(X) - \min(X)}. \quad (38)$$

Secondly, the normalized signals were mapped to the polar coordinate system using the following formulae, encoding the magnitude as the angular cosine and the time axis as the radius:

$$\phi = \arccos(\tilde{x}_i), -1 \leq \tilde{x}_i \leq 1, \tilde{x}_i \in \tilde{X} \quad (39)$$

$$r = \frac{t_i}{N}, t_i \in N$$

where t_i is a time point and N is a constant factor used to normalize the scale span of the polar coordinate system.

The formulas for computing the Gramian angular difference/summation field are presented below:

$$\text{GADF} = [\sin(\phi_i - \phi_j)] = \sqrt{I - \tilde{X}^2} \cdot \tilde{X} - \tilde{X}' \cdot \sqrt{I - \tilde{X}^2} \quad (40)$$

$$\text{GASF} = [\cos(\phi_i + \phi_j)] = \tilde{X}' \cdot \tilde{X} - \sqrt{I - \tilde{X}^2} \cdot \sqrt{I - \tilde{X}^2} \quad (41)$$

where I is the unit row vector. The 2D GADF and GASF transformed the image of a 1D sequence can be obtained by mapping the values of GADF and GASF to a 2D color image. The images of the PHM2009 dataset after coding for 8 different states are shown in figure 14.

3.4. Experimental validation

3.4.1. ITTAO algorithm performance validation. The performance of the model can be affected by the settings of certain hyperparameters during the training process. Furthermore, manually setting these hyperparameters has inherent limitations. Consequently, this paper utilizes the ITTAO algorithm for optimal hyperparameter configuration. The hyperparameters in the model include the InitialLearnRate, Mini-Batch Size,

and MaxEpochs. In the process of feature attention extraction, the ITTAO algorithm is employed to adaptively optimize the channel reduction factor parameter r and restoration factor parameter λ in the CAE and SAE modules. Additionally, optimization is conducted for the domain loss proportion factor ξ corresponding to the UDA framework, aiming to enhance the model's classification accuracy by dynamically adjusting the proportion of domain loss. Figure 15 presents the test results of the ITTAO algorithm alongside the bat algorithm (BAT), the dragonfly algorithm (DA), and the grey wolf algorithm (GWO) on some of the standard test functions. To ensure the validity of the experiments, each algorithm's population size N is set to 100, the number of iterations T is set to 500, and the dimensionality is set to 30. Each algorithm is tested 10 times.

Figure 15 illustrates the iterative processes of the ITTAO, BAT, DA, and GWO algorithms on several standard test functions. Compared to other algorithms, the ITTAO algorithm achieves faster convergence to the optimal solution on the F1–F4 unimodal test functions. On the multimodal test functions F8–F11, the ITTAO algorithm maintains its superior performance.

Table 4 presents the experimental results on the standard benchmark functions F1–F13, demonstrating that the proposed improved algorithm, ITTAO, outperforms the compared algorithms in most test cases. Moreover, compared to the original TTAO algorithm, ITTAO exhibits significant performance enhancements across nearly all test functions. The only exception is F8, where the improvement is marginal. These results indicate that the three proposed strategies effectively enhance both the exploration and exploitation capabilities of the algorithm.

Table 5 presents the optimal parameters required for the DMCSA-DRF model along with their respective value ranges.

To further validate the effectiveness of the ITTAO algorithm, the proposed model was evaluated under six different optimization strategies or algorithmic conditions. Specifically, 'Non' refers to a baseline model where no optimization technique was applied, and parameters were set manually based on empirical experience. The model being optimized is the proposed DMCSA-DRF. The other comparative algorithms include the GS algorithm, the improved Harris hawks optimization algorithm (IHOO) [53], the improved crayfish optimization algorithm (ICOA) [58], the improved whale optimization algorithm (IWOA) [59], and the proposed ITTAO algorithm. Each experiment was conducted five times, and the average performance was used as the final evaluation metric. The results are presented in table 6 and figure 16.

As shown in table 6, when no optimization algorithm was used (Non), the model had the shortest training time (7 min), but its prediction accuracy was only 96.34%, indicating the limitations of manually set parameters. In contrast, GS significantly improved the prediction accuracy to 99.16%. However, due to its exhaustive search method, the training time increased substantially to 117 min, resulting in a very high computational cost that makes it unsuitable for practical complex scenarios. Heuristic optimization algorithms (IHOO,

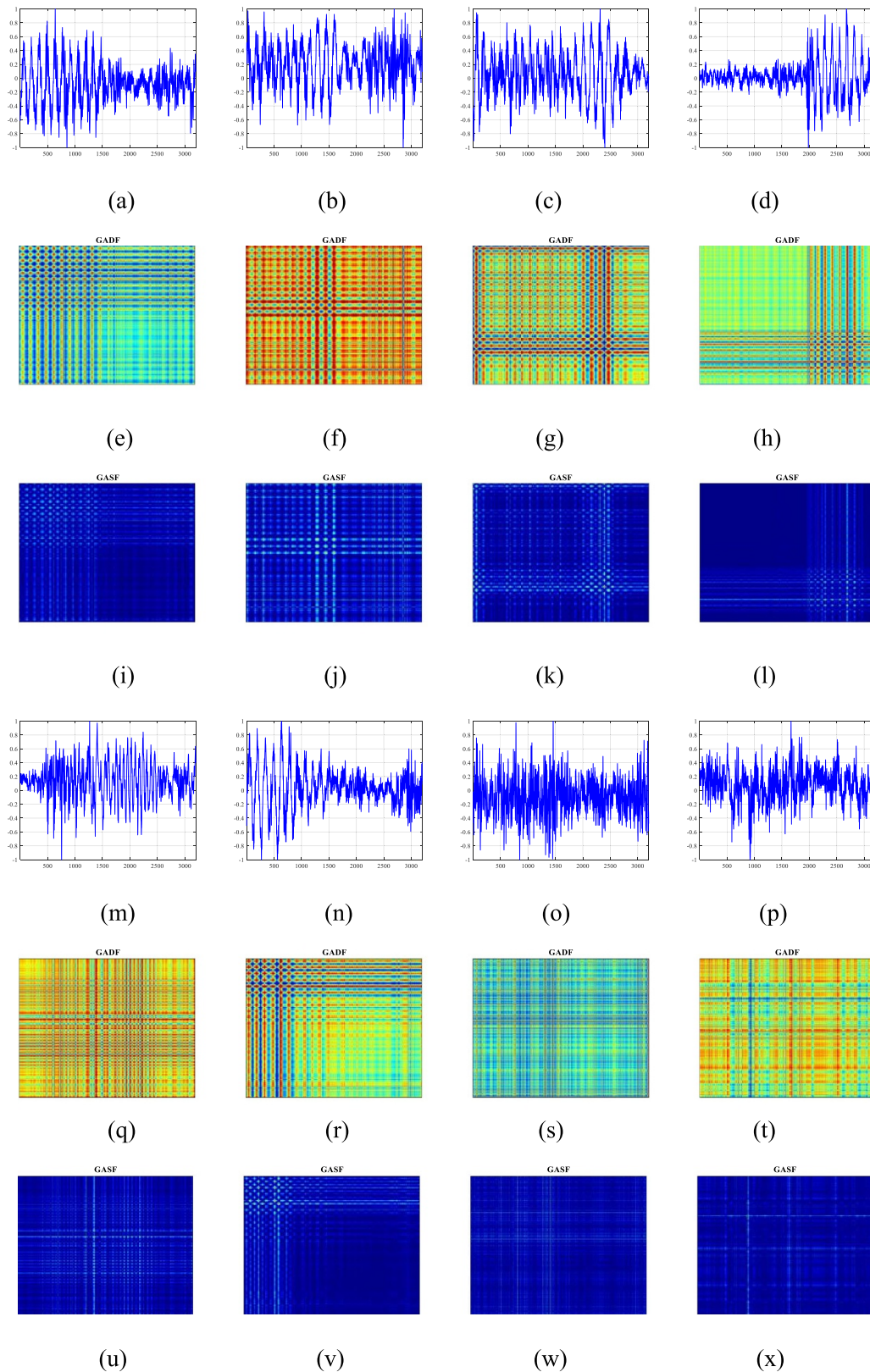


Figure 14. Vibration waveforms and GADF and GASF image for eight states at 35 Hz rpm for PHM2009 gearbox data: (a), (b), (c), (d), (m), a (n), (o), respectively (p) PHM2009 eight different data sets the vibration signals of the fault statuses; (e), (f), (g), (h), (q), (r), (s), (t) are respectively PHM2009 dataset GADF image features of 8 different fault statuses; (i), (j), (k), (l), (u), (v), (w), (x) are respectively PHM2009 dataset GASF image features of 8 different fault statuses.

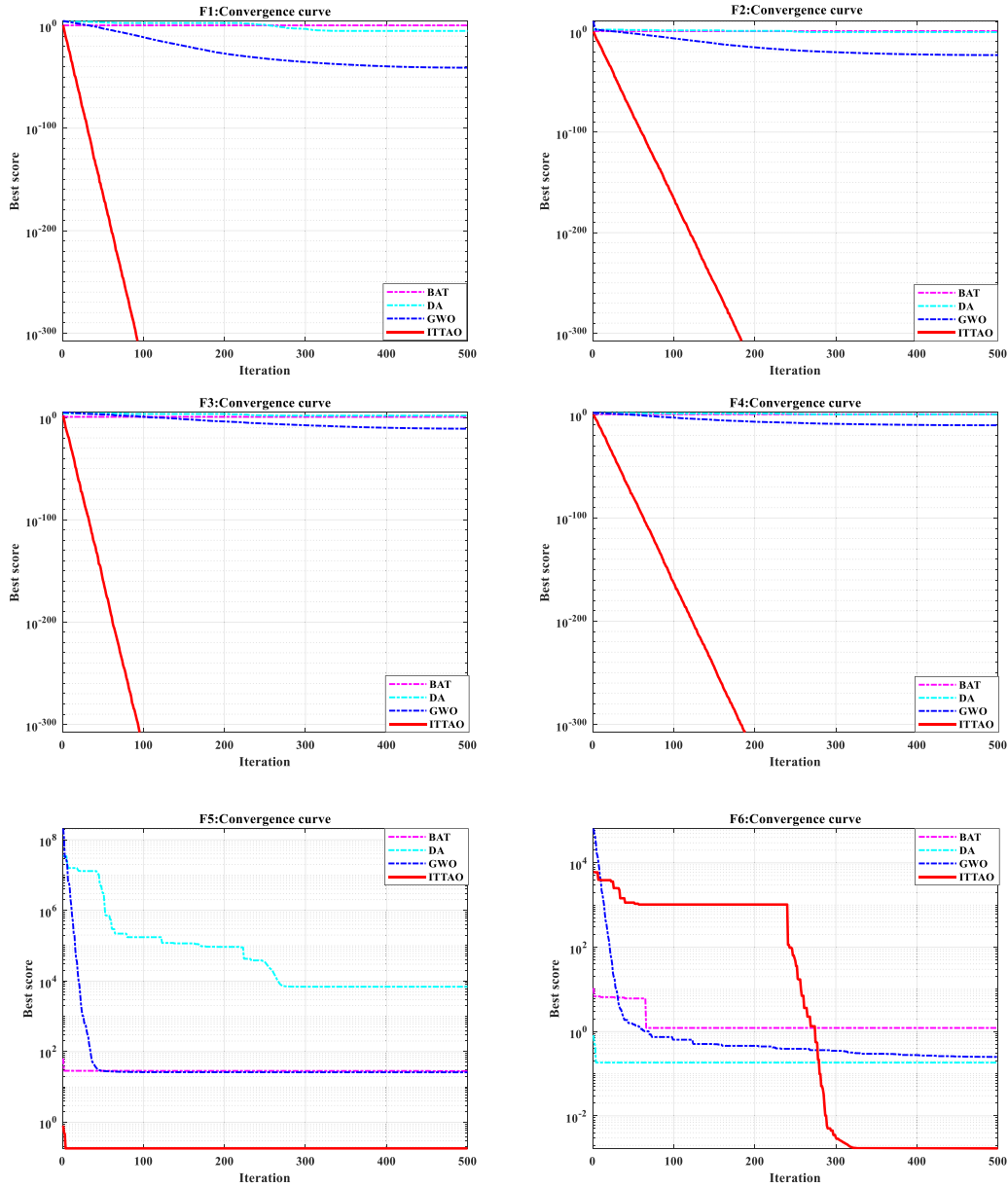


Figure 15. Test results of ITTAO algorithm and other optimization algorithms on some standard test functions.

ICOA, IWOA) achieved a better balance between training time and prediction accuracy compared to Non and GS. The prediction accuracies of IHHO, ICOA, and IWOA were 98.92%, 98.96%, and 98.65%, respectively, with training times controlled between 35 and 40 min. Compared to GS, heuristic optimization algorithms effectively reduced the training time while maintaining high prediction accuracy. Compared to Non, they significantly enhanced model performance, demonstrating that heuristic optimization algorithms can effectively improve the efficiency and quality of hyperparameter tuning. The ITTAO proposed in this paper further optimized the balance between training efficiency and model performance. It achieved the highest prediction accuracy of 99.86% with only 33 min of training time, significantly outperforming other optimization algorithms. Compared to Non, ITTAO increased

the training time by only approximately 26 min, yet resulted in a 3.52% accuracy improvement. Compared to GS, ITTAO reduced the training time by 72% while improving accuracy by 0.7%, reflecting its efficiency and optimization capability. Moreover, as seen in figure 16, although the optimization algorithms increased the computational overhead during the training phase, the differences in test times among different algorithms were minimal, all remaining between 0.56 s and 0.66 s. This suggests that the use of optimization algorithms mainly affects the training phase and does not impact the real-time performance of online deployment.

As indicated in table 6, the DMCSA-DRF model exhibits its optimal learning performance when the hyperparameters InitialLearnRate, MiniBatchSize, and MaxEpochs are configured at 10⁻³, 64, and 20 respectively. With the CAE unit's

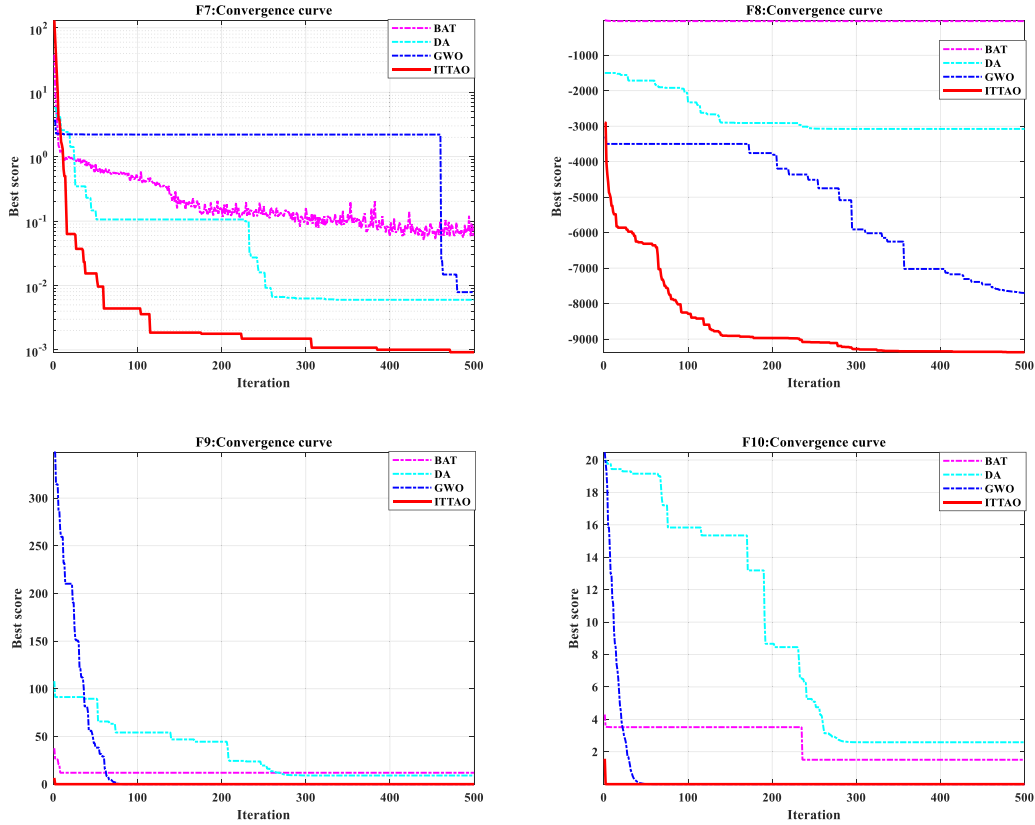


Figure 15. (Continued.)

channel reduction ratio r set to 8 and the SAE unit's channel restoration ratio ε set to 4, the model can effectively prioritize crucial feature information. Additionally, by setting the domain loss ratio factor ξ to 0.5, the model achieves a satisfactory balance between the two types of losses, ensuring accurate identification of various fault states. It is worth noting that when no optimization algorithm is used, manual tuning introduces an unquantifiable time cost, and the subjectivity of parameter setting can lead to unstable model performance. In contrast, ITTAO avoids the uncertainty associated with manual tuning by automatically searching for optimal parameters, resulting in more stable and efficient optimization. Moreover, ITTAO demonstrates its effectiveness in deep learning hyperparameter optimization by achieving faster convergence, lower computational costs, and superior optimization capabilities. This makes it a highly efficient optimization solution for engineering applications, such as gearbox fault diagnosis.

3.4.2. Model performance validation.

3.4.2.1. Performance verification of the data preprocessing.

To evaluate the effectiveness of GASF/GADF preprocessing in fault diagnosis tasks, this study designs a comparative experiment focusing on data preprocessing, aimed at investigating the impact of this preprocessing method on model performance. Since the original one-dimensional

vibration signals have certain limitations in expressing temporal features, the GASF/GADF method can convert time series into two-dimensional images with spatiotemporal dependencies, which helps improve the model's feature extraction capabilities. Therefore, we selected models with different architectures to compare their performance differences between unprocessed and preprocessed data, to validate the overall impact of the GASF/GADF transformation on accuracy, robustness, and computational efficiency. The experimental setup keeps the number of layers for 1D-CNN and 2D-CNN consistent, with only the input signals differing. The LSTM and GRU hidden layer units are set to 75 and 80, respectively, while the CNN in CNN-LSTM and CNN-GRU consist of 3×3 convolutional layers designed to process the 2D images. Each model experiment was conducted 5 times, with the average being taken as the final evaluation result. The experimental results are shown in table 7.

As shown in table 7, in terms of prediction accuracy, the models based on raw signals (1D-CNN, LSTM, GRU) achieved accuracies of 76.23%, 81.44%, and 83.42%, respectively. In contrast, models using GASF/GADF transformations (2D-CNN, CNN-LSTM, CNN-GRU) saw improvements in accuracy by 9.13%, 7.49%, and 7.17%, highlighting the advantages of preprocessing in feature representation. Although the training time increased from 7–9 s to 11–15 s, and the testing time also rose (from 0.23–0.27 s to 0.29–0.30 s), the increase in computational cost is still acceptable given the accuracy gains. Notably, the CNN-GRU model

Table 4. Test results of the improved algorithm and other algorithms on standard test functions.

Function		BAT	DA	GWO	TTAO	ITTAO
F1	Mean	0.3811	221.3359	3.61×10^{-41}	3.10×10^{-3}	0
	Std	0.1777	133.5912	3.949×10^{-41}	4.60×10^{-3}	0
F2	Mean	1.3692	5.7088	4.799×10^{-24}	3.98×10^{-2}	0
	Std	1.9974	4.8701	5.688×10^{-24}	3.87×10^{-2}	0
F3	Mean	5.2213	6.936×10^3	6.066×10^{-12}	126.94	0
	Std	11.5471	1.046×10^3	5.988×10^{-12}	47.92	0
F4	Mean	0.2987	13.2627	1.639×10^{-10}	4.768	0
	Std	0.3195	4.8746	4.825×10^{-11}	0.995	0
F5	Mean	44.5232	1.1206×10^4	25.8992	183.301	28.1295
	Std	34.4697	1.0161×10^4	0.7777	95.2583	0.1226
F6	Mean	4.3552	174.232	0.3496	2.1169	0.0032
	Std	3.8514	260.323	0.2265	0.4334	0.0036
F7	Mean	0.0162	0.108	7.2852×10^{-4}	0.0353	0.0067
	Std	0.0352	0.0732	1.7345×10^{-4}	0.0118	0.0041
F8	Mean	-116.6015	-5.911×10^{-3}	-6.557×10^3	-7.929×10^{-3}	-8.807×10^{-3}
	Std	0.4135	560.0519	489.0051	490.940	251.6433
F9	Mean	38.4139	95.438	1.4276	15.324	0
	Std	13.5351	35.7186	0.6835	3.4899	0
F10	Mean	0.3088	8.1203	2.815×10^{-14}	4.1682	4.441×10^{-16}
	Std	0.5069	5.4866	1.589×10^{-15}	1.1166	0
F11	Mean	0.0025	3.2396	0	0.0126	0
	Std	0.0050	3.5499	0	0.0084	0
F12	Mean	0.2601	9.7770	0.1772	0.9979	0.0181
	Std	0.04	13.2665	0.0403	0.6845	0.0097

Note: Bold values indicate the best performance results.

Table 5. Model parameter search results.

Parameter index	Hyperparameter name	Search range
h_1	InitialLearnRate	$[10^{-2}-10^{-6}]$
h_2	MiniBatchsize	[8,16,32, ...,521,1024]
h_3	MaxEpochs	[20-100]
h_4	Reduction ratio r in the CAE unit	[2-64]
h_5	Increase the ratio ε in the SAE unit	[2-64]
h_6	Domain loss Percentage Factor ξ	[0-1]

achieved superior training efficiency while maintaining high accuracy. Overall, GASF/GADF preprocessing effectively enhances the model's discriminative capability and improves fault diagnosis accuracy, while maintaining a good balance between training and inference efficiency.

3.4.2.2. Performance verification of the attention mechanism.

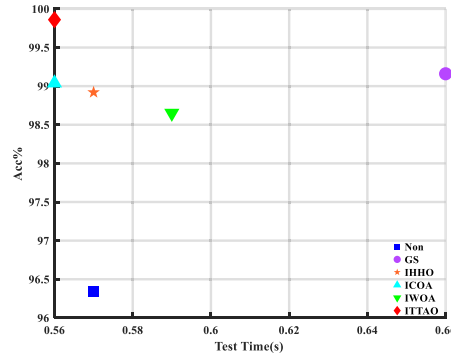
To validate the effectiveness of the proposed improved attention enhancement mechanism, ablation experiments were conducted comparing our method with other convolutional attention mechanisms, including SE [36], CSAR [39], CBAM [40], and GAM [60]. In these experiments, the same CNN architecture was used in conjunction with each attention mechanism for performance evaluation. The experimental results are shown in table 8.

As shown in table 8, the SE attention mechanism has the lowest number of parameters and computational complexity (6k and 0.091 G), with the shortest testing time (0.28 s). However, due to its consideration of only channel attention, the classification accuracy is limited to 88.33%. In comparison, the CBAM and GAM attention mechanisms calculate both channel and spatial attention, achieving classification accuracies of 91.06% and 92.22%, respectively. However, due to their multi-layer perceptron and large convolutional kernel structures, the parameters (105.4 k and 105.7 k) and computational complexity (FLOPs) of CBAM and GAM are

Table 6. Comparative test of different optimization methods.

Number	Optimization algorithm	Train time	Test time	ACC (%)	Parameter [h1–h6]
1	Non	7 min	0.57 s	96.34	[10 ⁻² , 32, 30, 4, 4, 0.4]
2	GS	117 min	0.66 s	99.16	[10 ⁻³ , 64, 30, 8, 4, 0.5]
3	IHHO	35 min	0.57 s	98.92	[10 ⁻² , 128, 20, 4, 4, 0.5]
4	ICOA	39 min	0.56 s	99.04	[10 ⁻³ , 32, 20, 4, 2, 0.4]
5	IWOA	40 min	0.59 s	98.65	[10 ⁻² , 64, 20, 4, 2, 0.5]
6	ITTAO	33 min	0.56 s	99.86	[10 ⁻³ , 64, 20, 8, 4, 0.5]

Note: The bold numbers represent the highest accuracy achieved among all methods.

**Figure 16.** Comparison of diagnostic performance and test time of different models.**Table 7.** Ablation study of GASF/GADF data preprocessing.

Model	Input signal type		ACC%	Evaluation metrics	
	Raw signal	GASF- GADF		Training time (s)	Test time (s)
1DCNN	✓		76.23	7.12	0.23
LSTM	✓		81.44	9.07	0.27
GRU	✓		83.42	8.66	0.25
2DCNN		✓	83.19	15.54	0.30
CNN-LSTM		✓	87.54	13.65	0.29
CNN-GRU		✓	89.43	11.60	0.29

Table 8. Ablation experiment on the DDS dataset.

Model	Attention	Parameters	FLOPs	ACC%	Test time (s)
CNN	SE	6k	0.091 G	88.33	0.28
	CSAR	32.1k	1.794 G	91.15	0.59
	CBAM	105.4k	20.447 G	91.06	0.66
	GAM	105.7k	36.558 G	92.22	0.85
	CAE-SAE	19.9k	0.597 G	94.37	0.52

significantly higher, at 20.447 G and 36.558 G, respectively. Consequently, the testing times (0.66 s and 0.85 s) are also higher than the other models, leading to greater computational resource demands. Although the parameter count (32.1 k) and FLOPs (1.794 G) of CSAR are lower than those of CBAM and GAM, its accuracy (91.15%) and testing time (0.59 s) are still lower than those of the CAE-SAE. Overall, the CAE-SAE attention mechanism designed in this study is comparable to SE in terms of parameter count (19.9 k) and computational complexity (FLOPs: 0.597 G), yet it achieves the highest classification accuracy (94.37%) with a testing time of just 0.52 s.

This demonstrates that the proposed method strikes a balance between high accuracy and efficient diagnostic performance.

3.4.2.3. Performance verification of the model structure.

To validate the effectiveness of the model designed in this study, ablation experiments were conducted on the PHM2009 dataset to assess the impact of different feature extraction modules on the overall network performance improvement. The experiments used a standard convolutional layer (CNN + BN + ReLU, CBR) as the baseline model, with input data processed in both single-input and multi-input modes. The results of the ablation experiments are presented in table 9.

Table 9 shows the results of the ablation experiments conducted on the PHM2009 dataset. For input GASF and GADF feature maps, the diagnostic accuracy of the single-branch CBR baseline model was 81.19% and 79.14%, respectively. After replacing the CBR baseline model with the MSCNN module, the diagnostic accuracy of the MSCNN model increased by 6.65%. The MSCNN model employs convolution kernels of multiple sizes, which not only enhance the

Table 9. Ablation experiment on the PHM2009 dataset.

	MSCNN	CAE	SAE	DRF			ACC%	PHM2009
					GASF	GADF		Learning parameters
CBR					√		81.19	4.8 M
CBR						√	79.14	4.8 M
MSCNN	√				√		86.97	0.9 M
MSCNN + CAE	√	√				√	92.45	1.1 M
MSCNN + SAE	√		√		√		93.66	1.1 M
MSCAM	√	√	√			√	95.76	1.3 M
MSCAM + DRF	√	√	√	√	√	√	97.55	1.7 M
DMSCAM + DRF	√	√	√	√	√	√	99.85	1.9 M

diagnostic performance but also significantly expand the number of learnable parameters. When the CAE and SAE modules were incorporated into the MSCNN model, the diagnostic accuracy further improved to 92.45% and 93.66%, respectively. Compared to the baseline CBR model, the diagnostic accuracy of the model improved by 7.14% and 7.77%, respectively, after incorporating the CAE and SAE attention mechanisms designed in this paper. The addition of the attention mechanisms allows for a more effective capture of key features representing faults in the data. However, it also slightly increases the number of learnable parameters. When the CAE and SAE channel and spatial attention modules were added to the MSCNN model, the diagnostic accuracy of the MSCAM model increased by 3.46% and 2.33%, respectively, compared to the single-attention models, MSCNN-CAE and MSCNN-SAE. By incorporating the DRF module into the MCSAM model, the diagnostic accuracy was enhanced by 1.83%. With the proposed dual-branch feature extraction model DMCSA-DRF, the diagnostic accuracy reached 99.85%. The experiments show that, in comparison to the single-branch MCSA-DRF model, the dual-branch DMCSA-DRF model improves diagnostic accuracy by 2.3%.

To validate the effectiveness of the proposed model, we replaced the attention mechanisms CAE/SAE, multiscale convolution MSCNN, and dense connection DRF in DMCSA-DRF, resulting in CNN-DRF, MSCNN-DRF, and MCSA-DRF, while keeping the overall layer count consistent. Additionally, we compared our models with two other methods combining different attention mechanisms with multiscale convolution, namely CSAM-MSCNN [61] and CBAM-MSCNN [62], the evaluation results are shown in figure 17.

As shown in figure 17, the prediction accuracy of the multi-scale feature extraction model MSCNN-DRF increased by 19.22% compared to the single-scale feature extraction model CNN-DRF. This indicates that the multi-scale structure can capture richer feature information from different receptive fields, aiding the model in uncovering deeper correlations between data. Compared to CNN-DRF and MSCNN-DRF, the DMCSA-DRF model shows an improvement in average classification accuracy by 27.6% and 7.03% across eight categories, respectively. This improvement suggests that the CAE and SAE mechanisms in the DMCSA-DRF model can

dynamically adjust attention distribution among various features, reducing redundant information and thereby enhancing the model's diagnostic performance. Compared to MCSA-DRF, DMCSA-DRF achieved a 1.97% increase in prediction accuracy, indicating that the dual-branch structure's input validates the model's ability to learn richer feature information. Furthermore, compared to the CBAM-MSCNN and CSAM-MSCNN, DMCSA-DRF exhibited improvements in prediction accuracy by 4.86% and 2.79%, respectively, suggesting that the parallel structure of channel and spatial attention in this paper holds an advantage over the concatenated structures of channel and spatial attention in the CBAM-MSCNN and CSAM-MSCNN.

Figure 18 displays the confusion matrices for various models on the transfer task from dataset B to A using the DDS gearbox data.

As illustrated in figure 18, the proposed MCSA-DRF feature extraction model achieves an average increase of 41.57% in diagnostic accuracy across five classes compared to CNN-DRF. This demonstrates that the integration of multi-scale features and attention mechanisms significantly enhances the model's feature learning capabilities. Compared to CBAM-MSCNN and CASM-MSCNN, the MCSA-DRF model improves average diagnostic accuracy by 4.82% and 2.06%, respectively. This indicates that the parallel attention mechanism in MCSA-DRF is more effective at enhancing key features than the sequential structures of CBAM and CASM. Compared to MSCNN-DRF, the MCSA-DRF model improves the average diagnostic accuracy for classes 3, 4, and 5 by 13.9%. This improvement is due to the attention enhancement mechanism in MCSA-DRF, which effectively filters out redundant features introduced by the multi-scale structure. Additionally, the DMCSA-DRF model increases the average diagnostic accuracy across five classes by 4.92% compared to MCSA-DRF. This indicates that the GASF and GADF images constructed from GAF encoding contain some uncorrelated feature information, and the dual-branch structure can more effectively learn critical feature information. The proposed dual-branch DMCSA-DRF model significantly enhances the representation of key features by fully extracting and utilizing multi-scale feature information, thereby achieving accurate recognition of different gearbox states. The performance

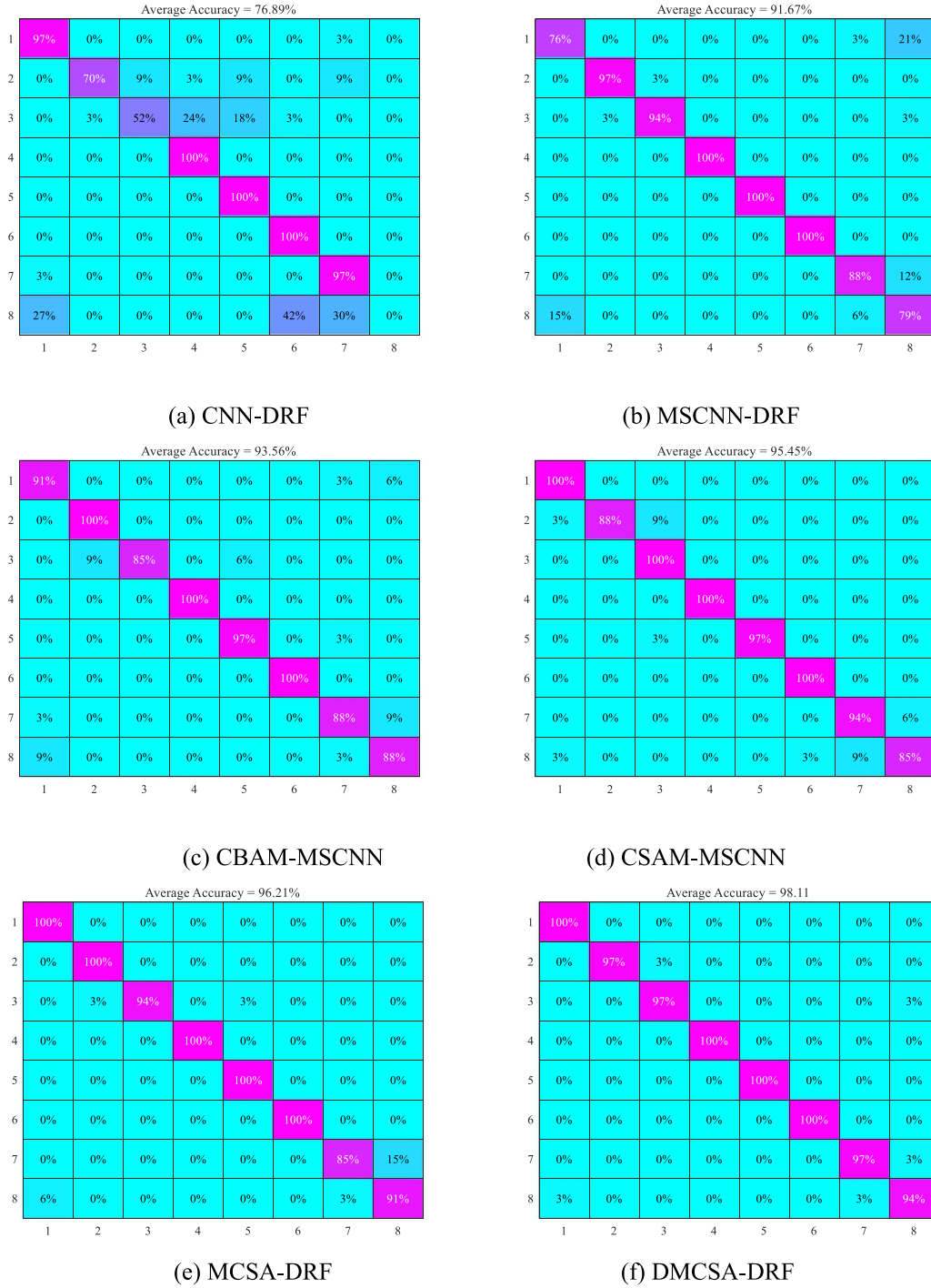


Figure 17. Confusion matrix for task A → E under different methods.

evaluation results of different models on the PHM2009 dataset are shown in figure 19.

As shown in figure 19, the CNN-DRF model exhibits a relatively low Precision, particularly evident in the 8th class, where the misdiagnosis rate reaches 100%. All faults of the 8th class are misclassified as the 1st, 6th, and 7th classes. Compared to CNN-DRF, the model MSCNN-DRF reduces the misclassification rate on class 8—21%, and the

Marco Recall and Marco F1-score of MSCNN-DRF are improved by 19.22% and 29.1%, respectively, which suggests that the proposed multiscale structure significantly improves the overall diagnostic performance of the model. DMCSA-DRF demonstrates improvements in Marco Recall of 27.6% and 7.02% compared to CNN-DRF and MSCNN-DRF, respectively. Additionally, DMCSA-DRF achieves enhancements in Marco F1-score of 38% and 6.11%, respectively.



Figure 18. Confusion matrix for task B → A under different methods.

These findings validate the robust feature extraction performance of the proposed dual-branch multi-scale attention structure. Compared to CBAM-MSCNN and CSAM-MSCNN, DMCSA-DRF exhibits improvements in Marco Recall of 6.88% and 4.82%, respectively. Moreover, DMCSA-DRF achieves enhancements in the Marco F1-score of 8.86% and 6.8%, respectively. Additionally, it achieves high diagnostic accuracy across all fault categories, further confirming the superiority of the proposed model.

3.4.2.4. Performance validation of model structure under different source domain conditions. To further verify the generalization performance of the model, the experiment involved setting data from different operating conditions sequentially as the source domain and other conditions as the target domain, to experimentally validate the diagnostic performance of different models. Each model was subjected to five experiments for each transfer task, and the mean classification accuracy was taken as the final evaluation metric, as depicted in figure 20.

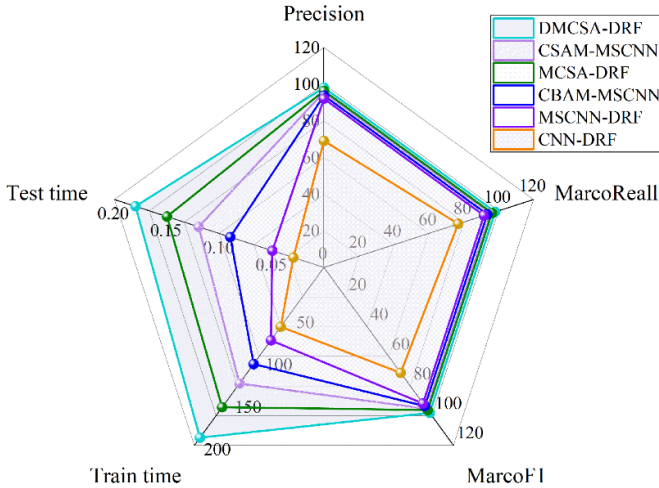


Figure 19. Performance validation of different models on the PHM2009 dataset.

Figures 20(a)–(e) depict the model evaluation results when conditions A, B, C, D, and E serve as the labeled source domain data, with the other four conditions acting as the target domain data, respectively. The CNN-DRF model achieved the lowest average accuracy of 72.36% in transfer tasks across five different conditions, whereas the multi-scale MSCNN-DRF model achieved an accuracy of 90.2%. The multi-scale MSCNN-DRF model improved accuracy by 24.64% compared to the single-scale CNN-DRF, indicating that multi-scale features contain richer fault information, which effectively enhances the model's diagnostic performance. The average accuracy of the MCSA-DRF proposed in this paper on the target conditions is 96.26%, while MSCNN-CBAM and CSAM-MSCNN obtain average accuracies of 92.87% and 95.2%, respectively, on these conditions. In comparison to the MCSA-DRF, the accuracies of MSCNN-CBAM and CSAM-MSCNN decrease by 3.52% and 1.1%, respectively. In comparison to the single-branch structure of MCSA-DRF, DMCSA-DRF demonstrates a 3.73% increase in average accuracy on the target conditions. This increase supports the effectiveness of DMCSA-DRF's dual-branch design, which incorporates both GADF and GASF inputs simultaneously to learn the underlying fault feature information within the data more effectively and comprehensively.

Figure 21 presents the evaluation metrics of different models on the target domain condition A. Notably, the Precision of the CNN-DRF model is subpar, particularly evident in class 3 and class 5, with misclassification rates of 47.67% and 62%, respectively. In contrast to the CNN-DRF model, both the MSCNN-DRF and MCSA-DRF models demonstrate decreased misclassification rates in class 3, reducing to 13% and 8.33%, and in class 5, decreasing to 14.67% and 5%, respectively. Analysis of the macro indicator Marco F1-score indicates a notable increase in performance for both the MSCNN-DRF and MCSA-DRF models by 33.53% and 43.16%, respectively, underscoring the significant enhancement in diagnostic accuracy facilitated by the

multi-scale attention structure. Compared to CBAM-MSCNN and CSAM-MSCNN, DMCSA-DRF achieves improvements in the MarcoF1-score of 9.58% and 6.67%, respectively, further validating the superiority of the proposed model.

Figure 22 shows the evaluation results of different models in transfer tasks under various conditions. The CNN-DRF feature extraction model had the poorest diagnostic performance when two different conditions were used as the source domain, achieving the lowest accuracy of 67.49%. The average accuracy of MSCNN-DRF on the target condition is 88.95%, which is 31.8% higher than that of the single-scale CNN-DRF, but 7.32% lower than that of MCSA-DRF. This is attributed to the multi-scale attention mechanism in MCSA, which not only provides rich feature information at different scales but also enhances the model's ability to extract key features. The average accuracy of MSCNN-CBAM and CSAM-MSCNN on the target condition is 91.94% and 93.92%, respectively. In comparison, the proposed DMCSA-DRF achieves an average accuracy of 99.94% on the target condition, which is 6.02% and 4.48% higher, respectively. This demonstrates that the parallel multi-scale attention enhancement mechanism proposed in this paper effectively improves the model's diagnostic performance. Compared to the single-branch MSCA-DRF, the DMCSA-DRF model with a dual-input branch structure achieves a 4.69% higher average accuracy on the target condition. This enhancement demonstrates that the dual-branch structure of DMCSA-DRF offers richer input feature information, allowing the model to capture more fault information.

3.4.3. Ablation experiments compared with other models.

To further evaluate the performance of the DMCSA-DRF model, this study compares it with other deep learning models. The comparison models include the residual network (ResNet) [63], LSTM networks [64], and GRU [65]. The GADF and GASF image data, encoded via Gram Angle Field, are used as inputs for the DMCSA-DRF model, while the vibration signal data, sampled and cropped, are input to the ResNet, LSTM, and GRU models. Figure 23 illustrates the accuracy and loss trends for different models for training.

As observed in figure 23(a), the DMCSA-DRF model demonstrates the fastest accuracy improvement on the training set, stabilizing after 100 iterations with a final accuracy approaching 100%, well above the performance of the other models. The LSTM model achieves the second-best training accuracy, reaching its peak after 300 iterations. However, it exhibits a slower convergence rate and more significant accuracy fluctuations. The GRU model performs slightly worse, with slower accuracy gains and more pronounced fluctuations. The ResNet model experiences the slowest accuracy improvement, and its final accuracy is lower than that of the other models. In the loss curve presented in figure 23(b), the DMCSA-DRF model achieves the fastest reduction in loss, stabilizing after approximately 100 iterations and reaching the lowest final loss. The ResNet model shows the second-fastest loss reduction, with a steeper decline than DMCSA-DRF in the first 25 iterations. This is because the DMCSA-DRF model,

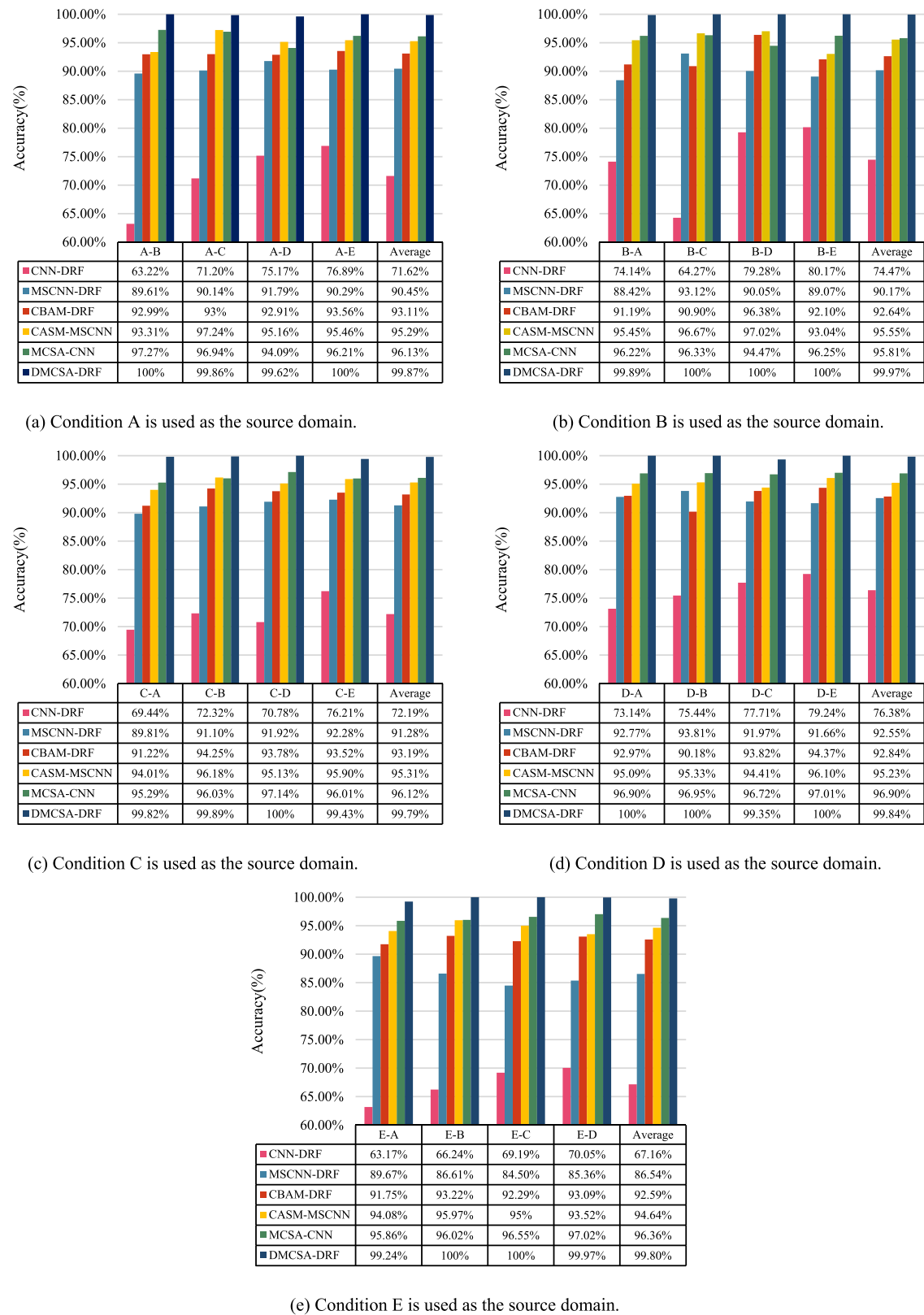


Figure 20. Prediction accuracy comparison of various models with different operating conditions as the source domain in the PHM2009 dataset.

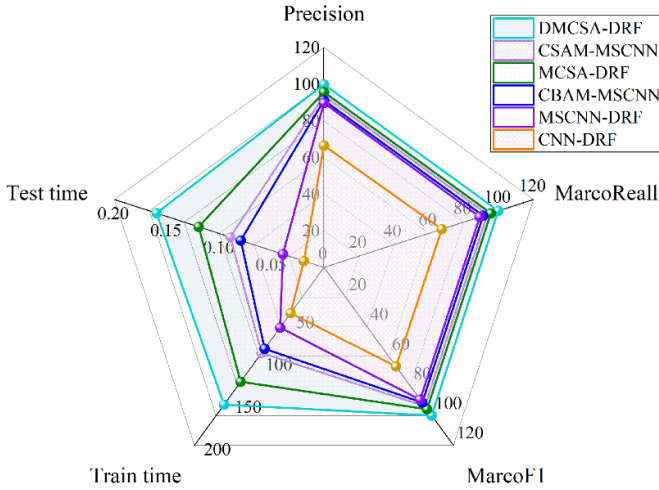


Figure 21. Performance validation of different models on the DDS dataset.

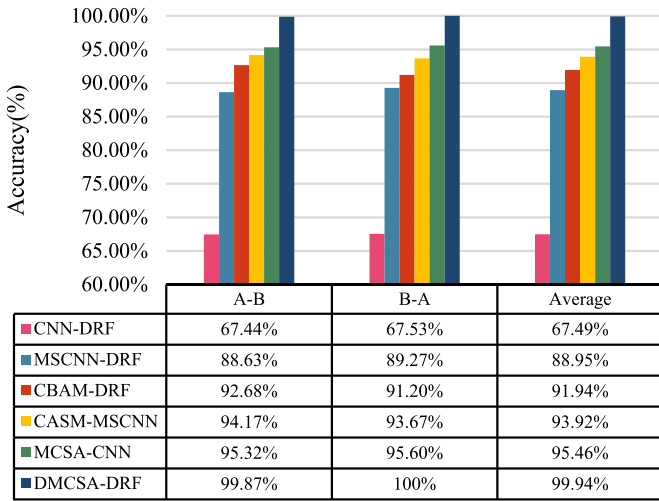
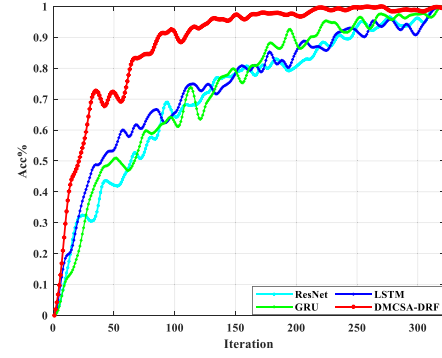


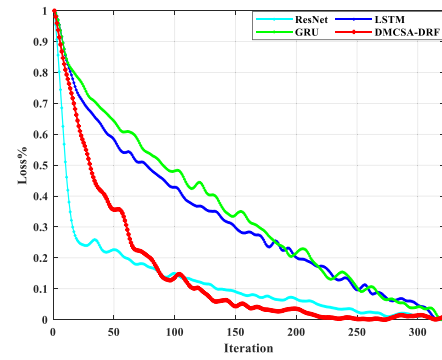
Figure 22. Prediction accuracy comparison of various models with different operating conditions as the source domain in the DDS dataset.

utilizing a dual-branch input structure, initially fits the data at a slower rate than ResNet. However, after the 25th iteration, the loss reduction rate of DMCSA-DRF exceeds that of all other models. The LSTM and GRU models demonstrate slower loss reduction rates and higher final loss values. In summary, the DMCSA-DRF model outperforms the other models in terms of both accuracy and convergence speed, providing superior diagnostic accuracy and training efficiency.

For further comparison of the models' classification performance, the T-SNE algorithm was employed to visualize the classification results of ResNet, LSTM, GRU, and DMCSA-DRF. T-SNE is capable of revealing the local structure and cluster characteristics within the data. As shown in figure 24, the visualization results use different numbers and colors to represent sample categories, with detailed category information listed in table 1. Increased dispersion among points of



(a) Training ACC



(b) Training LOSS

Figure 23. Training curves for different models.

different colors, and increased clustering among points of the same color, signifies higher recognition levels of the models for different fault types.

From figure 24(a), it is evident that Res-Net can separate different categories without any overlap between them. However, it divides class 2 into two parts, failing to achieve proper clustering for class 2. Additionally, the clustering for classes 3, 4, 5, 6, and 8 appears to be more dispersed. According to figure 24(b), the LSTM model successfully clusters all eight classes. Nevertheless, there is a slight overlap between class 8 and class 7, and the clustering of classes 2, 4, 5, and 6 appears to lack concentration. The visualization results for the GRU model are presented in figure 24(c). Compared to ResNet and LSTM, the clustering performance of GRU is slightly inferior for class 7, but relatively concentrated for other classes. However, apart from classes 3 and 5, there is a slight overlap among other categories. Compared to the other three models, the DMCSA-DRF model proposed in this paper achieves superior classification performance across all eight classes. Moreover, there is no overlap between different classes, ensuring accurate identification of all eight distinct fault classes.

As depicted in figure 25(a), ResNet can separate classes 1 and 2 from the other classes, but it performs poorly in clustering class 2, and there is substantial overlap among classes 3, 4, and 5. Figure 25(b) illustrates that LSTM can distinguish all five classes, although the distinction between classes 3 and

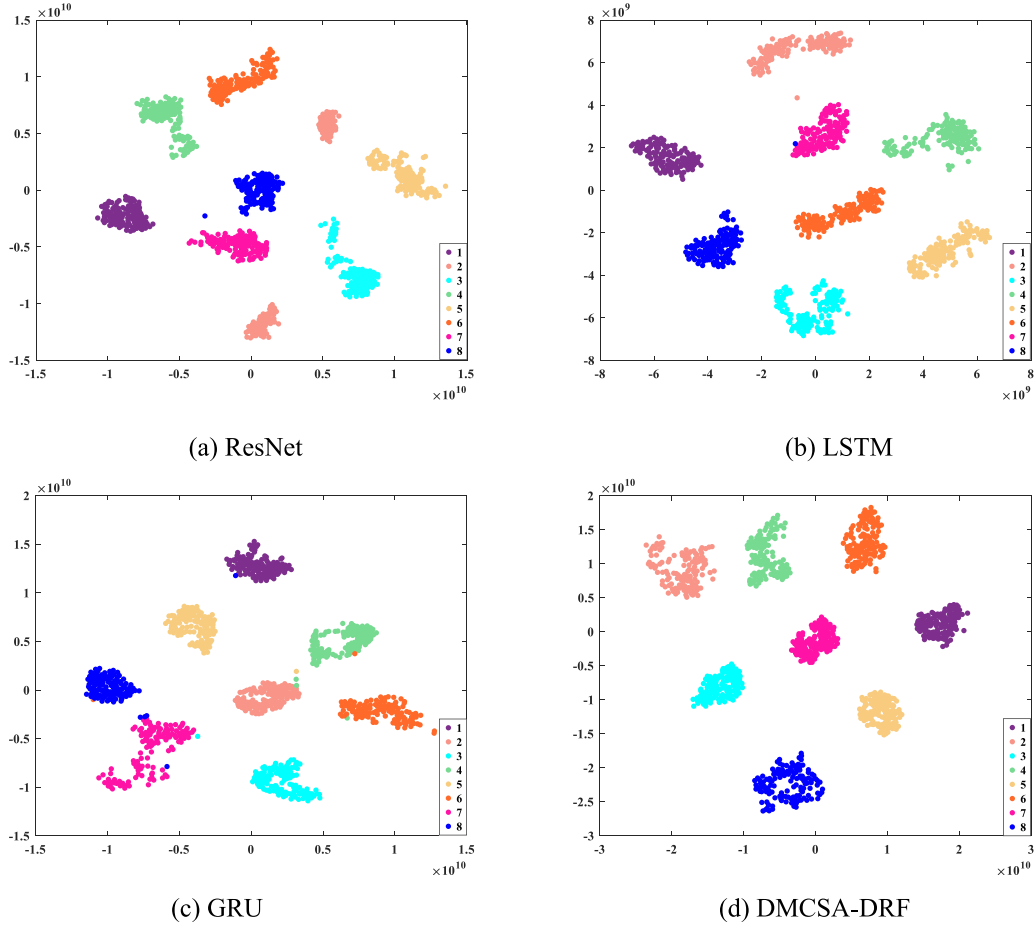


Figure 24. The T-SNE visualization results of different models in Tasks A–E: (a) ResNet; (b) LSTM; (c) GRU; (d) DMCSA-DRF.

4 is not pronounced, and there is slight overlap among the other classes. GRU can also distinguish all five classes, but there is some minor overlap between classes 4 and 5. The proposed DMCSA-DRF model achieves clear distinction among all five classes with no overlap, demonstrating its superiority. This advantage is due to the DMCSA-DRF model's multi-scale attention structure, which efficiently extracts crucial fault features, and the DRF module's capability to repeatedly utilize and learn from these multi-scale features, thereby improving the model's diagnostic accuracy.

To further validate the diagnostic performance of the proposed model and comparison models, a comparison is made based on the learnable parameters, FLOPs, and testing time of the models. The comparison results are shown in table 10.

Table 10 indicates that the LSTM model has the fewest learnable parameters (1.5 M) and the lowest computational complexity (24.5 G), leading to the fastest diagnostic speed, although its diagnostic accuracy ranks third. The ResNet model, with 20.7 M parameters and 47.4 GFLOPs, ranks third in terms of parameter size and computational complexity while exhibiting the worst diagnostic performance. The GRU model, with 22.3 M parameters and 55.1 GFLOPs, achieves second place in diagnostic accuracy. In contrast, the proposed DMCSA-DRF model, with 1.9 M parameters,

ranks second in both computational complexity and testing time, yet it achieves the highest diagnostic accuracy. The results indicate that the proposed model provides a better balance between diagnostic accuracy and efficiency, thus meeting the real-time fault detection requirements in industrial applications.

3.4.4. Anti-noise performance verification. To evaluate the proposed model's resistance to noise, Gaussian white noise was used to replicate the noise interference commonly found in industrial processes. We introduced Gaussian white noise with signal-to-noise ratios (SNR) of -5 dB, 0 dB, 5 dB, 10 dB, and 15 dB into the original datasets across different operational scenarios. The noise resistance performance of various models was subsequently validated through distinct TL tasks. Each task was executed five times per SNR level, and the average classification accuracy was recorded as the final assessment metric. The results of the validation are depicted in figures 26 and 27.

From figures 26 and 27, it is evident that as the injected noise increases, the predictive performance of all models across different operating conditions tasks declines to varying degrees on both datasets. Figures 26(a) and (e) show that

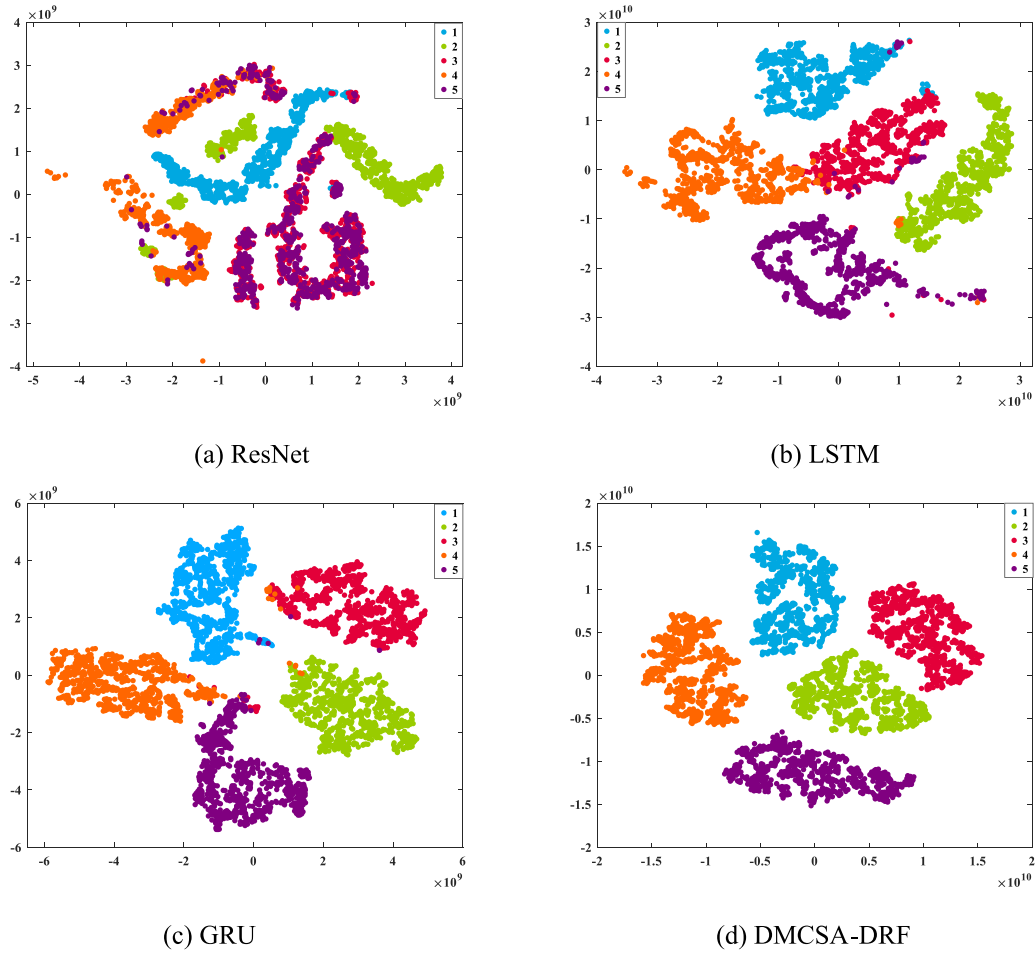


Figure 25. The T-SNE visualization results of different models in Tasks A-B: (a) ResNet; (b) LSTM; (c) GRU; (d) DMCSA-DRF.

Table 10. Comparative results of ablation experiments for different models.

Model	Attention	Parameters	FLOPs	Test time (s)	Acc rank
ResNet	CAE + SAE	20.7 M	47.4 G	1.56	4
LSTM		1.5 M	24.5 G	0.75	3
GRU		22.3 M	55.1 G	2.04	2
DMCSA-DRF		1.9 M	28.9 G	0.97	1

on the PHM2009 dataset, the average diagnostic accuracy of DMCSA-DRF decreases by 27.78% when noise with an SNR of -5 dB is introduced, compared to an SNR of 15 dB. Despite this decrease, the average diagnostic accuracy remains at 70.8%, which is still acceptable. With the addition of -5 dB noise, the average diagnostic accuracies for ResNet, LSTM, and GRU are 54.48%, 60.29%, and 63.39%, respectively. These values are 23.05%, 14.84%, and 10.47% lower than the diagnostic accuracy achieved by the proposed method. According to figures 27(a) and (b), on the DDS dataset, injecting noise with an SNR of -5 dB results in a 20.17%, 22.75%, and 15.73% decrease in the average diagnostic accuracies of the comparison models ResNet, LSTM, and GRU, respectively, when compared to the proposed DMCSA-DRF model.

The proposed model surpasses other models in diagnostic accuracy across all noise levels with SNR from -5 dB to 15 dB. Remarkably, even in the most adverse noise condition of -5 dB, its diagnostic accuracy is superior to that of the other three models. This is because, firstly, the other three comparison models solely accept vibration signals as input, whereas the model proposed in this study utilizes dual-branch GADF and GASF image data, exhibiting superior diversity and richness in input features compared to the other three models. Secondly, the proposed model's multi-scale attention-enhanced structure effectively enhances critical fault feature information and suppresses unnecessary features and noise. Furthermore, the model proposed in this paper employs an improved ITTAO algorithm for parameter optimization, while

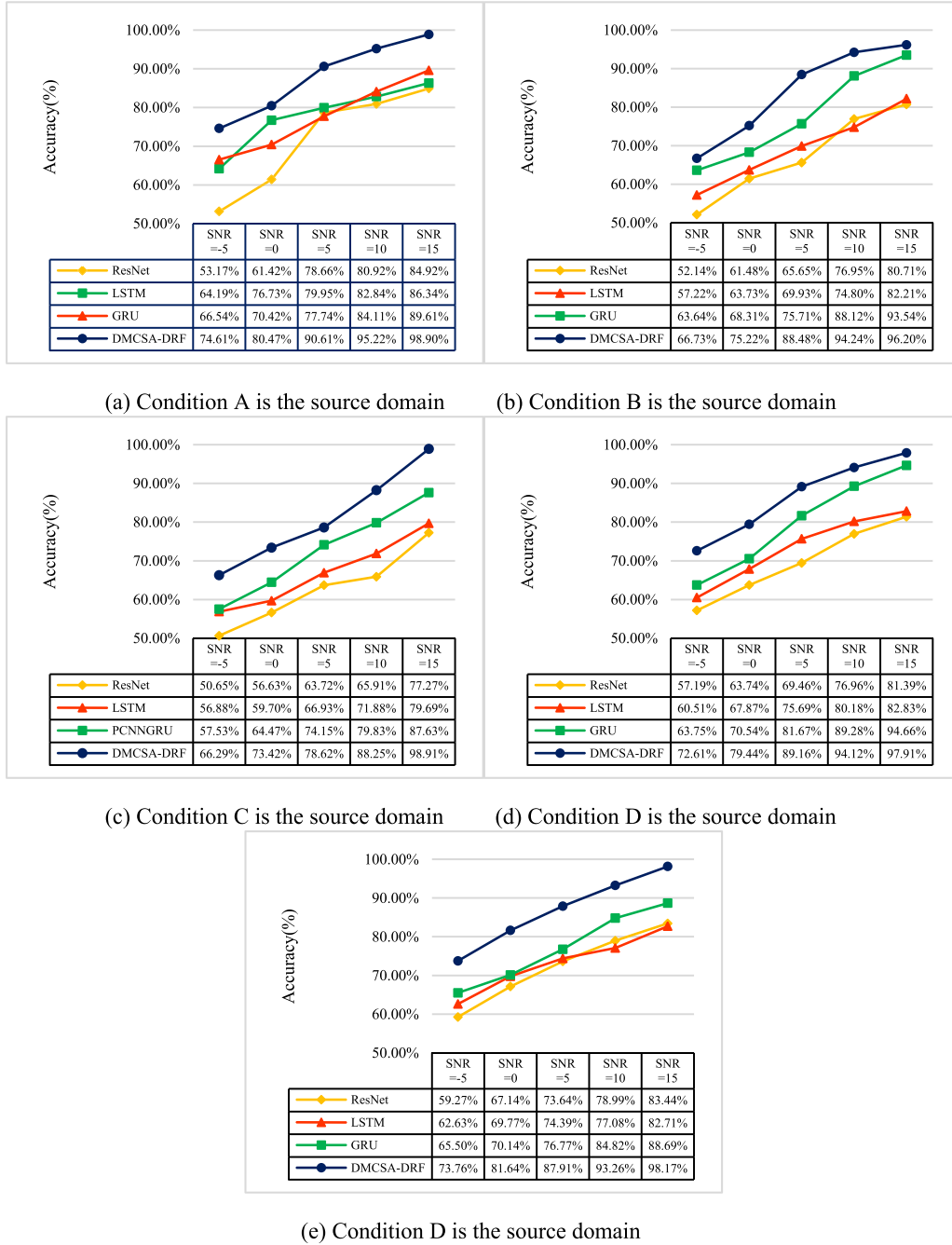


Figure 26. Comparison of classification accuracies of different models on the PHM2009 dataset under varying noise conditions.

the comparison models may suffer from suboptimal model parameters, resulting in lower diagnostic accuracy.

4. Conclusion

This paper presents an innovative lightweight and unsupervised method, ITTAO-DMCSA-DRF, for efficient cross-condition fault diagnosis in gearboxes. The proposed approach leverages GAF images as inputs, which provide enriched

feature representations, and incorporates a multi-scale attention mechanism to extract diverse and critical fault features. A dense residual structure is employed to enhance feature reuse and integration, further improving representational efficiency. To address domain discrepancies, the LMMD loss function aligns feature distributions between source and target domains, thereby boosting cross-condition diagnostic performance. Additionally, the improved ITTAO optimization algorithm refines both the model's structural parameters and its lightweight characteristics, enabling energy-efficient,

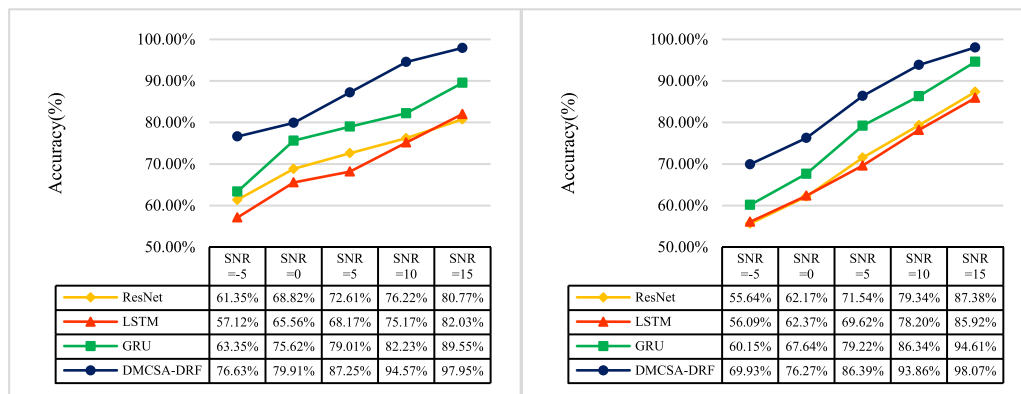


Figure 27. Comparison of classification accuracies of different models on the DDS dataset under varying noise conditions.

high-accuracy diagnosis while meeting industrial application requirements. Experimental results demonstrate that the proposed method achieved the highest prediction accuracies of 99.85% and 99.94% in transfer tasks across different conditions on two gearbox datasets, outperforming all comparison models. The main conclusions of this study are as follows:

- (1) A lightweight feature extraction model with multi-scale channel and spatial attention enhancement is proposed. This model enables efficient and accurate extraction and identification of critical gearbox fault features while consuming minimal computational resources. Furthermore, noise analysis experiments demonstrate that the improved attention mechanism effectively suppresses noise, thereby enhancing the model's robustness against noise interference.
- (2) The standard TTAO algorithm was improved by incorporating an adaptive synergistic elite weight dynamic balancing strategy, the Cauchy heavy-tailed distribution, and the tangent operator. These modifications enhance the algorithm's flexibility and diversity, strengthening its ability to escape local optima. Comparative evaluations of standard test functions against other classical algorithms validate their effectiveness. Furthermore, the ITTAO algorithm was employed for optimal hyperparameter tuning, optimizing the reduction ratio and increase ratio in CAE and SAE mechanisms. This optimization enables adaptive channel adjustment, thereby enhancing the representation of critical features.
- (3) The model uses GAF to encode one-dimensional signals into two-dimensional GASF/GADF image data, which is used as the input to the two-branch DMCSA for validation. Compared with MCSA with a single-branch structure, the prediction accuracy of DMCSA with a dual-branch structure is improved by 3.74% and 4.69% on the two datasets. The experimental results show that the GAF-based DMCSA with a dual-branch structure can still maintain a high diagnostic accuracy under sample-less conditions.

The proposed approach has notable limitations. It is constrained by the assumption that the fault modes in the source and target domains are identical, which limits the model's ability to detect new or unknown fault modes in the target domain during cross-condition fault diagnosis tasks. Additionally, the method has only been validated for diagnosing faults in a single type of gearbox across different operational conditions, neglecting the broader challenge of fault diagnosis in cross-equipment and cross-condition scenarios. Future work should address these gaps by developing methods capable of handling divergent fault modes between source and target domains and enabling robust fault diagnosis across equipment types and operational conditions. Simultaneously, further efforts to enhance the model's lightweight characteristics are crucial for practical implementation and deployment in industrial processes.

Data availability statement

No new data were created or analyzed in this study.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant Nos. 61563032, 61963025, 62203196, and 62463018), the Scientific and Technological Project of Gansu Province (Grant No. 24CXGA042), and the Gansu Provincial Science and Technology Program (Grant Nos. 22YF7GA164, 22CX8GA131, and 22CX3GD005).

Ethical Compliance

All procedures performed in studies involving human participants were following the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki Declaration and its later amendments or comparable ethical standards.

Conflict of interest

All authors declare that they have NO affiliations with or involvement in any organization or entity with any financial interest in the subject matter or materials discussed in this manuscript.

ORCID iDs

Wenchao Jia  <https://orcid.org/0009-0008-6149-7635>

Bin Gong  <https://orcid.org/0000-0002-5613-1419>

Yaoke Shi  <https://orcid.org/0000-0001-8533-3901>

References

- [1] Liang J *et al* 2020 A novel fault diagnostic method in power converters for wind power generation system *Appl. Energy* **266** 114851
- [2] Brożyna J, Strielkowski W, Fomina A and Nikitina N 2020 Renewable energy and EU 2020 target for energy efficiency in the Czech Republic and Slovakia *Energies* **13** 965
- [3] Szumilas-Kowalczyk H, Pevzner N and Giedych R 2020 Long-term visual impacts of aging infrastructure: challenges of decommissioning wind power infrastructure and a survey of alternative strategies *Renew. Energy* **150** 550–60
- [4] Xu X *et al* 2024 Total process of fault diagnosis for wind turbine gearbox, from the perspective of combination with feature extraction and machine learning: a review *Energy AI* **15** 100318
- [5] Gao Z and Liu X 2021 An overview on fault diagnosis, prognosis and resilient control for wind turbine systems *Processes* **9** 300
- [6] Kafeel A, Aziz S, Awais M, Khan M A, Afaq K, Idris S A, Alshazly H and Mostafa S M 2021 An expert system for rotating machine fault detection using vibration signal analysis *Sensors* **21** 7587
- [7] Zhang Y, Lv Y and Ge M 2021 Time–frequency analysis via complementary ensemble adaptive local iterative filtering and enhanced maximum correlation kurtosis deconvolution for wind turbine fault diagnosis *Energy Rep.* **7** 2418–35
- [8] Xu L, Chatterton S, Pennacchi P and Liu C 2020 A tacholeless order tracking method based on inverse short time Fourier transform and singular value decomposition for bearing fault diagnosis *Sensors* **20** 6924
- [9] Cai Y, Hao L, Zhou Y, Peterchev A V and Goetz S 2023 Rotating rectifier fault diagnosis of nuclear multi-phase brushless excitation system based on DTW metric and kNN classifier *IEEE Trans. Power Electron.* **39** 3289–99
- [10] Castellani F, Natili F, Astolfi D and Vidal Y 2024 Wind turbine gearbox condition monitoring through the sequential analysis of industrial SCADA and vibration data *Energy Rep.* **12** 750–61
- [11] Cheng J, Pan H, Zheng J and Tong J 2025 Global optimal Ramanujan spectrum: a feature extraction method without pseudo-monotonicity *Expert Syst. Appl.* **260** 125425
- [12] Miao Y, Zhang B, Li C, Lin J and Zhang D 2022 Feature mode decomposition: new decomposition theory for rotating machinery fault diagnosis *IEEE Trans. Ind Electron.* **70** 1949–60
- [13] Cheng J, Pan H and Zheng J 2024 Maximum Ramanujan spectrum signal-to-noise ratio deconvolution method: algorithm and applications *IEEE Trans. Ind. Inform.* **20** 11977–86
- [14] Peng Y, Qiao W, Cheng F and Della Santina C C 2021 Wind turbine drivetrain gearbox fault diagnosis using information fusion on vibration and current signals *IEEE Trans. Instrum. Meas.* **70** 1–11
- [15] Dhiman H S, Deb D, Muyeen S M and Kamwa I 2021 Wind turbine gearbox anomaly detection based on adaptive threshold and twin support vector machines *IEEE Trans. Energy Convers.* **36** 3462–9
- [16] Zhong T, Qu J, Fang X, Li H and Wang Z 2021 The intermittent fault diagnosis of analog circuits based on EEMD-DBN *Neurocomputing* **436** 74–91
- [17] Hu J, Liu Z, Chen J, Hu W, Zhang Z and Chen Z 2023 A novel deep learning–based fault diagnosis algorithm for preventing protection malfunction *Int. J. Electr. Power Energy Syst.* **144** 108622
- [18] Wang H, Liu Z, Peng D and Cheng Z 2022 Attention-guided joint learning CNN with noise robustness for bearing fault diagnosis and vibration signal denoising *ISA Trans.* **128** 470–84
- [19] Yang J, Bao W, Liu Y, Li X, Wang J, Niu Y and Li J 2021 Joint pairwise graph embedded sparse deep belief network for fault diagnosis *Eng. Appl. Artif. Intell.* **99** 104149
- [20] Zhang Z *et al* 2022 Discriminative sparse autoencoder for gearbox fault diagnosis toward complex vibration signals *IEEE Trans. Instrum. Meas.* **71** 1–11
- [21] Guo J, He Q, Gu F and Ball A D 2024 DMWMN: a deep modulation network for gearbox intelligent fault detection under variable working conditions *IEEE Trans. Syst. Man Cybern. Syst.* **54** 6082–92
- [22] Zhao Z and Jiao Y 2022 A fault diagnosis method for rotating machinery based on CNN with mixed information *IEEE Trans. Ind. Inform.* **19** 9091–101
- [23] Zhang B C, Sun S Q, Yin X J, He W-D and Gao Z 2023 Research on gearbox fault diagnosis method based on VMD and optimized LSTM *Appl. Sci.* **13** 11637
- [24] Zhao Z, Wang Q, Shao C, Chen N, Liu X-Y and Wang G-B 2023 A state detection method of offshore wind turbines' gearbox bearing based on the transformer and GRU *Meas. Sci. Technol.* **35** 025903
- [25] Chen Y, Liu X, Rao M, Qin Y, Wang Z and Ji Y 2025 Explicit speed-integrated LSTM network for non-stationary gearbox vibration representation and fault detection under varying speed conditions *Reliab. Eng. Syst. Saf.* **254** 110596
- [26] Jin Y, Hou L and Chen Y 2022 A time series transformer based method for the rotating machinery fault diagnosis *Neurocomputing* **494** 379–95
- [27] Jiang L, Shi C, Sheng H, Li X and Yang T 2024 Lightweight CNN architecture design for rolling bearing fault diagnosis *Meas. Sci. Technol.* **35** 126142
- [28] Wang Z, Lu H, Shi Y and Wang X 2023 Lightweight CNN architecture design based on spatial-temporal tensor and its application in bearing fault diagnosis *IEEE Trans. Instrum. Meas.* **73** 1–12
- [29] Li Y, Wang S and Xie J, Wang T, Yang J, Pan T and Yang B 2024 A lightweight dual-compression fault diagnosis framework for high-speed train bogie bearing *IEEE Trans. Instrum. Meas.* **73** 3530114
- [30] Zhao K, Xiao J, Li C, Xu Z and Yue M 2023 Fault diagnosis of rolling bearing using CNN and PCA fractal based feature extraction *Measurement* **223** 113754
- [31] Ruan D, Wang J, Yan J and Gühmann C 2023 CNN parameter design based on fault signal analysis and its application in bearing fault diagnosis *Adv. Eng. Inf.* **55** 101877
- [32] Wang H, Liu Z, Peng D and Zuo M J 2023 Interpretable convolutional neural network with multilayer wavelet for Noise-Robust Machinery fault diagnosis *Mech. Syst. Signal Process.* **195** 110314
- [33] Song B, Liu Y, Fang J, Liu W, Zhong M and Liu X 2024 An optimized CNN-BiLSTM network for bearing fault

- diagnosis under multiple working conditions with limited training samples *Neurocomputing* **574** 127284
- [34] An Y, Zhang K, Chai Y, Liu Q and Huang X 2023 Domain adaptation network base on contrastive learning for bearings fault diagnosis under variable working conditions *Expert Syst. Appl.* **212** 118802
- [35] Wang X, Jiang H, Mu M and Dong Y 2025 A dynamic collaborative adversarial domain adaptation network for unsupervised rotating machinery fault diagnosis *Reliab. Eng. Syst. Saf.* **255** 110662
- [36] Hu J, Shen L and Sun G 2018 Squeeze-and-excitation networks *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp 7132–41
- [37] Wang Q et al 2020 ECA-Net: efficient channel attention for deep convolutional neural networks *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition* pp 11534–42
- [38] Hu J, Shen L, Albanie S, Sun G, Vedaldi A 2018 Gather-excite: exploiting feature context in convolutional neural networks *Advances in Neural Information Processing Systems* p 31
- [39] Zhu Y, Pei Y, Wang A, Xie B and Qian Z 2023 A partial domain adaptation scheme based on weighted adversarial nets with improved CBAM for fault diagnosis of wind turbine gearbox *Eng. Appl. Artif. Intell.* **125** 106674
- [40] Hu Y, Li J, Huang Y and Gao X 2019 Channel-wise and spatial feature modulation network for single image super-resolution *IEEE Trans. Circuits Syst. Video Technol.* **30** 3911–27
- [41] Chen X, Yang R, Xue Y, Housden R J, Hajnal J and Rhode K 2023 Deep transfer learning for bearing fault diagnosis: a systematic review since 2016 *IEEE Trans. Instrum. Meas.* **72** 1–21
- [42] Li J, Lin M, Li Y and Wang X 2022 Transfer learning with limited labeled data for fault diagnosis in nuclear power plants *Nucl. Eng. Des.* **390** 111690
- [43] Ding Y, Jia M, Zhuang J, Cao Y, Zhao X and Lee C-G 2023 Deep imbalanced domain adaptation for transfer learning fault diagnosis of bearings under multiple working conditions *Reliab. Eng. Syst. Saf.* **230** 108890
- [44] Tang S, Ma J, Yan Z, Zhu Y and Khoo B C 2024 Deep transfer learning strategy in intelligent fault diagnosis of rotating machinery *Eng. Appl. Artif. Intell.* **134** 108678
- [45] Yang D, Zhang W and Jiang Y 2023 Mechanical fault diagnosis based on deep transfer learning: a review *Meas. Sci. Technol.* **34** 112001
- [46] Zhang H, Wang X, Zhang C, Li W, Wang J, Li G and Bai C 2023 Dynamic condition adversarial adaptation for fault diagnosis of wind turbine gearbox *Sensors* **23** 9368
- [47] Ma Y, Liu Y, Yang Z, Cheng M and Meng H 2023 Deep adversarial transfer neural network for fault diagnosis of wind turbine gearbox *Int. J. Green Energy* **20** 1750–62
- [48] Huang Q, Liang B, Dai X, Su S and Zhang E 2024 Unmanned aerial vehicle fault diagnosis based on ensemble deep learning model *Meas. Sci. Technol.* **35** 046205
- [49] Wang B, Qiu W, Hu X and Wang W 2024 A rolling bearing fault diagnosis technique based on recurrence quantification analysis and Bayesian optimization SVM *Appl. Soft Comput.* **156** 111506
- [50] Zhao Y, Song Z and Li D, Qian R and Lin S 2024 Wind turbine gearbox fault diagnosis based on multi-sensor signals fusion *Protect. Control Mod. Power Syst.* **9** 96–109
- [51] Shen J, Wang Y, Zhu H, Zhang L and Tang Y 2025 AGWO-PSO-VMD-TEFCG-AlexNet bearing fault diagnosis method under strong noise *Measurement* **242** 116259
- [52] Zhao S, Zhang T, Cai L and Yang R 2024 Triangulation topology aggregation optimizer: a novel mathematics-based meta-heuristic algorithm for continuous optimization and engineering applications *Expert Syst. Appl.* **238** 121744
- [53] Wang M, Wang J S, Li X D, Zhang M and Hao W-K 2022 Harris hawk optimization algorithm based on Cauchy distribution inverse cumulative function and tangent flight operator *Appl. Intell.* **52** 10999–1026
- [54] Mostafa R R, Gaheen M A, Abd Elaziz M, Al-Betar M A and Ewees A A 2023 An improved gorilla troops optimizer for global optimization problems and feature selection *Knowl.-Based Syst.* **269** 110462
- [55] Chen G, Xiang D, Liu T, Xu F and Li W 2024 Deep conditional adversarial subdomain adaptation network for unsupervised mechanical fault diagnosis *Knowl.-Based Syst.* **300** 112180
- [56] Shao S, McAleer S, Yan R and Baldi P 2018 Highly accurate machine fault diagnosis using deep transfer learning *IEEE Trans. Ind. Inform.* **15** 2446–55
- [57] Li G, Ao J, Hu J, Hu D, Liu Y and Huang Z 2024 Dual-source Gramian angular field method and its application on fault diagnosis of drilling pump fluid end *Expert Syst. Appl.* **237** 121521
- [58] Jia H, Rao H, Wen C and Mirjalili S 2023 Crayfish optimization algorithm *Artif. Intell. Rev.* **56** 1919–79
- [59] Su Y, Dai Y and Liu Y 2021 A hybrid hyper-heuristic whale optimization algorithm for reusable launch vehicle reentry trajectory optimization *Aerosp. Sci. Technol.* **119** 107200
- [60] Liu Y, Shao Z and Hoffmann N 2021 Global attention mechanism: retain information to enhance channel-spatial interactions (arXiv:2112.05561)
- [61] Xu Q, Jiang H, Zhang X, Li J and Chen L 2023 Multiscale convolutional neural network based on channel space attention for gearbox compound fault diagnosis *Sensors* **23** 3827
- [62] Du X, Jia W, Yu P, Shi Y and Cheng S 2022 A remaining useful life prediction method based on time–frequency images of the mechanical vibration signals *Measurement* **202** 111782
- [63] Chen W, Sun K, Li X, Xiao Y, Xiang J and Mao H 2023 Adaptive multi-channel residual shrinkage networks for the diagnosis of multi-fault gearbox *Appl. Sci.* **13** 1714
- [64] Shi J, Peng D, Peng Z, Zhang Z, Goebel K and Wu D 2022 Planetary gearbox fault diagnosis using bidirectional-convolutional LSTM networks *Mech. Syst. Signal Process.* **162** 107996
- [65] Xie F, Wang G, Shang J, Sun E and Xie S 2023 Gearbox fault diagnosis based on multi-sensor deep spatiotemporal feature representation *Mathematics* **11** 2679



A multi-variable driven dual-stage modal-decoupling framework integrating deterministic–uncertainty modeling for wind power forecasting with feature interpretability analysis

Wenchao Jia^a, Aimin An^{a,c,*}, Bin Gong^a, Yaoke Shi^b, Zheming Yan^a

^a College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou, 730050, People's Republic of China

^b School of Computer and Communication, Lanzhou University of Technology, Lanzhou, 730050, People's Republic of China

^c Key Laboratory of Gansu Advanced Control for Industrial Processes, Lanzhou University of Technology, Lanzhou, 730050, People's Republic of China

ARTICLE INFO

Keywords:

Wind power
Modal structure
Uncertainty interval
Multivariate coordination
Parameter optimization

ABSTRACT

Wind power signals exhibit strong non-stationarity, high-frequency noise, and temporal coupling characteristics, which make it difficult for traditional models to effectively capture feature representations, temporal dependencies, and prediction reliability. To address the challenges of non-stationarity, multi-scale temporal dynamics, and uncertain interval estimation, this study proposes a unified deterministic–probabilistic deep ensemble forecasting framework for wind power prediction. First, the complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) is employed to decompose the original power series, followed by K-means clustering to reconstruct intrinsic mode functions (IMFs). The high-frequency components are further processed using feature model decomposition (FMD) to reduce mode mixing and enhance feature separability. Then, a hybrid deep forecasting model is constructed by integrating a multi-scale temporal convolutional network (MSTCN), bidirectional gated recurrent unit (BiGRU), and non-stationary self-attention (NSA) mechanism to jointly model multi-scale dynamic and contextual dependencies, thereby improving prediction accuracy and stability. In addition, a multi-strategy improvement black-winged kite algorithm (MSBKA) is designed for multi-strategy optimization of model parameters, enhancing hyperparameter search efficiency and global convergence capability. Finally, an improved non-crossing quantile regression (INCQR) method is introduced by combining H-Pinball loss and Softplus mapping to ensure structural consistency and monotonicity of prediction intervals, thus improving the reliability of uncertainty quantification. Experimental results demonstrate that the proposed model achieves the highest fitting accuracy across four seasonal datasets ($R_{spr}^2 = 0.9533$, $R_{sum}^2 = 0.9628$, $R_{aut}^2 = 0.9802$, and $R_{win}^2 = 0.9810$) and exhibits superior consistency and coverage of prediction intervals (PICP = 92.65 %, PINAW = 0.1044).

1. Introduction

1.1. Research motivation

In recent years, both global wind turbine installed capacity and power generation have steadily increased, making wind energy a crucial force in enhancing the share of renewable energy [1]. At the same time, wind power forecasting plays an increasingly vital role in wind farm operation and grid dispatching. However, the stochastic, non-stationary, and intermittent nature of wind energy poses significant challenges for grid integration [2], increasing the risk to system stability and

substantially raising the operational costs of wind farms. Therefore, accurately forecasting wind power has become a critical issue for ensuring the efficient operation of wind farms and the stable functioning of power systems [3–5].

1.2. Research methods overview

In wind power prediction, signal preprocessing and model design are vital to achieving high accuracy [6,7]. Preprocessing addresses issues like non-stationarity, nonlinearity, and noise, yielding cleaner inputs for modeling [8,9]. Variational mode decomposition (VMD) [10], empirical

* Corresponding author. College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou, 730050, People's Republic of China.
E-mail address: anaimin@lut.edu.cn (A. An).

<https://doi.org/10.1016/j.energy.2025.139884>

Received 24 June 2025; Received in revised form 14 December 2025; Accepted 31 December 2025

Available online 6 January 2026

0360-5442/© 2025 Published by Elsevier Ltd.

mode decomposition (EMD) [11], complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) [12], and wavelet packet decomposition (WPD) [13]. Liu et al. [14] employed VMD to stabilize wind speed signals, integrating it with LSTM to improve short-term prediction accuracy. Similarly, Zhao et al. [15] utilized VMD to mitigate signal volatility and enhance spatiotemporal feature extraction, resulting in substantial improvements in forecasting performance. Although VMD effectively reduces noise and enhances stationarity, it has limited adaptability to highly nonstationary signals and requires a predefined number of modes. To address these limitations, some researchers have turned to EMD, which offers better nonlinear adaptability, higher temporal resolution, and lower parameter dependence [16]. Yuzgec et al. [17] developed an EMD-based model with echo state networks (ESN), which outperformed VMD-ESN and WD-ESN on multiple datasets. However, EMD is prone to mode mixing during decomposition, which affects accuracy [18,19].

To overcome the inherent limitations of EMD, the CEEMDAN introduces adaptive noise to effectively alleviate the mode mixing problem. It also eliminates the need to predefine the number of modes, thereby enhancing model adaptability. He et al. [20] proposed a CEEMDAN-LSTM framework, in which wind speed signals are first decomposed into several IMFs using CEEMDAN. Each IMF is then modeled separately using LSTM networks, and the final prediction is obtained by aggregating the outputs. In tests conducted at wind farms in Austria and Almeria, the model achieved mean absolute percentage errors (MAPE) as low as 0.3285 and 0.1455, respectively. Karijadi et al. [21] proposed a hybrid forecasting framework incorporating CEEMDAN, which significantly enhanced prediction accuracy under abrupt wind speed changes, highlighting CEEMDAN's strengths in robust feature extraction.

In wind power forecasting, while effective signal decomposition methods are essential for enhancing feature representation, the selection of an appropriate prediction model is equally critical in determining forecasting performance. Machine learning-based forecasting approaches improve the modeling capability for meteorological variables such as wind speed by establishing nonlinear mapping relationships between input features and power output. Commonly used models include support vector machines (SVM) [22], decision trees (DT) [23], random forests (RF) [24], K-nearest neighbors (KNN) [25], and artificial neural networks (ANN) [26]. Although these methods offer strong nonlinear modeling and high-dimensional feature processing capabilities, they typically rely heavily on feature engineering and exhibit limited ability to model complex temporal dependencies.

In recent years, deep learning has been widely applied to wind power prediction due to its end-to-end modeling capability and automatic feature extraction, effectively improving the modeling of complex nonlinear and temporal features. The main architectures include: (a) convolutional neural networks (CNN) [27,28], which extract local spatial features; (b) long short-term memory networks (LSTM) [29,30], which capture temporal dependencies via gating mechanisms; (c) gated recurrent units (GRU) [31,32], which simplify gating structures to enhance computational efficiency; and (d) transformer models [33,34], which leverage self-attention mechanisms to capture global contextual information. Wei et al. [35] proposed an LSTM model integrating maximal information coefficient (MIC) and multi-task learning (MTL) to jointly predict wind speed and power, enhancing multi-objective modeling. Xiao et al. [36] optimized the GRU model using weighted principal component analysis (WPCA) and particle swarm optimization (PSO), achieving superior performance on non-stationary wind power data. Bidirectional structures such as BiLSTM and BiGRU, which model forward and backward temporal dependencies simultaneously, have demonstrated stronger feature extraction and higher prediction accuracy [37,38]. Furthermore, to leverage the complementary advantages of spatial and temporal modeling, hybrid models have gained increasing attention. Gao et al. [39] proposed a CNN-GRU hybrid model that extracts spatial features from multi-source inputs via CNN and models

temporal dependencies using GRU, enabling more comprehensive feature representation. Sun et al. [40] designed a CNN-LSTM architecture to address complex spatiotemporal coupling in offshore wind power prediction, significantly improving forecasting performance. These studies indicate that hybrid deep learning architectures hold great promise for enhancing wind power prediction accuracy.

At present, most existing wind power forecasting methods adopt deterministic modeling, which provides a single future power value through point prediction and has achieved certain progress in improving forecasting accuracy. However, due to the inherent nonstationarity and randomness of wind power signals, deterministic prediction fails to fully characterize the uncertainty in the forecasting process, thereby limiting its applicability in grid scheduling and risk management [41,42]. In contrast, probabilistic modeling can effectively quantify prediction uncertainty through prediction intervals or probability distributions, offering more reliable risk assessment references for power system operation [43,44]. Nevertheless, relying solely on probabilistic modeling may enhance interval expressiveness at the expense of point prediction accuracy, leading to an overall decline in predictive performance. Therefore, integrating deterministic and probabilistic forecasting within a unified framework can simultaneously ensure forecasting accuracy and provide reasonable interval estimation, achieving a balance between precision and reliability—this is one of the core motivations of the forecasting framework proposed in this study. Among probabilistic modeling approaches, quantile regression (QR) has become mainstream because it can output prediction intervals under various confidence levels [45,46]. Jonkers et al. [47] proposed a CNN-QR-based regional wind power forecasting framework capable of flexibly generating arbitrary quantile distributions without retraining the model. Hu et al. [48] combined conformalized quantile regression (CQR) with temporal convolutional networks (TCN), achieving both efficient modeling and quantile calibration, thereby significantly improving interval prediction accuracy and reliability. However, traditional QR suffers from the quantile crossing problem, which leads to inconsistent and unreliable prediction intervals. To address this issue, some studies have introduced regularization and order-consistency constraints to ensure the monotonicity and stability of the predicted intervals [49,50].

Despite the remarkable progress achieved in wind power forecasting technology, several fundamental challenges remain to be addressed in practical applications.

- (1) Insufficient high-frequency modal analysis and representation: Existing methods often struggle to distinguish and suppress high-frequency features and noise, leading to inadequate capture of high-frequency dynamic information in wind power sequences. Residual noise further interferes with subsequent feature modeling [51,52].
- (2) Limited modeling of nonstationary features: Conventional deep forecasting models have difficulty simultaneously extracting local dynamic features and global temporal dependencies. Their limited capacity for nonstationary signal modeling compromises prediction accuracy and stability.
- (3) Low efficiency of model optimization: The hyperparameter optimization of complex models is typically inefficient, prone to local optima or slow convergence, thereby constraining overall performance and generalization ability [53].
- (4) Insufficient reliability in uncertainty prediction: Current approaches, such as quantile regression, often suffer from quantile crossing and inconsistent prediction intervals, resulting in unstable and unreliable uncertainty estimates [54].

To address the aforementioned challenges, this study develops an integrated forecasting framework for non-stationary wind power time series, characterized by cross-scale analysis, deep frequency-aware representation learning, and probabilistic consistency inference. The

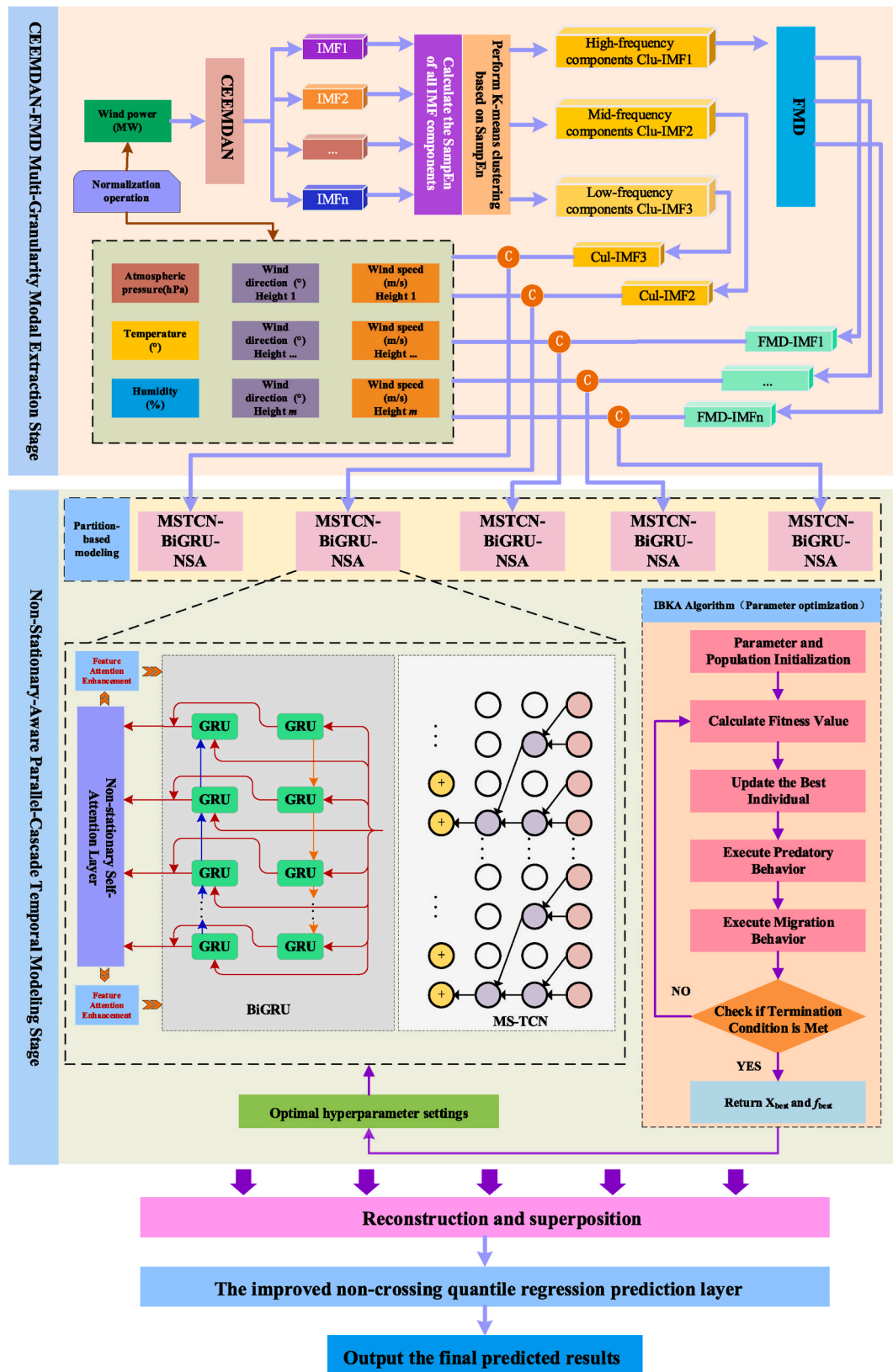


Fig. 1. MSBKA-MSTCN-BiGRU-NSA-INCQR overall prediction framework.

framework is built upon hierarchical modal decoupling, frequency-informed deep modeling, adaptive hyperparameter optimization, and structurally consistent interval estimation. Specifically, the cross-scale cooperative decomposition of CEMDAN-FMD, the frequency-fusion

prediction architecture of MSTCN-BiGRU-NSA, the multi-strategy global optimization mechanism of MS-BKA, and the consistency-preserving probabilistic modeling of INCQR are jointly integrated to form a systematic pipeline from signal analysis to predictive inference.

This design substantially enhances structural feature representation, forecasting accuracy, optimization robustness, and the reliability of uncertainty quantification. The major contributions of this work are as follows.

- (1) A hierarchical CEEMDAN-FMD modal decoupling mechanism is proposed to establish a macro-scale stabilization and micro-scale resolution enhancement cross-scale analytic scheme, thereby improving frequency-band discriminability and modal separation quality for non-stationary signals.
- (2) A modal-level frequency-aware deep prediction framework, MSTCN-BiGRU-NSA, is developed to jointly capture local dynamics, global dependencies, and non-stationary sensitivities, enabling comprehensive representation of multi-scale non-stationary features.
- (3) A multi-strategy enhanced MS-BKA optimization mechanism is introduced, where mutation-based reverse mapping and dynamic heavy-tailed perturbations significantly strengthen global search efficiency and convergence robustness in complex hyperparameter spaces.
- (4) An improved non-crossing quantile regression method (INCQR) is proposed, incorporating the H-Pinball loss and Softplus-based structural constraints to ensure interval monotonicity and consistency, thereby enhancing the reliability of uncertainty estimation.

1.3. Chapter content distribution

The first section of this paper discusses the current research status of wind power forecasting. The second section covers the theoretical background, including the fundamental theories and model structures employed in this study. The third section presents the experimental work, while the fourth section concludes the paper and offers prospects for future research.

2. Theoretical overview

2.1. Definition of the forecasting problem

The wind power forecasting task proposed in this study is formulated as a typical multivariate time series prediction problem. Its objective is to predict the wind power output at future time steps based on given historical observation sequences and meteorological environmental features.

At time step t , the input feature vector is expressed as:

$$\mathbf{X}_t = \{\mathbf{x}_t^{(1)}, \mathbf{x}_t^{(2)}, \dots, \mathbf{x}_t^{(d)}\} \quad (1)$$

where, d denotes the dimension of the input features, which include wind speed and wind direction measured at multiple heights of the meteorological mast (m_1, m_2, m_3, m_4), wind speed and wind direction at the turbine hub, as well as environmental variables such as temperature, atmospheric pressure, and humidity.

Given a historical observation sequence of length L :

$$\mathbf{X}_{t-L+1:t} = \{\mathbf{X}_{t-L+1}, \dots, \mathbf{X}_t\} \quad (2)$$

The objective of the forecasting task is to construct a nonlinear function $F(\cdot)$ that predicts the wind power output over the forthcoming h time horizons:

$$\hat{\mathbf{y}}_{t+h} = F(\mathbf{X}_{t-L+1:t}, \Theta) \quad (3)$$

where, Θ denotes the model parameters, and $\hat{\mathbf{y}}_{t+h}$ represents the predicted values.

In terms of the optimization objective, this study simultaneously focuses on deterministic and probabilistic forecasting.

- (a) Deterministic forecasting objective: The goal is to enhance point prediction accuracy by minimizing the error between the predicted values $\hat{\mathbf{y}}_{t+h}$ and the true observations $\mathbf{y}_{t+h}$:

$$\min_{\Theta} \mathbb{E}[l(\mathbf{y}_{t+h}, \hat{\mathbf{y}}_{t+h})] \quad (4)$$

where, $l(\cdot)$ can be selected from common error metrics such as RMSE.

- (b) Probabilistic forecasting objective: The prediction interval $[\hat{\mathbf{y}}_{t+h}^{\alpha_1}, \hat{\mathbf{y}}_{t+h}^{\alpha_u}]$ is constructed through quantile regression, which not only satisfies the desired coverage constraints but also avoids quantile crossing, thereby enhancing the structural consistency and reliability of the predictive intervals.

$$\mathbb{P}(\mathbf{y}_{t+h} \in [\hat{\mathbf{y}}_{t+h}^{\alpha_1}, \hat{\mathbf{y}}_{t+h}^{\alpha_u}] \approx 1 - \alpha) \quad (5)$$

Overall, the wind power forecasting problem is formulated as a comprehensive joint modeling framework that aims to achieve both high point forecasting accuracy and robust interval reliability.

2.2. Overall framework of the prediction model

To systematically characterize the multi-scale and strongly non-stationary dynamic evolution of wind power time series, this study develops a cross-scale driven hybrid forecasting framework from a unified perspective of signal structural analysis, temporal dependency representation, and uncertainty-consistent modeling. The framework is structurally grounded in a hierarchical CEEMDAN-FMD modal decoupling mechanism, which employs a cooperative strategy of macro-scale stabilization and micro-scale resolution enhancement to achieve multi-granular frequency-band reconstruction and modal refinement, thereby providing separable and stable modal representations for subsequent temporal modeling. Building upon this foundation, a modal-level frequency-aware deep modeling paradigm based on MSTCN-BiGRU-NSA is introduced to project the decomposed modal features into a unified temporal representation space, enabling the joint characterization of local dynamics, long-range dependencies, and non-stationary variation responses. Furthermore, a multi-strategy enhanced MS-BKA optimization mechanism is incorporated to improve search efficiency and convergence robustness in complex hyperparameter spaces. Finally, on the basis of deterministic forecasts, an improved INCQR method is employed to achieve reliable uncertainty quantification under structural consistency and monotonicity constraints. The overall framework is illustrated in Fig. 1.

The workflow of the proposed model consists of the following steps.

- (1) Primary modal decomposition: CEEMDAN is applied to decompose the input wind power series into multiple IMFs, reducing signal complexity.
- (2) K-means clustering: Sample entropy is calculated for each IMF, and the IMFs are clustered into high-, mid-, and low-frequency groups based on entropy values.
- (3) Secondary modal decomposition: The MS-BKA algorithm is used to optimize the parameters of FMD, after which the high-frequency group is further decomposed via FMD to simplify complex fluctuations. The resulting subcomponents are recombined with the medium- and low-frequency IMFs to form a refined IMF set.
- (4) Model training and optimization: Each IMF is modeled individually, and model parameters are fine-tuned using the MS-BKA algorithm by minimizing prediction error. The optimized MSTCN-BiGRU-NSA-INCQR model is then trained for integrated forecasting.
- (5) Model testing and assessment: The trained model is evaluated on the test dataset, with both deterministic and probabilistic forecasting performance metrics examined.

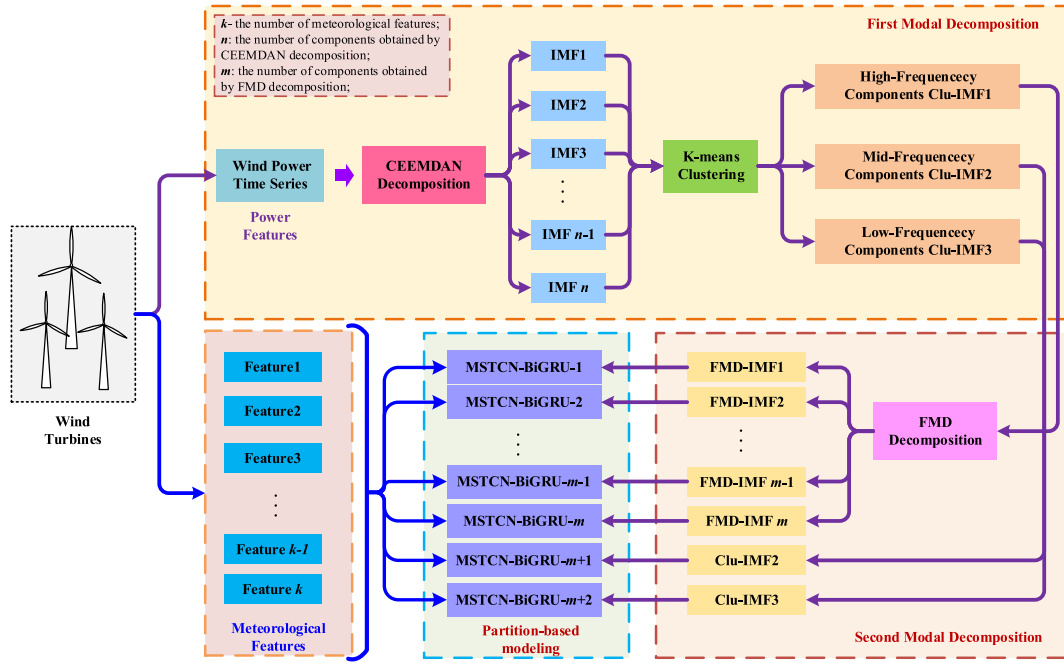


Fig. 2. Schematic diagram of two-stage modal decoupling.

2.3. Two-stage modal decoupling strategy

To improve high-frequency dynamic feature modeling and suppress the adverse effects of modal aliasing and noise, this paper adopts a two-stage modal decoupling framework based on CEEMDAN and FMD. The schematic representation of the proposed decoupling process is depicted in Fig. 2.

2.3.1. CEEMDAN-based mode decomposition

CEEMDAN [55] is an improved method based on EMD. CEEMDAN introduces adaptive noise components to enhance the completeness and robustness of signal decomposition. The CEEMDAN algorithm for signal decomposition is presented in Algorithm 1.

Algorithm 1

CEEMDAN decomposition algorithm

Input: Measured wind power time series $x(t)$;
CEEMDAN parameters: number of realizations N_e , standard deviation of added noise σ , and maximum number of decomposition iterations K .
Output: Intrinsic Mode Functions $\{IMF_1, IMF_2, \dots, IMF_K\}$ and residual signal $r(t)$, representing multi-scale dynamic components of wind power fluctuations.

1 **Signal standardization** Removes scale differences and suppresses extreme fluctuations.

$$x'(t) = \frac{x(t) - \mu_x}{\sigma_x}$$

where, μ_x and σ_x denote the mean and standard deviation of the wind power signal, respectively.

2 **Noise-assisted Ensemble Generation**

$$x_i(t) = x'(t) + \sigma w_i(t), i = 1, 2, \dots, N_e$$

where, $w_i(t)$ is Gaussian white noise. Enhances robustness against stochastic wind disturbances.

3 **Extraction of the First Intrinsic Mode Function (IMF₁)** Each noise-assisted realization is decomposed via EMD, and the ensemble-averaged first IMF is computed as:

$$IMF_1(t) = \frac{1}{N_e} \sum_{i=1}^{N_e} EMD(x_i(t))$$

(continued on next column)

Algorithm 1 (continued)

$$r_1(t) = x'(t) - IMF_1(t)$$

where, $IMF_1(t)$ captures high-frequency power variations induced by rapid wind speed changes, while $r_1(t)$ retains the lower-frequency dynamics associated with gradual power evolution.

4 **Iterative Extraction of Subsequent IMFs**

$$IMF_k(t) = \frac{1}{N_e} \sum_{i=1}^{N_e} EMD(r_{k-1}(t) + \sigma w_i(t)), k = 2, 3, \dots, K$$

$$r_k(t) = r_{k-1}(t) - IMF_k(t)$$

Each successive IMF represents a progressively lower-frequency oscillation component.

5 **Output and Physical Interpretation**

After K iterations, obtain:

Returns $\{IMF_1, \dots, IMF_K, r_K\}$.

High-frequency IMFs capture transient power fluctuations due to turbulence, while low-frequency IMFs and the residual reflect long-term operational and environmental trends.

The computational complexity of CEEMDAN primarily arises from multiple EMD executions and the iterative sifting operations involved in each decomposition. Assuming the signal length is N , the number of noise realizations is N_e , the total number of decomposed IMF is M , and the maximum number of EMD iterations is K . The algorithm mainly consists of three stages: noise generation, noise sample decomposition, and multiple noisy EMD extractions. Accordingly, the computational complexity of noise generation is denoted as $O(N \times N_e)$, that of noise sample decomposition as $O(N \times N_e \times M \times K)$, and that of multiple noisy EMD decompositions as $O(N \times N_e \times M \times K)$. Therefore, the overall time complexity of the CEEMDAN algorithm can be expressed as:

$$T_{CEEMDAN} = O(N \times N_e \times M \times K) \quad (6)$$

The computational complexity of CEEMDAN is linearly related to the signal length, the number of noise realizations, the number of modes, and the maximum number of sifting iterations.

2.3.2. FMD-based mode decomposition

Feature mode decomposition (FMD) is a non-iterative signal

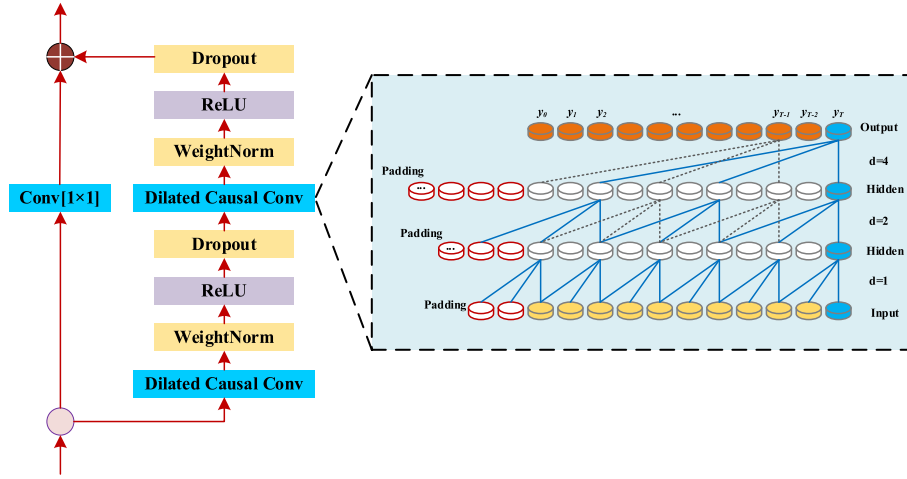


Fig. 3. TCN structure diagram.

processing method based on dynamic filter design [56]. In this study, the input to FMD is the high-frequency modal component obtained from CEEMDAN decomposition and subsequent K-means clustering. This component contains rapid power fluctuations caused by wind turbulence and gust effects, while also including noise interference. The objective of FMD is to adaptively separate the effective dynamic features from random disturbances within high-frequency signals, thereby providing more robust high-frequency representations for subsequent forecasting.

This process can be formulated as a constrained convex optimization problem:

$$\begin{aligned} \underset{\{f_k(l)\}}{\operatorname{argmax}} \{ & CK_M(u_k) = \sum_{n=1}^N \left(\prod_{m=0}^M u_k(n - mT_s) \right)^2 \\ & \left/ \left(\sum_{n=1}^N u_k(n)^2 \right)^{M+1} \right\} \\ \text{s.t. } & u_k(n) = \sum_{l=1}^L f_k(l)x(n - l + 1) \end{aligned} \quad (7)$$

where, u_k and X are defined as follows:

$$\begin{aligned} u_k &= \begin{bmatrix} u_k(1) \\ \vdots \\ u_k(N - L + 1) \end{bmatrix}, \\ X &= \begin{bmatrix} x(1) & \cdots & x(L) \\ \vdots & \ddots & \vdots \\ x(N - L + 1) & \cdots & x(N) \end{bmatrix}, f_k = \begin{bmatrix} f_k(1) \\ \vdots \\ f_k(L) \end{bmatrix} \end{aligned} \quad (9)$$

The correlated kurtosis (CK) of the decomposed mode is defined as follows:

$$CK_M(u_k) = \frac{u_k^H W_M u_k}{u_k^H u_k} \quad (10)$$

where, the superscript H denotes the conjugate transpose of a matrix. W_M is an intermediate variable used to regulate the weighted correlation matrix, and its corresponding expression is given as follows:

$$W_M = \begin{bmatrix} \left(\prod_{m=0}^M u_k(1 - mT_s) \right)^2 & 0 & \cdots & 0 \\ 0 & \left(\prod_{m=0}^M u_k(2 - mT_s) \right)^2 & & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \cdots & \left(\prod_{m=0}^M u_k(N - L + 1 - mT_s) \right)^2 \end{bmatrix} \frac{1}{\sum_{n=1}^{N-L+1} u_k^{M-1}(n)} \quad (11)$$

where, $u_k(n)$ denotes the k -th decomposed mode corresponding to the high-frequency component of the wind power signal; f_k represents the k -th FIR filter with a filter length of T_s denotes the estimated signal period; and M is the displacement order. In the context of wind power forecasting, u_k corresponds to the power fluctuation patterns within different frequency ranges.

The matrix form corresponding to the decomposed modes is expressed as follows:

$$u_k = X f_k \quad (8)$$

Substituting Equation (8) into Equation (10) yields the following expression:

$$CK_M(u_k) = \frac{f_k^H X^H W_M X f_k}{f_k^H X^H X f_k} = \frac{f_k^H R_{XW} f_k}{f_k^H R_{XX} f_k} \quad (12)$$

where, R_{XW} denotes the weighted correlation matrix, and R_{XX} represents the correlation matrix.

In Equation (12), the filter coefficients can be optimized by solving a generalized eigenvalue problem, where the maximum eigenvalue λ

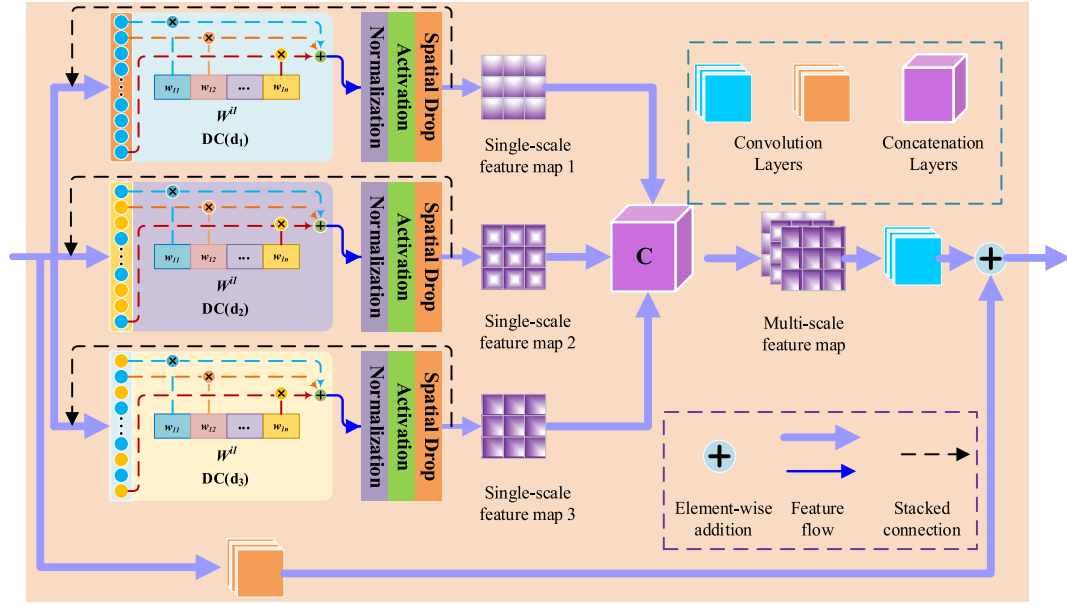


Fig. 4. MSTCN structure diagram.

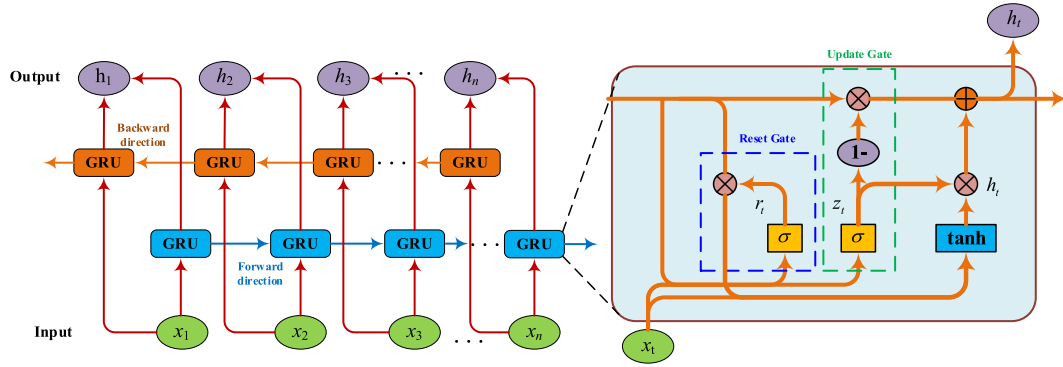


Fig. 5. BiGRU structure diagram.

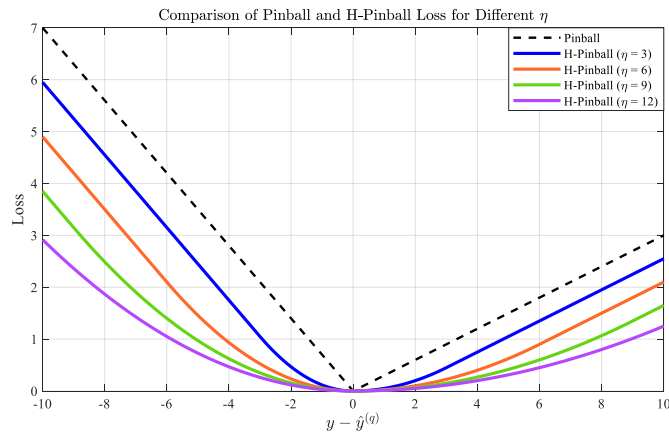


Fig. 6. Pinball loss function and H-Pinball loss function with different thresholds.

corresponds to the largest f coefficient.

$$R_{xwx}f_k = R_{xx}f_k\lambda \quad (13)$$

The filter coefficients corresponding to the maximum eigenvalue can maximize CK, thereby extracting the mode most strongly associated

with high-frequency power fluctuations.

To avoid redundant modes, the correlation coefficients between modes are computed after each filtering operation:

$$CC_{pq} = \frac{\sum_{n=1}^N (u_p(n) - \bar{u}_p)(u_q(n) - \bar{u}_q)}{\sqrt{\sum_{n=1}^N (u_p(n) - \bar{u}_p)^2} \sqrt{\sum_{n=1}^N (u_q(n) - \bar{u}_q)^2}} \quad (14)$$

where, u_p and u_q represent any two modal components, while $\bar{u}_p$ and $\bar{u}_q$ denote their respective mean values. By applying a joint constraint based on CK and the correlation coefficient, components exhibiting weak dynamic correlation with wind power variations are eliminated, retaining only the key modes. This iterative process continues until the predefined number of modes n is achieved, resulting in the final output $U_{HF} = \{u_1, u_2, \dots, u_n\}$. These modes physically correspond to the effective high-frequency power dynamics induced by wind turbulence and gust effects rather than unstructured noise. When combined with the medium- and low-frequency components derived from CEEMDAN, they form a multi-scale feature input used for both point and interval forecasting of wind power.

The computational complexity of FMD mainly arises from three stages: (1) Initialization of the filter bank, which requires generating B band-pass FIR filters, each of length L , with a computational complexity

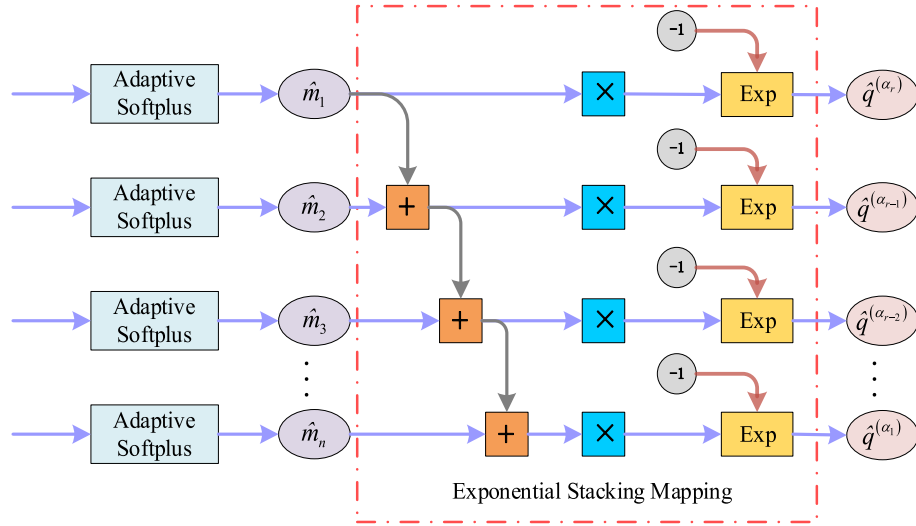


Fig. 7. Improved non-crossed quantile regression module.

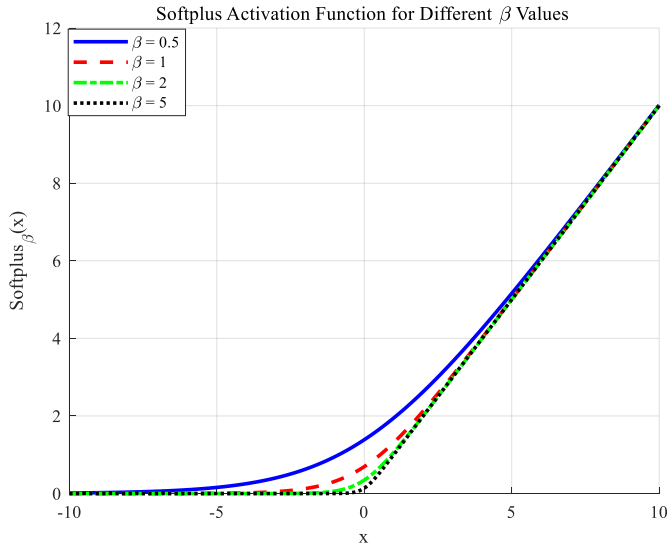
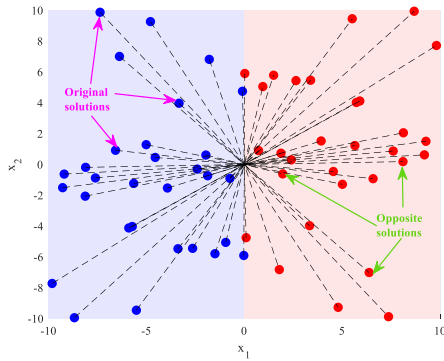
Fig. 8. Comparison of Softplus activation function ($\beta = 1$) and Softplus activation function under different β parameters.

Fig. 9. Schematic diagram of OBL solution space.

of $O(B \times L)$; (2) Iterative correlated-kurtosis deconvolution. For each sub-band signal, every iteration involves matrix construction and inversion $O(L^2N + L^3)$, matrix multiplication and filter updating $O(M \times L \times N)$ (where M denotes the shift order), and Hilbert-based autocorrelation period estimation $O(N \times \log N)$. If the number of iterations for each sub-band is R , the corresponding complexity is $O(R[L^2 + M \times L + \log N] + L^3)$; (3) Correlation-matrix computation and modal elimination, where $b \times b$ correlation matrices must be calculated per iteration, resulting in an accumulated time complexity of approximately $O(N \times B^3)$. By combining the above stages, the overall time complexity of the FMD algorithm can be expressed as:

$$T_{FMD} = O(B \times R[N \times (L^2 + M \times L + \log N) + L^3]) + O(N \times B^3) \quad (15)$$

Among these stages, the filter optimization and the iterative correlated-kurtosis deconvolution of sub-band signals constitute the primary sources of computational cost in the FMD algorithm.[57]

2.4. Model structure

2.4.1. Multi-scale temporal convolutional network structure

The TCN [57] is a type of neural network well-suited for processing time-series data. The TCN primarily consists of two key operations: dilated causal convolution and residual connection. The overall architecture of the TCN is illustrated in Fig.3.

Traditional single-scale TCNs utilize a fixed dilation rate in their convolutional layers, restricting the model to specific feature scales. Following the approach in Ref. [58], a multi-scale TCN with a parallel structure is introduced to enhance the extraction of multi-scale temporal features. The architecture of the proposed multi-scale TCN is shown in Fig. 4, and the corresponding algorithm is described below.



Fig. 10. The location of the wind farms. The map of China is from a freely available map website: <http://bzdt.ch.mnr.gov.cn/>.

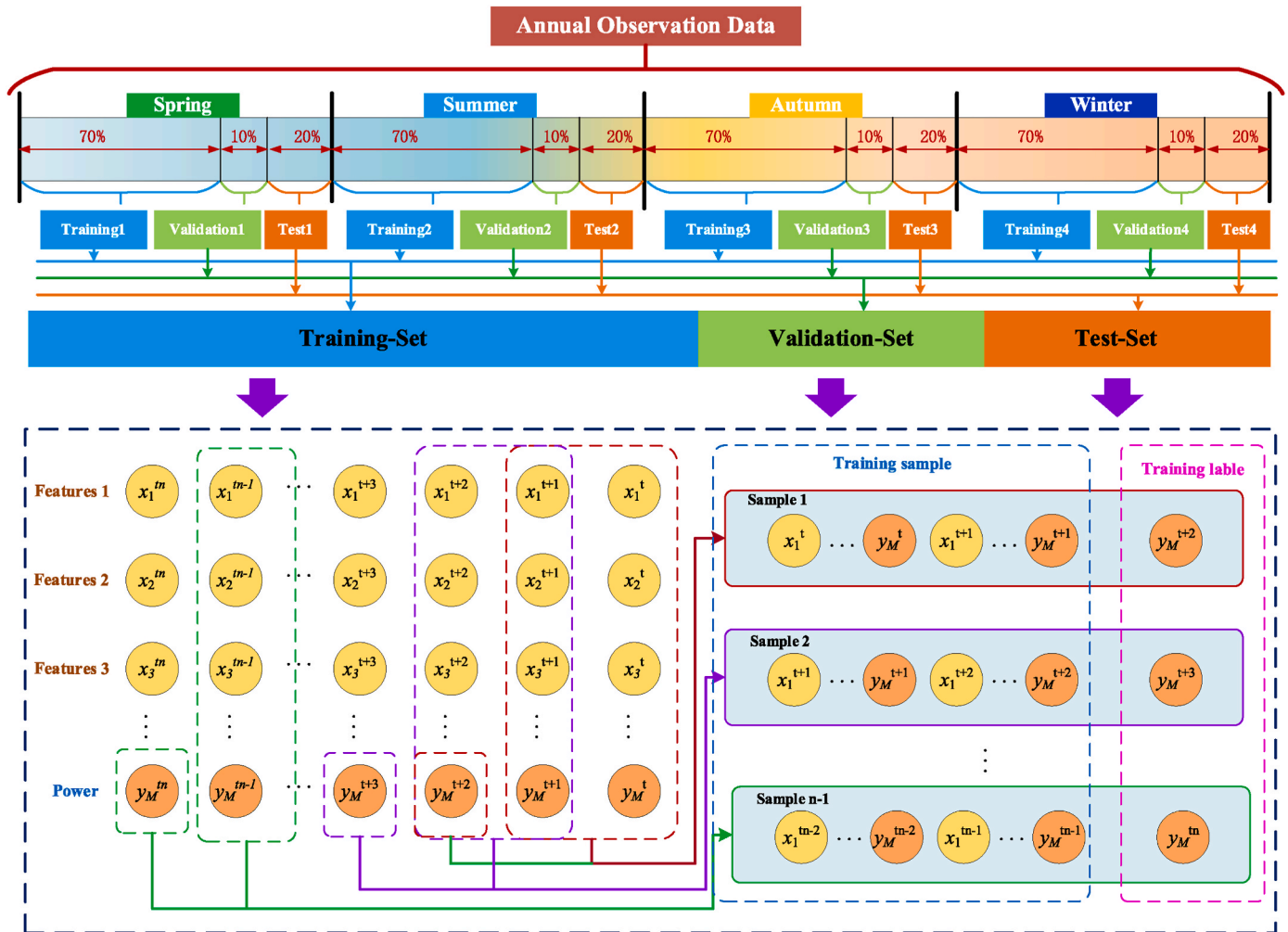


Fig. 11. Cascaded seasonal data isolation strategy.

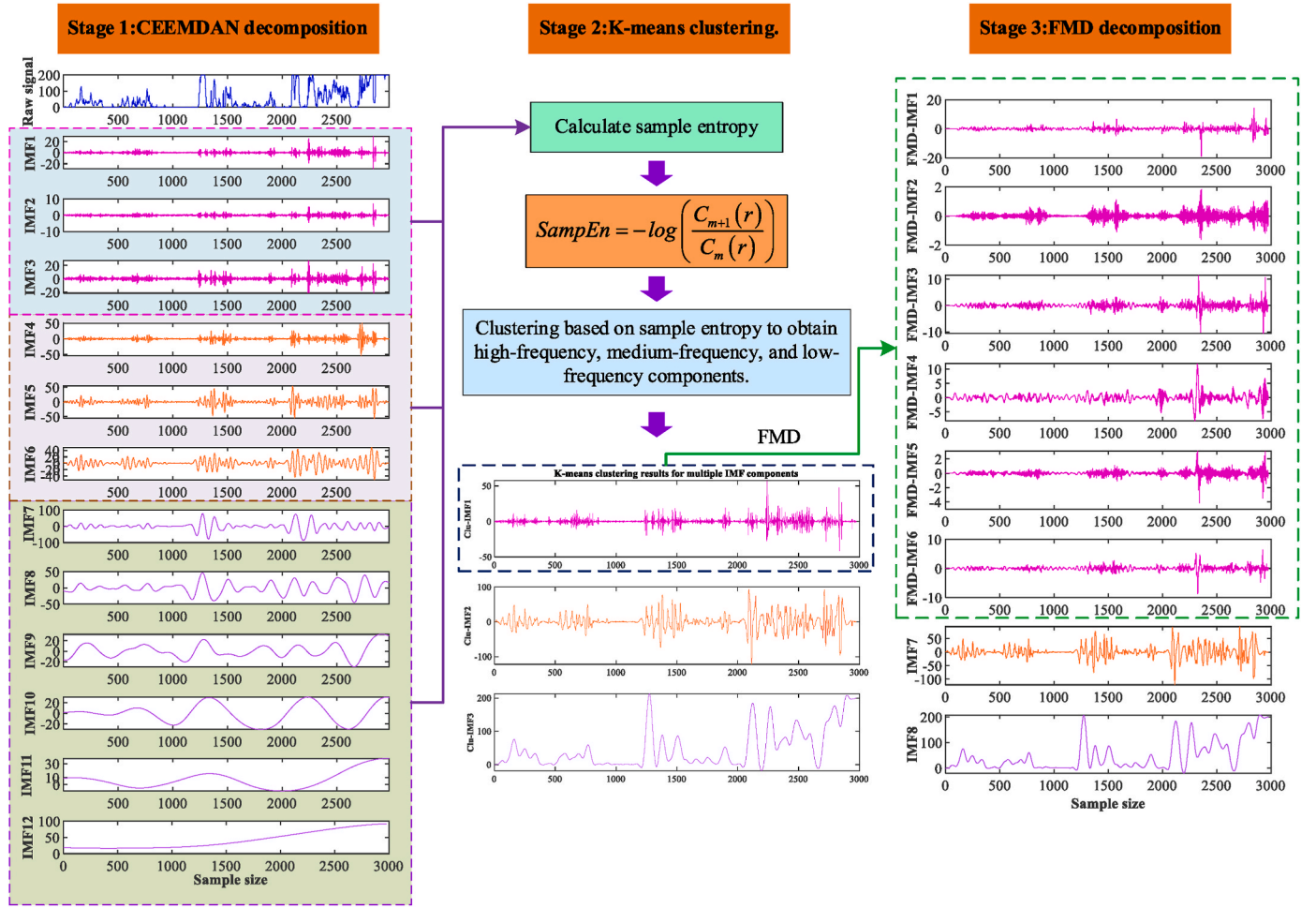


Fig. 12. CEEMDAN-FMD two-stage modal decoupling process diagram.

Algorithm 2**MSTCN**

Input: The time-series data $X \in \mathbb{R}^{N \times T}$ corresponding to multiple meteorological features, where N is the number of features and T denotes the number of time steps.
Output: The fused multi-scale features F_{out} .

1 Parallel Multi-Scale Feature Extraction

For each TCN branch, the following procedures are applied:

For j = 1 to 3 do:

(1) Feature computation in the first DDC layer:

$$F_{DC}^{ij} = W_{DC}^{ij} * X_{in} + b_{DC}^{ij}$$

(2) The batch normalization operation:

$$F_{BN}^{ij} = f_B(F_{DC}^{ij}, \gamma^{ij}, \beta^{ij})$$

(3) GELU Activation:

$$F_{GL}^{ij} = f_\Phi(F_{BN}^{ij})$$

(4) Dropout Factor

$$F_{SD}^{ij} = f_S(F_{GL}^{ij}, \tau)$$

(5) Feature computation in the second DDC layer:

$$F_{DC}^{2j} = W_{DC}^{2j} * F_{SD}^{1j} + b_{DC}^{2j}$$

(6) BN and GELU activation:

Algorithm 2 (continued)

$$F_{SD}^{2j} = f_S(f_\Phi(f_B(F_{DC}^{2j}, \gamma^{2j}, \beta^{2j})), \tau)$$

End for

2 Multi-scale feature fusion

(1) Concatenation operation, resulting in feature channels of $3 \times c$.

$$F_{MS} = f_{con}(F_{SD}^{21}, F_{SD}^{22}, F_{SD}^{23})$$

(2) The 1×1 convolution for dimensionality reduction, resulting in feature channels of size c .

$$F'_{MS} = W_{1 \times 1} * F_{MS} + b_{1 \times 1}$$

3 Residual connection, outputting multi-scale features.

$$F_{out} = f_c(F_{MS}) \oplus f_c(X_{in}) \\ = (W_1 * F_{MS} + b_1) \oplus (W_2 * X_{in} + b_2)$$

The Gaussian error linear unit (GELU) is employed in place of ReLU within the multi-scale TCN framework, introducing a more flexible activation mechanism than that used in conventional TCN. The GELU function applies a smooth probabilistic gating to the input by combining it with the cumulative distribution function of the standard normal distribution. This approach maintains nonlinear representational capacity while effectively mitigating issues such as gradient vanishing and neuron "death."

(continued on next column)

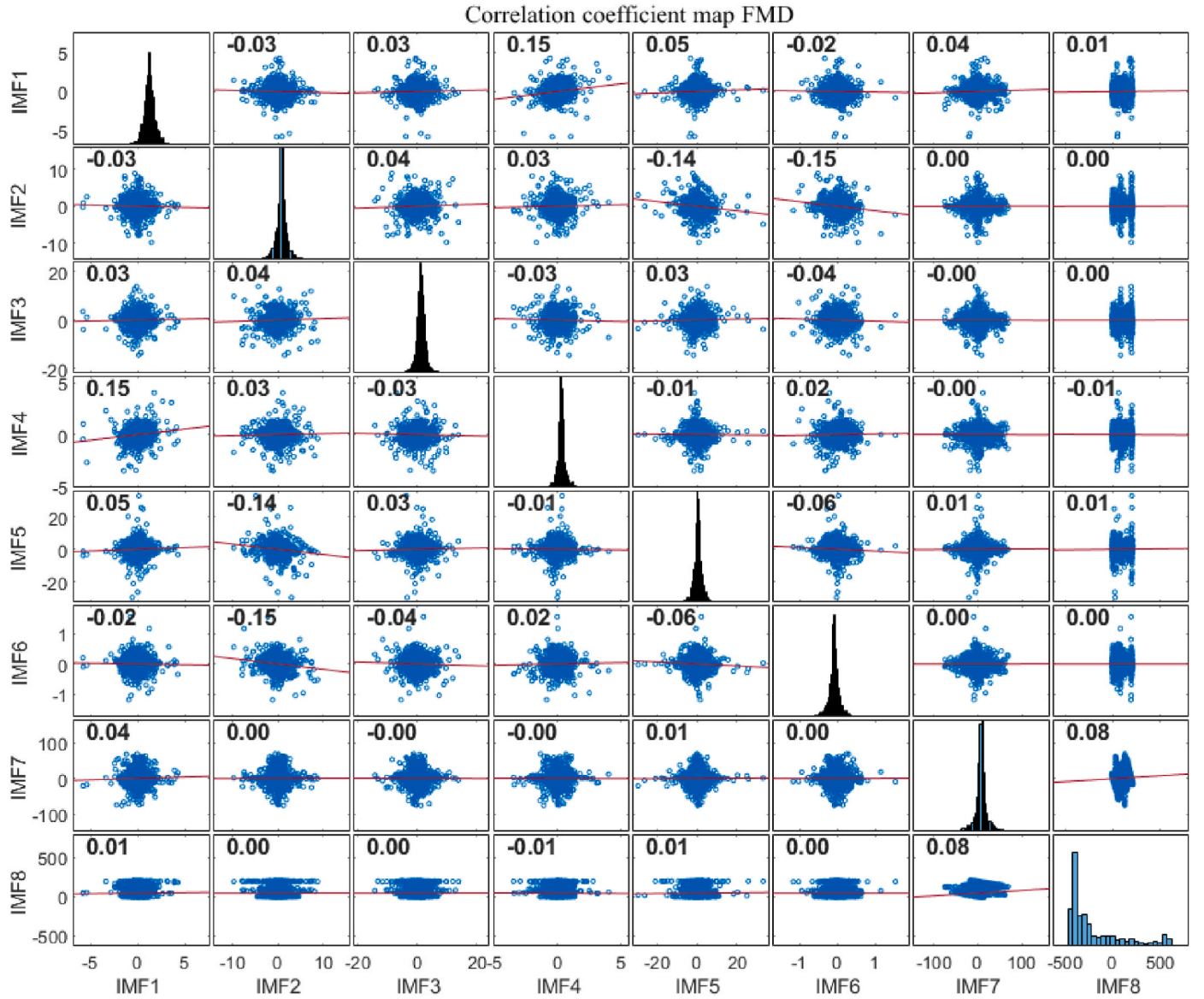


Fig. 13. Correlation spectrum of each component after CEEMDAN-FMD decomposition.

$$F_{GL}(x) = x\Phi(x) = \frac{x}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{t^2}{2}} dt \quad (16)$$

where, $\Phi(x)$ denotes the cumulative distribution function of the standard normal distribution.

2.4.2. Bidirectional gated recurrent unit

BiGRU leverages bidirectional propagation to capture forward and backward dependencies in sequence data. The model architecture is presented in Fig. 5.

Given an input sequence $X = (x_1, x_2, \dots, x_T)$, where x_T represents the feature vector at the t -th time step, and T denotes the length of the time series, the computational process of the GRU unit is described as follows.

$$\begin{cases} z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \\ r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \\ \tilde{h}_t = \tanh(W_h \cdot [r_t \cdot h_{t-1}, x_t] + b_h) \\ h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \end{cases} \quad (17)$$

where, z_t denotes the update gate signal, r_t represents the reset gate signal, $\tilde{h}_t$ is the candidate hidden state, and h_t is the updated hidden

state. x_t corresponds to the current input, while h_{t-1} denotes the output from the previous time step. σ indicates the sigmoid activation function; W_z , W_r , and W_h are the weight matrices; b_z , b_r , and b_h represent the bias terms.

2.4.3. Non-stationary self-attention mechanism

To address the non-stationarity in wind speed and power characterized by trends, cycles, and sudden changes, we incorporate a non-stationary self-attention mechanism [59]. This approach not only captures complex dynamic patterns but also mitigates the loss of non-stationary signals caused by normalization and decomposition, improving overall prediction accuracy.

For a given input time series x_t , the non-stationary self-attention mechanism is described as follows:

(1) Standardization

Calculate the mean μ_x and variance σ_x^2 of the input sequence, and perform standardization:

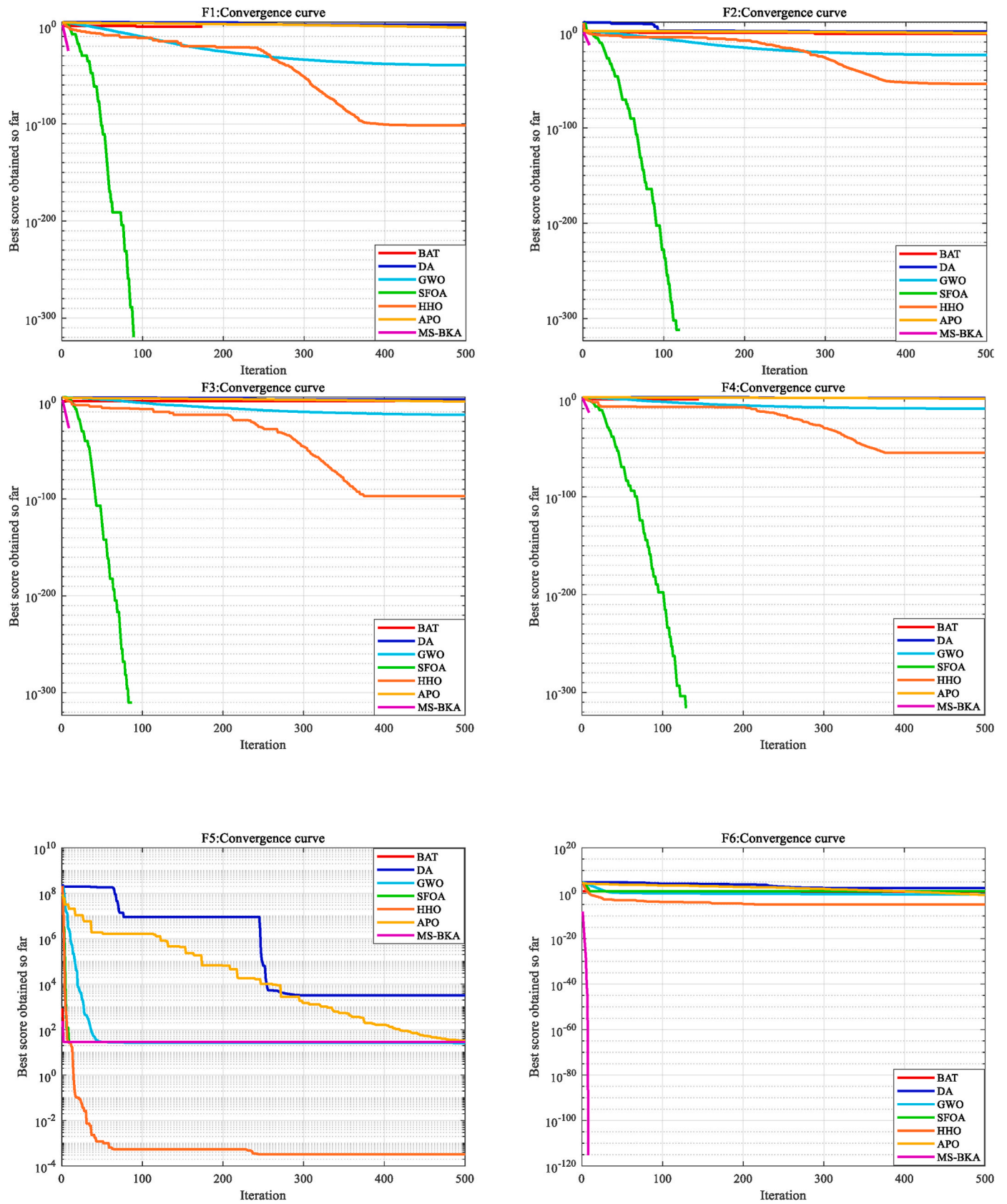


Fig. 14. Convergence experiments of different algorithms.

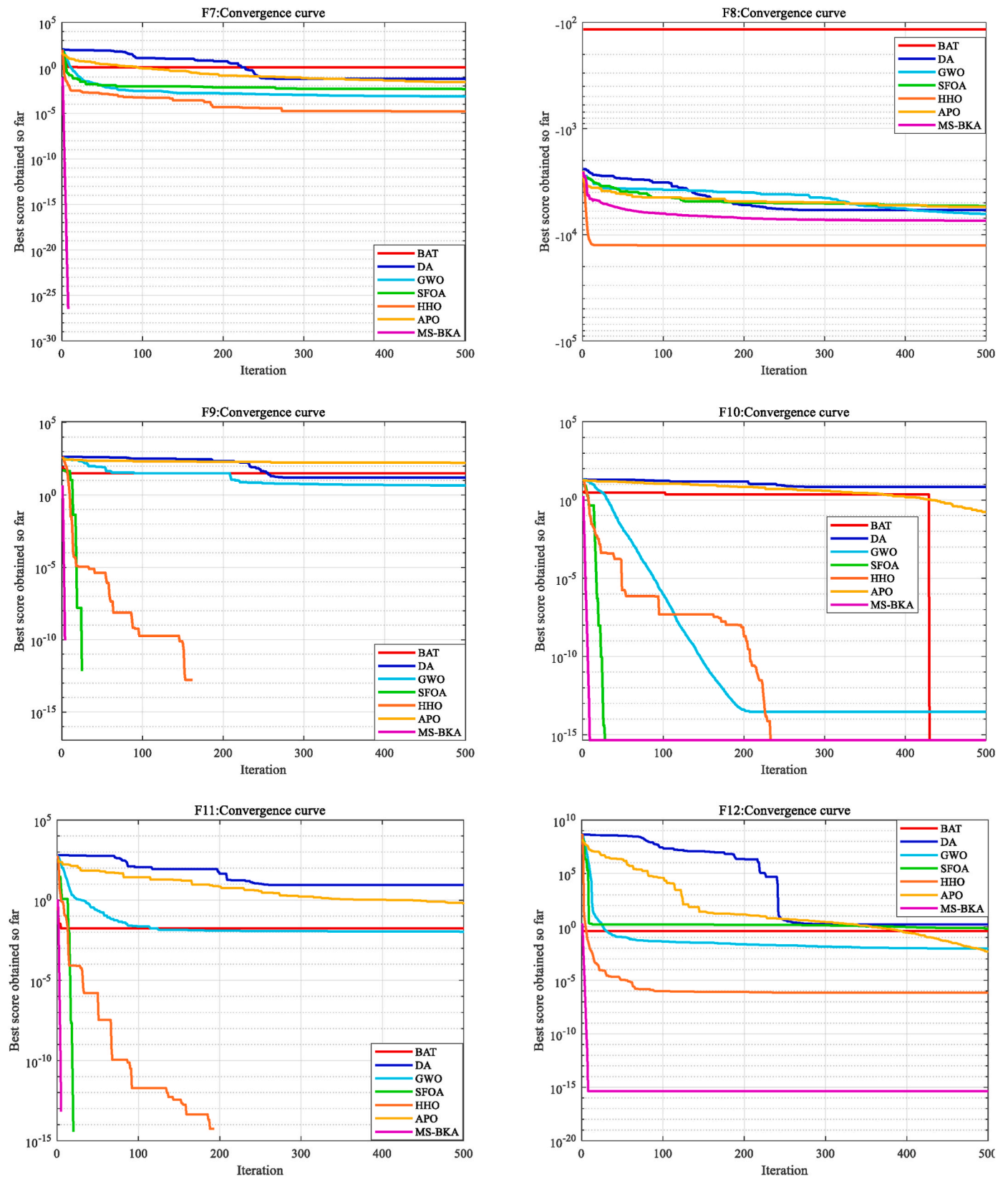


Fig. 14. (continued).

Table 1

Performance comparison results of different algorithms on test functions.

Function	Index	BAT	DA	GWO	SFOA	HHO	APO	MS-BKA
F1	Mean	2.47E-01	9.20E+01	1.10E-40	0.00E+00	4.15E-103	1.15E-01	0.00E+00
	Std	3.39E-01	5.88E+01	1.39E-40	0.00E+00	8.79E-103	4.51E-02	0.00E+00
	Rank	6	7	4	2	3	5	1
F2	Mean	7.80 E−01	1.12E+01	3.86E-24	0.00E+00	3.68E-55	2.48E-01	0.00E+00
	Std	7.40E-01	8.13E+00	1.80E-24	0.00E+00	7.29E-55	4.60E-02	0.00E+00
	Rank	6	7	4	2	3	5	1
F3	Mean	1.68E+00	1.65E+03	8.21E-13	0.00E+00	1.77E-98	4.34 E+00	0.00E+00
	Std	2.59E+00	1.08E+03	1.15E-12	0.00E+00	3.42E-98	1.86E+00	0.00E+00
	Rank	5	7	4	2	3	6	1
F4	Mean	0.00E+00	1.90E+01	2.74E-10	0.00E+00	1.87E-54	2.17 E+00	0.00E+00
	Std	0.00E+00	1.43 E+01	2.90E-10	0.00E+00	4.06E-54	5.79E-01	0.00E+00
	Rank	3	7	5	2	4	6	1
F5	Mean	6.07E+01	1.84E+05	2.65E+01	2.90E+01	2.73E-03	3.40E+01	2.85E+01
	Std	7.15E+01	3.83E+05	5.85E-01	3.42E-04	3.98E-03	2.38E+00	2.61E-01
	Rank	6	7	4	2	1	5	3
F6	Mean	5.76E+00	2.22E+02	2.92E-01	6.19E+00	1.09E-05	1.70E-01	−4.92E-120
	Std	3.22E+00	3.01E+02	2.78E-01	1.18E+00	1.50E-05	7.92E-02	1.55E-119
	Rank	6	7	4	5	2	3	1
F7	Mean	1.14E+00	6.21E-02	7.23E-04	4.47E-03	1.57E-05	2.81E-02	9.31E-166
	Std	1.52E+00	7.24E-02	1.79E-04	2.78E-03	1.43E-05	4.25E-03	0.00E+00
	Rank	7	6	3	4	2	5	1
F8	Mean	−1.17E+02	−5.85E+03	−6.35E+03	−5.35E+03	−1.27E+04	−5.45E+03	−7.37E-03
	Std	3.90E+03	7.29E+02	1.40 E+03	3.56E+02	5.43E-02	3.77E+02	2.23E+02
	Rank	7	5	6	3	1	4	2
F9	Mean	3.75E+01	1.12E+02	1.80E+00	0.00E+00	0.00E+00	1.69E+02	0.00E+00
	Std	7.73E+00	6.00E+01	2.47E+00	0.00E+00	0.00E+00	8.48E+00	0.00E+00
	Rank	5	6	4	2	3	7	1
F10	Mean	2.06E-01	7.22E+00	2.89E-14	4.44E-16	4.44E-16	6.17E+00	4.44E-16
	Std	3.29E-01	1.61E+00	0.00E+00	0.00E+00	0.00E+00	1.27E+00	0.00E+00
	Rank	4	7	5	2	3	6	1
F11	Mean	3.41E-02	5.04E+00	2.17E-03	0.00E+00	0.00E+00	3.12E-01	0.00E+00
	Std	4.19E-02	2.87E+00	4.85E-03	0.00E+00	0.00E+00	1.89E-01	0.00E+00
	Rank	5	7	4	2	3	6	1
F12	Mean	4.17E-01	1.73E+00	9.17E-03	6.49E-01	6.72E-07	4.67E-03	8.25E-18
	Std	2.11E-01	1.58E+00	7.32E-03	1.73E-01	7.31E-07	1.73E-03	2.41E-17
	Rank	6	7	4	5	2	3	1

Table 2

Comparison results of different optimization algorithms in model parameter optimization.

Method	Total time (s)	MAE	RMSE	R ²
WOA	1890.2	9.2828	12.1889	0.9562
NGO	1894.3	8.4977	11.5567	0.9606
HHO	2548.1	8.1892	11.0566	0.9673
GWO	2366.4	8.9752	12.1331	0.9566
BKA	2498.6	8.5147	12.8795	0.9584
IGWO	2583.7	8.0244	10.8514	0.9687
IHHO	2944.7	7.9560	10.5232	0.9706
ISSA	2751.6	7.6549	10.1317	0.9724
MS-BKA	2567.6	6.2521	8.3380	0.9801

$$\begin{cases} \mu_x = \frac{1}{L} \sum_{i=1}^L x_i \\ \sigma_x^2 = \frac{1}{L} \sum_{i=1}^L (x_i - \mu_x)^2 \\ z_i = \frac{1}{\sigma_x} \odot (x_i - \mu_x) \end{cases} \quad (18)$$

(2) Generation of non-stationary factors

Using the raw data statistics—the mean μ_x and variance σ_x^2 —compensation factors τ and Δ are calculated through a multilayer perceptron (MLP).

$$\begin{cases} \log \tau = \text{MLP}(\sigma_x, z) \\ \Delta = \text{MLP}(\mu_x, z) \end{cases} \quad (19)$$

where, τ represents the time scaling compensation factor, and Δ denotes the offset compensation term.

(3) Non-stationary attention

The aforementioned time scaling compensation factor τ and offset compensation term Δ are incorporated into the traditional attention scoring function, resulting in the following final attention calculation expression:

$$SA_{N-s}(Q, K, V, \tau, \Delta) = \text{Soft max} \left(\frac{\tau QK^T + 1\Delta^T}{\sqrt{d_k}} \right) V \quad (20)$$

The non-stationary attention mechanism enables the model to

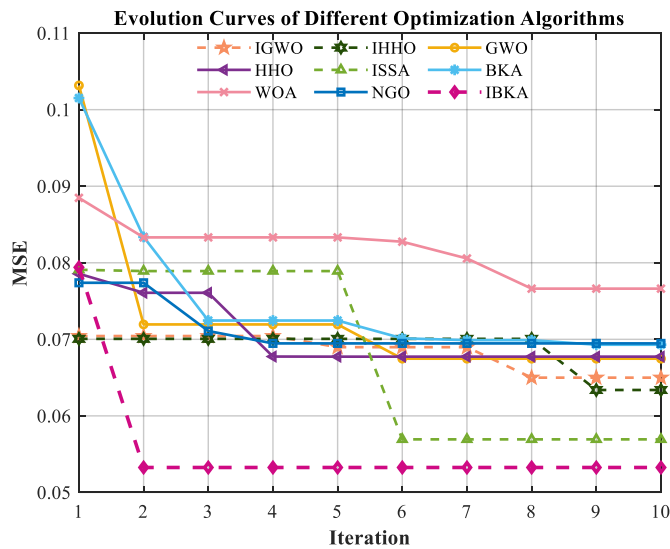


Fig. 15. Comparison of evolution curves of different algorithms.

effectively compensate for information loss caused by normalization and data decomposition operations while capturing long-term dependencies, thereby enhancing its capability to model the non-stationary

Table 3
Model parameter optimization results.

Parameter Index	Hyperparameter name	Search Range	Best Search Results
h ₁	The FMD filter parameter F	[10,50]	45
h ₂	The FMD modal component parameter M	[3,12]	6
h ₃	The number of hidden units in the BiGRU layer.	[10,128]	50
h ₄	The NumHeads parameter of multi-head attention.	[1,8]	5
h ₅	The KeyDimension parameter of multi-head attention.	[10,100]	30
h ₆	InitialLearnRate	[10 ⁻¹ -10 ⁻⁶]	10 ⁻²
h ₇	MiniBatchsize	[8,16,32, ...,521,1024]	32
h ₈	MaxEpochs	[20-200]	150
h ₉	Dropout	[0.1-0.5]	0.2

Table 4
Comparison results of different decomposition methods.

Model	Decomposition strategy	Decomposition Method	VMD	CEEMDAN	FMD	EMD	Decomposition time (s)	Evaluation indicators	
								MAE	R ²
MSTCN-BiGRU-NSA	Single decomposition	-	✓	-	-	-	0.2433	11.4421	0.9234
			-	✓	-	-	31.2240	9.1124	0.9394
			-	-	✓	-	1.2451	9.1524	0.9424
			-	-	-	✓	0.0751	10.2217	0.9271
	Two-stage decomposition	VMD	✓	-	-	-	0.4234	10.5202	0.9357
			-	✓	-	-	32.6624	8.6637	0.9511
			-	-	✓	-	1.4437	8.0234	0.9589
			-	-	-	✓	0.3211	9.2351	0.9502
		EMD	✓	-	-	-	0.3561	9.3575	0.9587
			-	✓	-	-	35.2248	8.5144	0.9469
			-	-	✓	-	1.2514	8.2791	0.9526
			-	-	-	✓	0.1422	9.0576	0.9315
		FMD	✓	-	-	-	1.3549	8.2241	0.9405
			-	✓	-	-	34.7181	8.3614	0.9414
			-	-	✓	-	2.6347	8.0244	0.9596
			-	-	-	✓	1.2454	8.1127	0.9524
		CEEMDAN	✓	-	-	-	33.7560	8.0076	0.9652
			-	✓	-	-	63.1270	8.3511	0.9569
			-	-	✓	-	28.4622	6.2521	0.9801
			-	-	-	✓	36.5547	8.0911	0.9607

characteristics of wind power time series.

2.4.4. Improved non-crossed quantile regression

(1) Quantile loss function

The classical quantile regression (CQR) method estimates conditional quantiles by minimizing the Pinball loss function, and has been widely applied in uncertainty modeling and interval forecasting tasks. The loss function assigns different weights to the positive and negative parts of the prediction errors, and is expressed as follows:

$$L_q^{(\alpha)}(y_n, \hat{q}_n^{(\alpha)}) = \begin{cases} \alpha(y_n - \hat{q}_n^{(\alpha)}), & y_n \geq \hat{q}_n^{(\alpha)} \\ (1 - \alpha)(\hat{q}_n^{(\alpha)} - y_n), & y_n < \hat{q}_n^{(\alpha)} \end{cases} \quad (21)$$

Although the Pinball loss function has demonstrated strong performance in quantile estimation, its non-differentiability can cause oscillations near zero when using gradient-based optimization algorithms, thereby affecting the stability of model convergence. To address this issue, a fully differentiable Huber-norm Pinball (H-Pinball) loss function [54] is adopted as a replacement for the conventional Pinball loss. The mathematical formulation of the H-Pinball loss function is defined as follows:

$$\rho(y_n, \hat{q}_n^{(\alpha)}, \eta) = \begin{cases} |y_n - \hat{q}_n^{(\alpha)}| - \eta/2, & |y_n - \hat{q}_n^{(\alpha)}| \geq \eta \\ (y_n - \hat{q}_n^{(\alpha)})^2 / (2\eta), & |y_n - \hat{q}_n^{(\alpha)}| < \eta \end{cases} \quad (22)$$

$$L_H^{(\alpha)}(y_n, \hat{q}_n^{(\alpha)}) = \begin{cases} \alpha \rho(y_n, \hat{q}_n^{(\alpha)}, \eta), & y_n - \hat{q}_n^{(\alpha)} \geq 0 \\ (1 - \alpha) \rho(y_n, \hat{q}_n^{(\alpha)}, \eta), & y_n - \hat{q}_n^{(\alpha)} < 0 \end{cases} \quad (23)$$

where, η is a positive threshold value. The comparison between the H-Pinball and traditional Pinball loss functions under different η values is illustrated in Fig. 6. As shown in Fig. 6, the H-Pinball loss is differentiable everywhere.

Table 5
Comparison results of different hybrid strategies.

Model	MAE	RMSE	R ²
VMD-ATT-LSTM	13.0144	16.2240	0.9175
VMD-SSA-SVR	10.9389	13.1677	0.9204
Proposed	6.2521	7.3024	0.9801

(2) Improved non-cross quantile regression method

onventional quantile regression often suffers from the quantile crossing problem, in which higher quantiles produce smaller predicted values than lower ones. This violates the inherent monotonicity property of quantiles, resulting in unreliable probabilistic forecasts and potential decision-making risks, particularly in wind power prediction scenarios. To address this issue, studies [54,60] introduced non-crossing quantile regression approaches. Building upon this idea, the present study proposes an enhanced non-crossing quantile regression method, as illustrated in Fig. 7.

As shown in Fig. 7, the improved INCQR framework avoids directly producing quantile predictions. Instead, it first generates a series of non-negative quantile elements $\{\hat{m}_1, \hat{m}_2, \dots, \hat{m}_n\}$ as intermediate representations. The final quantiles are derived through a negative exponential summation mapping, formulated as:

$$\hat{q}^{(a_i)} = \exp\left(-\sum_{j=1}^{r-i+1} \hat{m}_j\right) \quad (24)$$

The predicted quantiles are strictly monotonic, as expressed by $1 > \hat{q}^{(a_r)} > \hat{q}^{(a_{r-1})} > \hat{q}^{(a_{r-2})} > \dots > \hat{q}^{(a_1)} > 0$. Unlike the NCQR approach in Ref. [54], the proposed method introduces an adaptive Softplus activation function to enforce non-negativity on the quantile elements produced by the model. The formulation of the adaptive Softplus function is given as:

$$f(x) = \frac{1}{\beta} \log(1 + e^{\beta x}) \quad (25)$$

where, β is a trainable parameter. Compared with traditional activation functions such as ReLU and Softplus used for non-negativity constraints, the adaptive Softplus activation function can automatically adjust its "activation shape" according to the training data, thereby enhancing the model's adaptability. The Softplus activation functions under different values of β are illustrated in Fig. 8.

Table 6
Comparison of model structure parameter settings.

Model	Network Architecture Summary	Parameter settings
CNN- BiGRU	CNN($3 \times 1, 16$) + BN + ReLU + MaxPool(3×3) CNN($3 \times 1, 32$) + BN + ReLU + MaxPool(2×2) BiGRU(35) + Concat + FC(1) Regression	InitialLearnRate = 10^{-3} MiniBatchsize = 64 MaxEpochs = 150 Dropout = 0.2
TCN- BiGRU	[TCN($5 \times 1, 64$, dilations [1]) $\times 2$] $\times 2$ LayerNorm + ReLU + Dropout BiGRU(35) + Concat + FC(1) Regression	InitialLearnRate = 1×10^{-2} MiniBatchsize = 64 MaxEpochs = 150 Dropout = 0.1
LSTM- BiGRU	LSTM(50) ReLU + Dropout BiGRU(35) + Concat + FC(1) Regression	InitialLearnRate = 1×10^{-2} MiniBatchsize = 128 MaxEpochs = 150 Dropout = 0.1
Transformer- BiGRU	$2 \times$ Multi-Head Self-Attention (4 heads, 128 key channels) BiGRU(35) + FC(1) Regression	InitialLearnRate = 1×10^{-3} MiniBatchsize = 32 MaxEpochs = 100 Dropout = 0.2
MSTCN-BiGRU	[TCN($5 \times 1, 64$, dilations [1–3]) $\times 3$] $\times 2$ LayerNorm + ReLU + Dropout + Concat BiGRU(35) + Concat + FC(1) Regression	InitialLearnRate = 1×10^{-3} MiniBatchsize = 64 MaxEpochs = 150 Dropout = 0.2

2.5. Model hyperparameter optimization

2.5.1. Black-winged kite algorithm

The BKA algorithm [61] simulates the hunting strategies and migratory behaviors of black kites to solve complex optimization problems. The mathematical formulation of the algorithm is presented as follows.

Algorithm 3

Black-winged kite algorithm

Input: The population size pop , the maximum number of iterations T , and the variable dimension dim .
Output: The best quasi-optimal solution obtained by BKA for a given optimization problem.

- Initialization phase**
- Initialization of the position of Black-winged kites and evaluation of the objective function.
- Calculate the fitness value of each Black-winged kite.
- while** ($t < T$) **do**
- Attacking behavior**
- if** $p < r$
- $y_{i+1}^{j,j} = y_i^{j,j} + n(1 + \sin(r)) \times y_i^{j,j}$
- else if do**
- $y_{i+1}^{j,j} = y_i^{j,j} + n \times (2r - 1) \times y_i^{j,j}$
- end if**
- Migration behavior**
- if** $F_i < F_{ti}$ **do**
- $y_{i+1}^{j,j} = y_i^{j,j} + C(0,1) \times (y_i^{j,j} - L_i^j)$
- else if do**
- $y_{i+1}^{j,j} = y_i^{j,j} + C(0,1) \times (L_i^j - m \times y_i^{j,j})$
- Select the best individual**
- if** $y_{i+1}^{j,j} < L_j$
- $X_{best} = y_{i+1}^{j,j}, F_{best} = f(y_{i+1}^{j,j})$
- else if do**
- $X_{best} = L_j, F_{best} = f(L_j)$
- end if**
- end while**
- Return** X_{best} and F_{best}

2.5.2. Algorithm improvement

As iterations progress, the BKA algorithm's population tends to cluster in a particular region, raising the likelihood of becoming trapped in local optima, especially when optimizing complex multimodal functions. To overcome this problem, an opposition-based learning (OBL) strategy is adopted in the later iteration stage to improve convergence stability and exploration efficiency. By reflecting the solution search region across a symmetric counterpart, the method promotes population diversity and enhances the algorithm's capacity to escape local optima. This reverse learning process is shown in Fig. 9.

(1) Mutation OBL strategy

The mutation OBL strategy introduces a mutation factor p on top of the OBL approach. This factor determines the direction of the learning strategy, enabling adaptive selection based on the value of p . The mathematical formulation of the mutation-based opposition learning strategy is presented as follows.

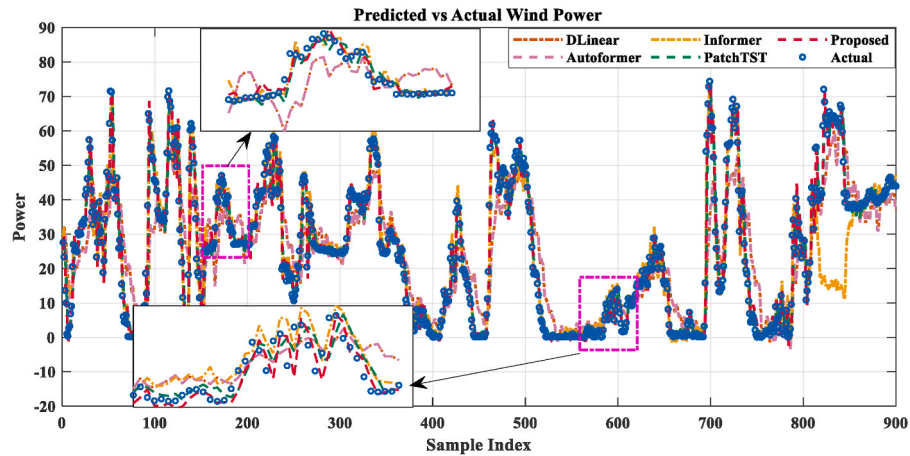
$$X_{new}^t = \begin{cases} lb_i + R(ub_i - X_i^t) & , b \leq p \\ lb_i + (ub_i - X_i^t) & , b > p \end{cases} \quad (26)$$

where, R follows a uniform distribution over the interval $(0, 1)$; b is a

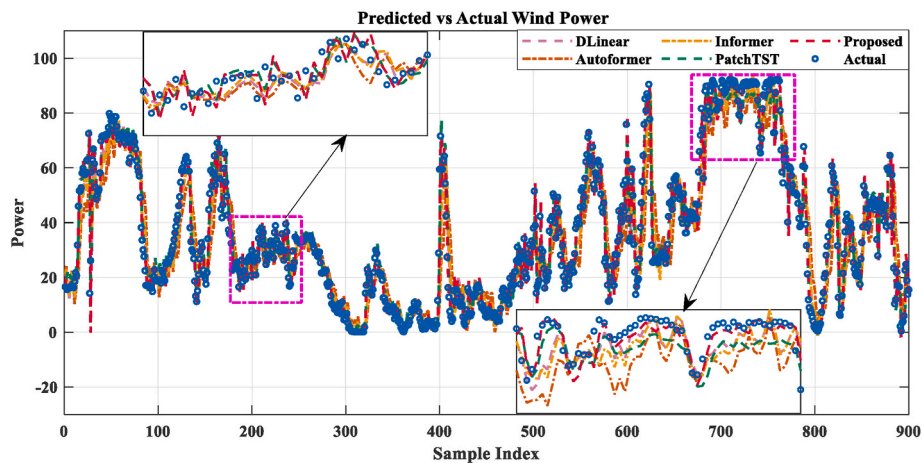
Table 7

Comparison of model performance under different architectures and attention mechanisms.

Model	Experimental conditions	Attention		Model Complexity		Evaluation indicators			
		SA	NSA	Trainable parameters (M)	FLOPs (G)	Train Time(s)	Test Time (s)	MAE	R ²
CNN- BiGRU	M1	–	–	0.0251	0.5510×10^{-3}	1425.6	0.0880	22.3386	0.8357
	M2	✓	–	0.0262	0.5541×10^{-3}	1627.3	0.3511	17.6208	0.8978
	M3	–	✓	0.0322	0.5581×10^{-3}	1833.4	0.3782	14.3591	0.9166
TCN- BiGRU	M1	–	–	0.1420	0.3680×10^{-1}	1214.3	0.0751	11.3399	0.9355
	M2	✓	–	0.1560	0.3690×10^{-1}	1277.8	0.2200	8.3429	0.9514
	M3	–	✓	0.1591	0.3712×10^{-1}	1294.6	0.1522	7.8567	0.9668
LSTM- BiGRU	M1	–	–	0.0372	0.7403×10^{-3}	1007.1	0.0813	15.6744	0.8722
	M2	✓	–	0.0375	0.7441×10^{-3}	1144.6	0.1470	13.9967	0.9154
	M3	–	✓	0.0377	0.7490×10^{-3}	1158.6	0.1480	12.0211	0.9381
Transformer- BiGRU	M2	✓	–	0.4322	0.8542×10^{-3}	1345.5	0.1582	9.2207	0.9432
	M3	–	✓	0.4401	0.8591×10^{-3}	1366.2	0.1641	8.0081	0.9622
MSTCN-BiGRU	M1	–	–	0.3492	0.1100	1869.4	0.2560	8.0284	0.9589
	M2	✓	–	0.3590	0.1110	2510.6	0.3571	7.3511	0.9702
	M3	–	✓	0.3654	0.1111	2567.6	0.4891	6.2521	0.9801

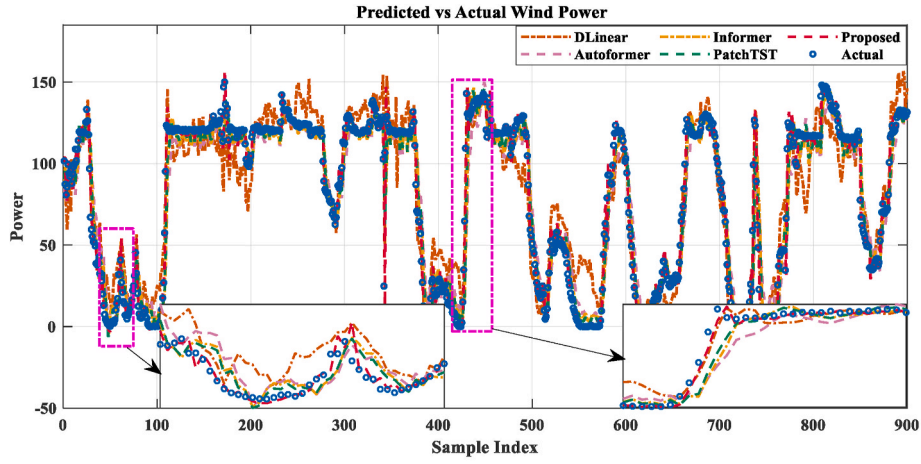


(a) A-2023 Test Results

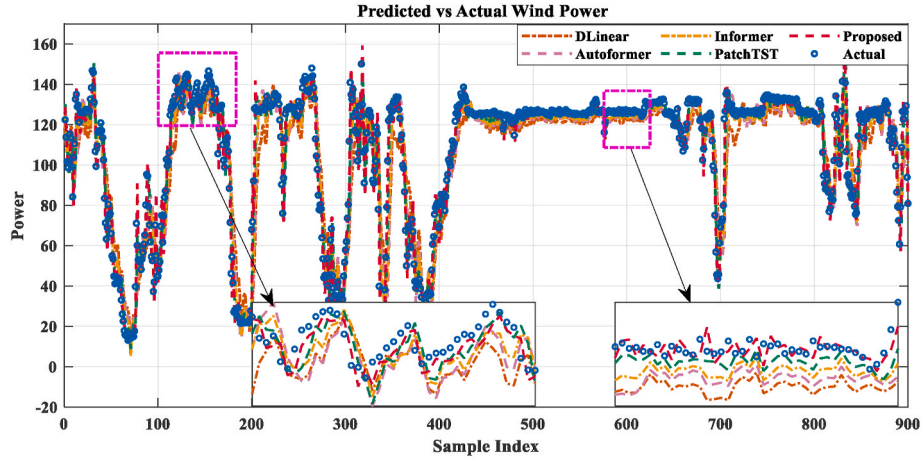


(b) A-2024 Test Results

Fig. 16. Comparison of prediction performance of different models.



(c) B-2023 Test Results



(d) B-2024 Test Results

Fig. 16. (continued).

random number in the range $[0, 1]$; and p denotes the mutation probability.

The algorithm uses a greedy strategy to keep the best solutions for the next generation.

$$X_i^{t+1} = \begin{cases} X_{newi}^t, & f(X_{newi}^t) \leq f(X_i^{t+1}) \\ X_i^{t+1}, & f(X_{newi}^t) > f(X_i^{t+1}) \end{cases} \quad (27)$$

A dynamic comparison-based strategy facilitates the algorithm's evolution toward the target position, leading to faster convergence.

(2) Dynamic perturbation mechanism leveraging the properties of the Cauchy distribution

During the attack phase of the BKA algorithm, two update strategies are utilized: one employs a nonlinear transformation based on the sine function, while the other uses an update mechanism governed by a decay factor n . The mathematical expressions are given below:

$$y_{t+1}^{ij} = \begin{cases} y_t^{ij} + n(1 + \sin(r)) \times y_t^{ij} & p < r \\ y_t^{ij} + n \times (2r - 1) \times y_t^{ij} & p \geq r \end{cases} \quad (28)$$

where, n is a dynamic decay factor that decreases as the number of iterations increases. Its calculation formula is as follows:

$$n = 0.05 \times e^{-2\left(\frac{t}{T}\right)^2} \quad (29)$$

In the later stages of the algorithm, the exponential decay of parameter n causes new solutions to cluster near the current ones, limiting the global search capability and making the quality of new solutions highly dependent on the current solutions. This paper introduces a dynamic perturbation mechanism based on the Cauchy distribution, leveraging its heavy-tailed property to design a jump-based solution update strategy that enhances the algorithm's global exploration capability. The mathematical formulation of the Cauchy distribution is as follows:

$$CauchyTnverCumDist(dim)_x = a + b \cdot \tan(\pi(p - 0.5)) \quad (30)$$

where, p is a random variable, a is the location parameter, and b is the scale parameter.

The improved position update formula is:

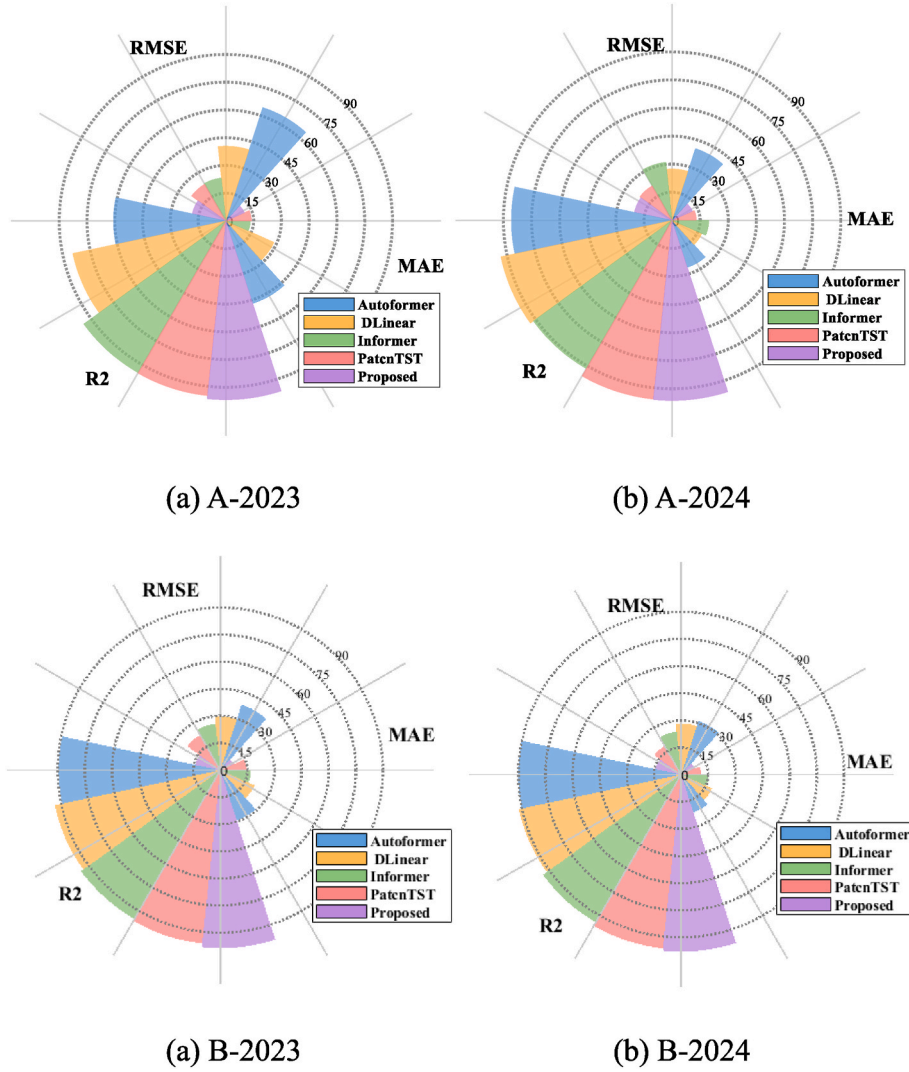


Fig. 17. Comprehensive performance comparison of different models.

$$y_{t+1}^{ij} = \begin{cases} y_t^{ij} + n(1 + \sin(r)) \times y_t^{ij} & p < r \\ y_t^{ij} + n \times (2r - 1) \times y_t^{ij} \times \text{CauchyTnverCumDist}(\dim) & p \geq r \end{cases} \quad (31)$$

The computational complexity of the BKA algorithm consists of two main stages: the initialization stage with a time complexity of $O(M)$, and the fitness evaluation stage with a time complexity of $O(T \times M) + O(T \times M \times D)$, where M denotes the population size, T represents the maximum number of iterations, and D is the dimensionality of the solution space. The overall computational complexity of the BKA algorithm is $O(M \times (T + T \times D + 1))$, and after eliminating constant and lower-order terms with negligible influence, the final time complexity can be expressed as $O(M \times T \times D)$. For the improved BKA algorithm, the initialization stage has a time complexity of $O(M)$, the dynamic Cauchy perturbation strategy corresponds to a complexity of $O(D)$, and the mutation-based opposition learning contributes a complexity of $O(M \times T \times D)$. Therefore, the total time complexity of the improved BKA algorithm is $O(M \times T \times D) + O(M \times T \times D) + O(D)$, and by retaining only the dominant terms, its final time complexity is $O(M \times T \times D)$. The incorporation of these hybrid improvement strategies does not significantly increase the algorithm's overall computational cost, making the proposed dual-strategy enhancement an acceptable and effective means of improving convergence performance.

3. Experimental part

3.1. Experimental environment and data preprocessing

3.1.1. Experimental environment

The experiments were implemented using Python 3.10, PyTorch 2.2.1, and CUDA 12.1 deep learning frameworks. The computational platform was equipped with an Intel Core i5-12600KF CPU and an NVIDIA GeForce RTX 4060 Ti GPU.

3.1.2. Dataset description

The dataset used in this study consists of monitoring data from two wind farms located in northern China (Wind farm A and Wind farm B), as illustrated in Fig. 10. The input features include wind speed and wind direction measurements at heights of 10 m, 30 m, 50 m, and 70 m from the meteorological mast, wind speed and wind direction at the turbine hub, as well as environmental variables such as temperature, atmospheric pressure, and humidity. The dataset contains two years of monitoring data (2023–2024) from both wind farms, with a sampling interval of 15 min.

3.1.3. Cascaded seasonal data isolation strategy

A multi-feature input scheme is employed to forecast wind turbine power, using a dataset containing 13 meteorological features and one

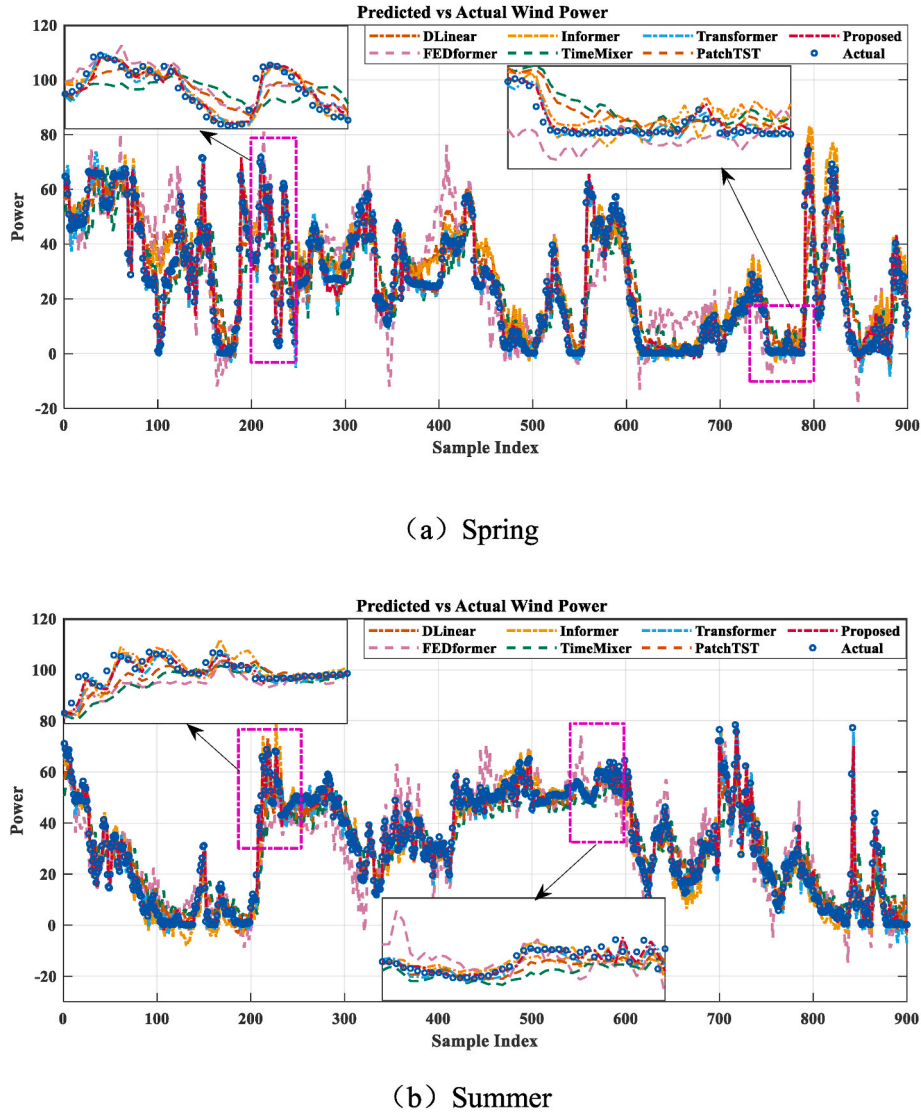


Fig. 18. Wind power prediction results of different models for different seasons (Wind Farm A, 2023).

power output variable. To meet model input requirements, the data are segmented and preprocessed accordingly. To prevent information leakage common issue in time series prediction where future data are unintentionally introduced during training cascaded seasonal data isolation strategy is applied before sample generation. The yearly data are split into spring, summer, autumn, and winter subsets, each independently divided into training, validation, and testing sets, with no temporal overlap between seasons. A sliding window method is then used within each seasonal training set to generate input samples, ensuring that each sample at time t relies solely on historical and current data. The overall multivariate dataset construction procedure is shown in Fig. 11.

Fig. 11 presents the construction process of multi-feature input samples. Specifically, the model input consists of samples from time steps t and $t+1$, while the power value at time $t+2$ is used as the prediction target. This process is iteratively repeated to construct a dataset with multi-feature inputs.

3.1.4. Two-stage modal decoupling

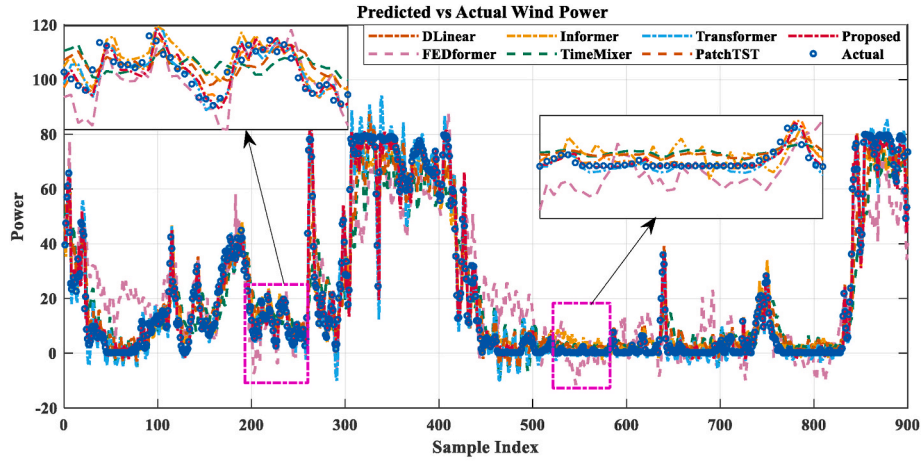
To enhance the modelability of non-stationary signals, a two-stage modal decoupling strategy is proposed, combining CEEMDAN for extracting dominant components with FMD for further refinement and

feature enhancement. This approach facilitates effective signal decomposition and modeling, as shown in Fig. 12.

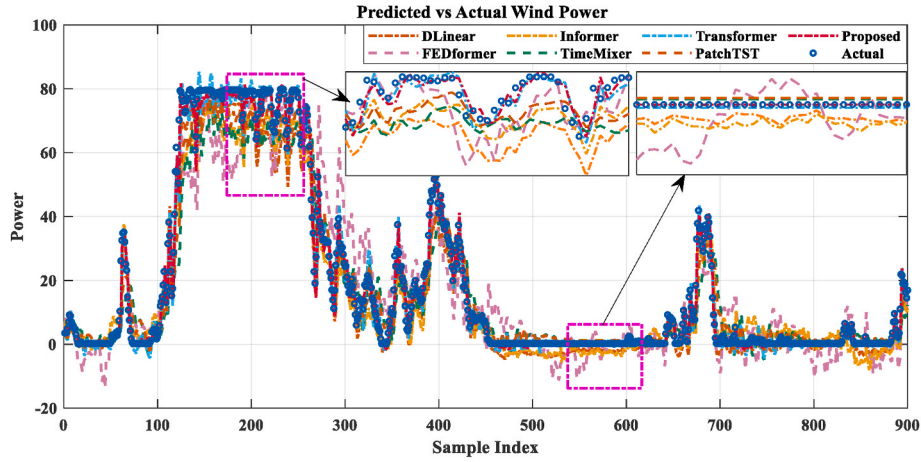
Fig. 12 illustrates that the original wind turbine power signal exhibits no clear pattern, making accurate prediction challenging for traditional models. To address the complex fluctuations caused by seasonality, operational conditions, and random disturbances, a two-stage CEEMDAN-FMD modal decoupling approach is employed. First, CEEMDAN decomposes the signal into 12 intrinsic mode functions. Next, sample entropy is calculated, and the IMFs are clustered into high-frequency, medium-frequency, and low-frequency groups using K-means clustering. Finally, the high-frequency group undergoes further decomposition via FMD, yielding six subcomponents. The medium- and low-frequency groups, along with the FMD-derived components, are integrated as inputs for the prediction model.

To further analyze the correlation characteristics among different modes, a correlation map based on the Pearson correlation coefficient was generated, as shown in Fig. 13.

In the correlation maps shown in Fig. 13, each subplot illustrates the joint distribution between two IMF components. The blue scatter points represent the value pairs of the two components at each time instance, while the red line denotes their linear regression trend. As observed from Fig. 13, the scatter distributions of most IMF pairs appear to be rather



(c) Autumn



(d) Winter

Fig. 18. (continued).

dispersed and exhibit no obvious linear patterns. Moreover, the trend lines tend to be nearly horizontal, indicating weak linear correlations between the components. This suggests that the CEEMDAN-FMD method effectively achieves modal decoupling and mitigates information redundancy across modes. Specifically, the correlation coefficient between IMF1 and IMF4 is 0.15, and that between IMF2 and IMF6 is -0.15 , indicating a certain degree of positive or negative linear relationship, albeit with low magnitude, which does not compromise the overall independence of the components. Experiments further demonstrate that the modes are independent and capable of extracting structurally distinct and physically meaningful feature components from the raw signals, providing a more decoupled and non-interfering data foundation for subsequent feature extraction and modeling.

3.2. Evaluation metrics

(1) Mean Squared Error (MSE)

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - y_i^p)^2 \quad (32)$$

where, n denotes the number of samples, y_i represents the actual values,

and y_i^p corresponds to the predicted values generated by the model.

(2) Root Mean Squared Error (RMSE)

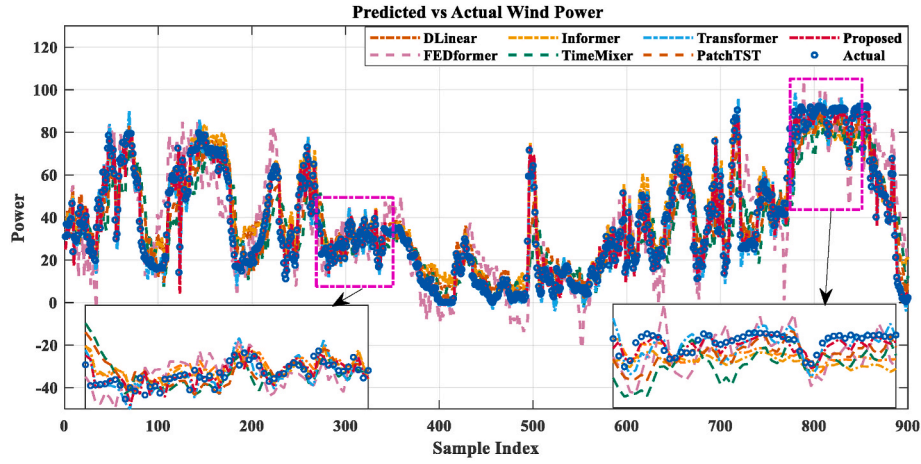
$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y_i^p)^2} \quad (33)$$

(3) Mean Absolute Error (MAE)

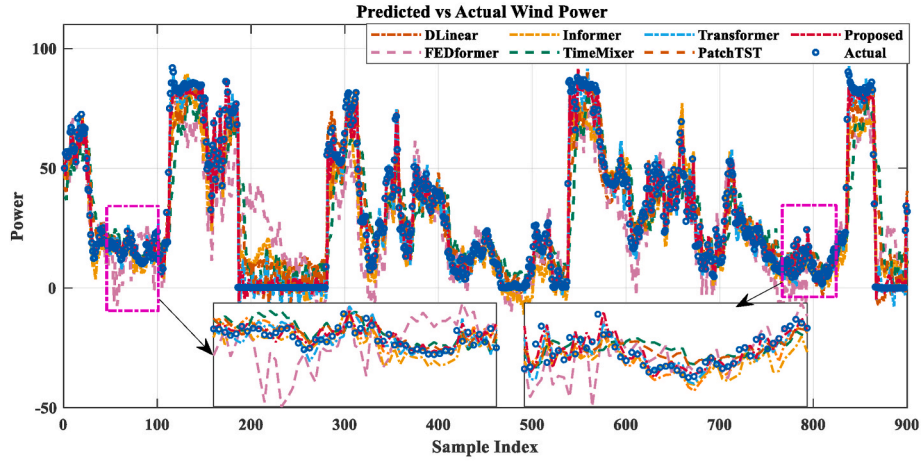
$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - y_i^p| \quad (34)$$

(4) Mean Absolute Percentage Error (MAPE)

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - y_i^p}{y_i} \right| \times 100 \quad (35)$$



(a) Spring



(b) Summer

Fig. 19. Wind power prediction results of different models for different seasons (Wind Farm A, 2024).

(5) Coefficient of determination (R^2)

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - y_i^p)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (36)$$

where, $\bar{y}$ denotes the mean of the true values.

(6) Prediction Interval Normalized Average Width (PINAW)

$$PINAW = \frac{\sum_{i=1}^N (U_i - L_i)}{N \cdot (\text{Max}(\text{real}) - \text{Min}(\text{real}))} \quad (37)$$

where, N represents the total number of actual samples, while $\text{Max}(\text{real})$ and $\text{Min}(\text{real})$ denote the lower and upper bounds of the observed values, respectively. For the i -th predicted sample, U_i and L_i correspond to the lower and upper confidence limits of its prediction interval.

(7) Prediction Interval Coverage Probability (PICP)

$$PICP = \frac{\sum_{j=1}^N \Pi(L_j \leq \text{real}_j \leq U_j)}{N} \quad (38)$$

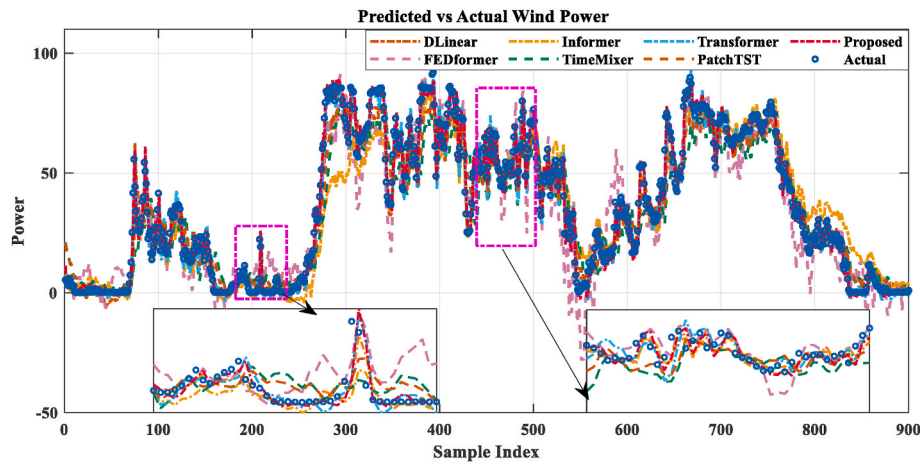
where, $\Pi(\cdot)$ denotes the indicator function, which equals 1 when the condition is true and 0 otherwise. PINAW measures the compactness of the prediction interval, with smaller values indicating higher precision. PICP reflects the coverage probability of the interval containing the true values, where values closer to the confidence level indicate greater reliability.

3.3. MSTCN-BiGRU-NSA model performance verification

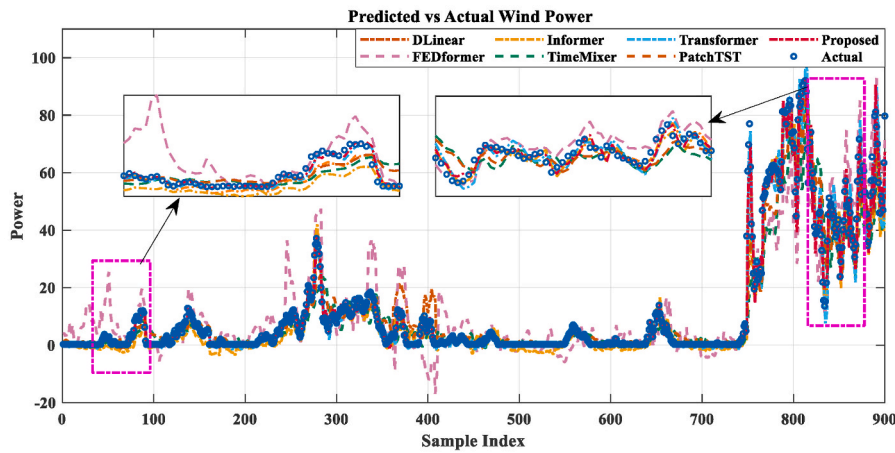
3.3.1. BKA algorithm performance verification

(1) Performance verification on standard test functions

A series of comparative experiments was conducted to evaluate the proposed MS-BKA algorithm against a suite of established metaheuristic methods, including bat algorithm (BAT), dragonfly algorithm (DA), grey wolf optimizer (GWO), starfish optimization algorithm (SFOA), harris



(c) Autumn



(d) Winter

Fig. 19. (continued).

hawks optimization (HHO), and arctic puffin optimization (APO), using selected standard benchmark functions. To maintain the integrity of experimental design, all algorithms were tested under identical settings: a population size of 100, 500 iterations, and a fixed dimensionality of 30. Each algorithm underwent 10 independent runs to minimize stochastic variation. The experimental results are summarized in Fig. 14.

Fig. 14 presents a comparison of the convergence curves of various optimization algorithms across 12 test functions. As shown in Fig. 14, the proposed MS-BKA algorithm demonstrates superior convergence speed and higher accuracy than the compared algorithms on the unimodal benchmark functions F1, F2, F3, F4, F6, and F7. For the unimodal test function F5, its convergence performance ranks only behind the HHO and SFOA algorithms. Regarding the convergence validation on multimodal test functions, the improved MS-BKA algorithm demonstrates superior convergence on F9, F11, and F12 compared to other algorithms. On test function F10, although the final convergence values of the SFOA, HHO, and BAT algorithms are comparable to those of MS-BKA, the MS-BKA algorithm achieves rapid convergence during the early iteration phase, exhibiting higher convergence efficiency than the others. For test function F8, the convergence performance of MS-BKA is second only to that of HHO. Results suggest that the MS-BKA algorithm maintains a competitive edge in convergence behavior on the majority of test functions, thereby validating the

practical impact of the proposed modifications.

The results obtained from testing the enhanced MS-BKA algorithm and several other algorithms on 12 standard benchmark functions are compiled in Table 1. As shown in Table 1, the improved MS-BKA algorithm achieved the best results in 10 out of the 12 test functions, except for F5 and F8, accounting for 83.3 % of the optimal outcomes. The MS-BKA algorithm shows relatively uniform performance across the tested functions when compared to other approaches, with the evaluated algorithms maintaining robust convergence characteristics.

(2) Performance verification based on different optimization algorithms on the model

To verify the effectiveness of the proposed improved MS-BKA optimization algorithm in practical hyperparameter tuning, a unified comparative experiment is conducted under a fixed MSTCN-BiGRU-NSA architecture. The model hyperparameters are defined as joint search variables, and several metaheuristic algorithms-including the whale optimization algorithm (WOA), northern goshawk optimization (NGO), HHO, GWO, as well as their improved variants (IGWO [62], IHHO [63], and ISSA [64])-are employed for parameter optimization alongside the proposed MS-BKA. The mean absolute percentage error (MAPE) is used as the objective function, and the generalization performance of each

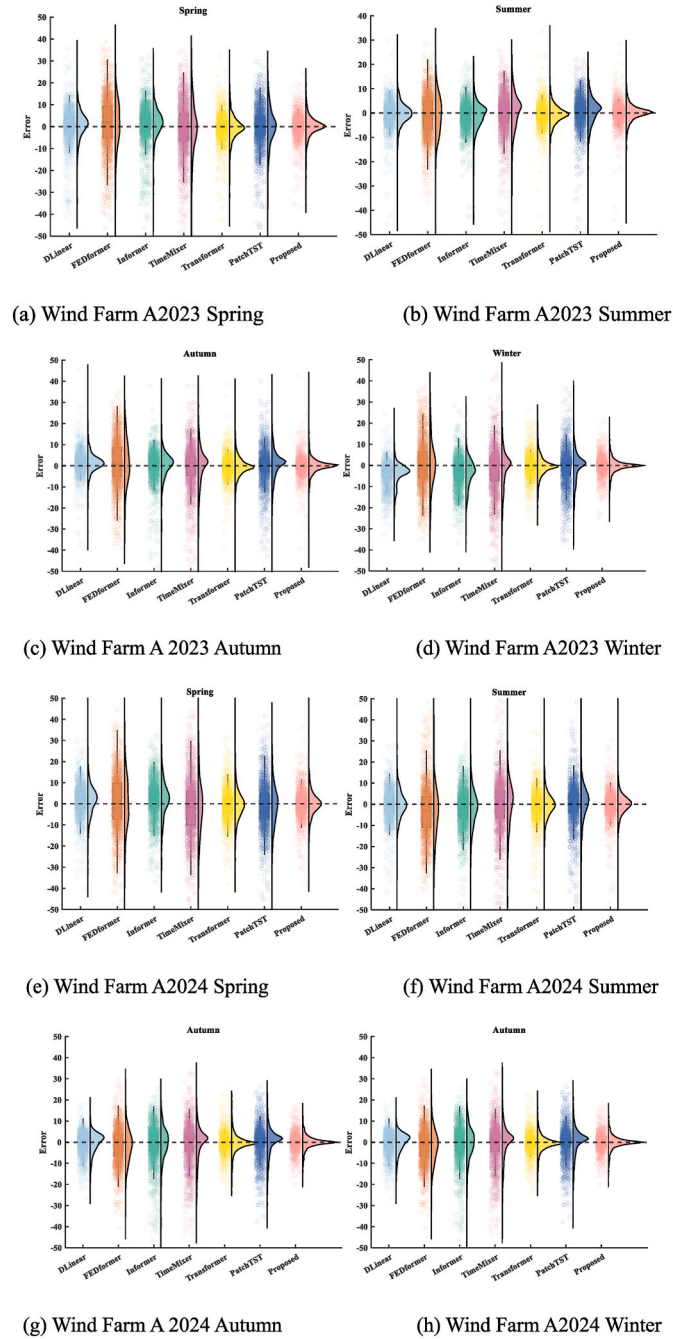


Fig. 20. Cloud and rain maps corresponding to different models in different seasons (Wind Farm A 2023–2024).

algorithm is evaluated based on the prediction metrics obtained from the test set. In addition, the evolutionary convergence curves of the different algorithms are recorded and visualized. The experimental results are summarized in Table 2 and Fig. 15.

As shown in Table 2, traditional algorithms without any improvement strategies (WOA, NGO, HHO, GWO, and BKA) exhibit slightly lower optimization time costs than their improved counterparts; however, their optimization performance remains limited, and they are prone to premature convergence to local optima. Compared with the original BKA algorithm, the proposed improved MS-BKA achieves an increase of 2.26 % in R^2 , while reducing MAE and RMSE by 26.57 % and 35.26 %, respectively, without a significant rise in optimization time. The performance enhancement of MS-BKA primarily stems from the opposition-based learning mechanism, which enhances population

diversity, and the dynamic Cauchy perturbation mechanism, which improves the algorithm's ability to escape local optima. Furthermore, compared with other improved algorithms (IGWO, IHHO, ISSA), MS-BKA still achieves the best predictive performance, with a slight increase in runtime (approximately 2567 s), demonstrating its superior efficiency and effectiveness in deep model hyperparameter optimization.

As shown in Fig. 15, some traditional algorithms (such as GWO, HHO, SSA, and BKA) converge quickly in the early stages, but become trapped in local optima in the later stages, with the curves flattening out. In contrast, their improved versions (IGWO, IHHO, ISSA, and MS-BKA) exhibit better optimization stability and lower error levels in the mid-to-late stages. In particular, the MS-BKA algorithm quickly reaches the optimal MSE in the early iteration stage and maintains the lowest error throughout, demonstrating excellent convergence speed and robustness. This indicates that MS-BKA achieves a good balance between global search and local exploitation, obtaining the optimal solution in fewer iterations, thereby significantly improving the model's convergence efficiency and prediction accuracy.

The hyperparameter optimization results for the wind power forecasting method proposed in this study are presented in Table 3.

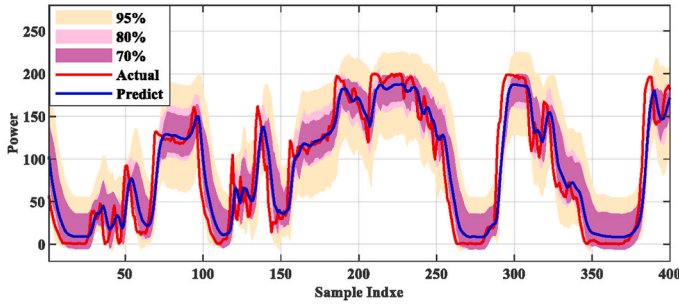
As shown in Table 3, by optimizing the FMD decomposition algorithm with MS-BKA and using minimum sample entropy as the fitness function, the optimal filter length and number of modal decompositions were determined to be $F = 45$ and $M = 6$, respectively. In the model architecture, the prediction error was set as the optimization objective. The optimal number of hidden units in the BiGRU layer was 50, while the best values for the multi-head attention layer were NumHeads = 5 and KeyDimension = 30. The model parameters were set as follows: InitialLearnRate to 10^{-2} , MiniBatchSize to 32, MaxEpochs to 150, and Dropout rate to 0.2. These parameter settings ensure the model's performance is fully leveraged and achieves optimal results.

3.3.2. Performance verification of two-stage modal decoupling strategy

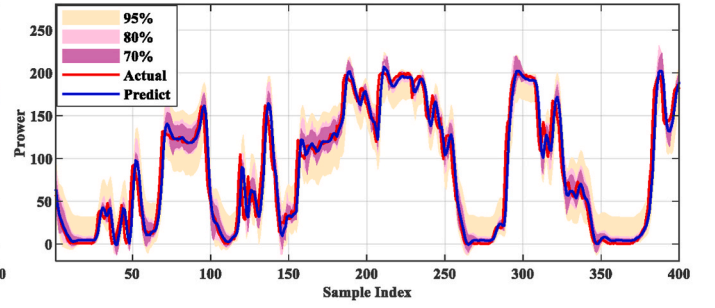
(1) Verification of the effectiveness of CEEMDAN-FMD

To investigate the impact of different signal decomposition strategies on wind power forecasting performance, multiple comparative experiments were conducted. The experiments included two categories of decomposition schemes: the first category consisted of single-stage decomposition methods (VMD, CEEMDAN, FMD, and EMD), which were used to evaluate the baseline performance of each algorithm in multi-scale feature extraction; the second category comprised two-stage decomposition strategies, in which different decomposition methods were combined to verify the effectiveness of the two-stage approach. All experiments were performed under identical data partitioning and training parameter settings to ensure the comparability and fairness of the results. Table 4 presents the forecasting performance comparison of various decomposition strategies under a fixed model architecture.

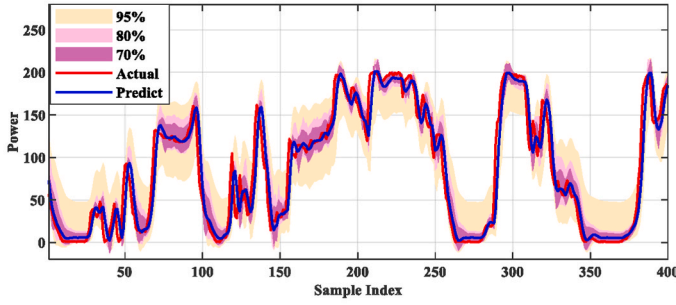
As shown in Table 4, different decomposition strategies were compared under the MSTCN-BiGRU-NSA model. The results indicate that single-stage methods are insufficient to distinguish high-frequency noise from informative signal components, leading to limited forecasting improvements. To overcome this limitation, this study applies a secondary FMD decomposition to the high-frequency modes obtained from the initial CEEMDAN step. Compared with single-stage approaches (VMD, CEEMDAN, FMD, and EMD), the proposed CEEMDAN-FMD achieves substantial performance gains, reducing MAE by 45.36 %, 31.39 %, 31.69 %, and 38.84 %, and improving R^2 by 6.14 %, 4.33 %, 4.00 %, and 5.72 %, respectively. These results confirm that CEEMDAN-FMD significantly enhances model precision by improving high-frequency feature separability and effectively suppressing noise. In comparison with other CEEMDAN-based hybrid strategies (CEEMDAN-VMD, CEEMDAN-CEEMDAN, and CEEMDAN-EMD), CEEMDAN-FMD achieves further MAE reductions of 21.92 %, 25.13 %, and 22.73 %, and



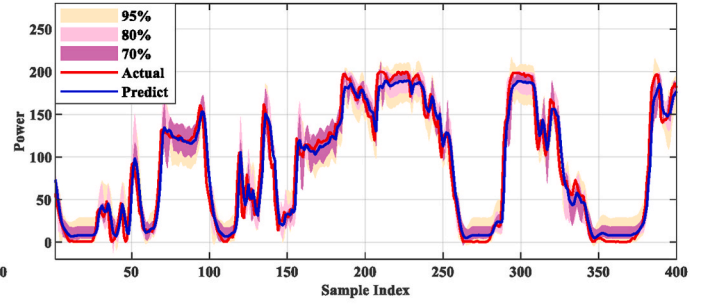
(a) FEDformer



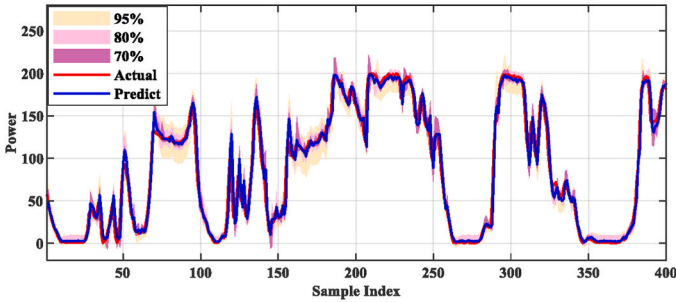
(b) DLinear



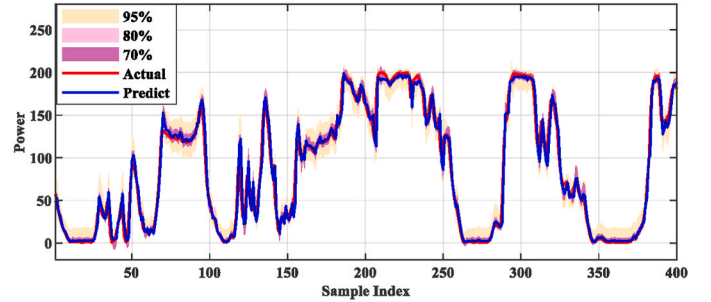
(c) Informer



(d) Transformer



(e) PatchTST



(f) Proposed

Fig. 21. Interval prediction results of different models under different confidence levels.

R^2 improvements of 1.54 %, 2.42 %, and 2.02 %, respectively, maintaining its leading performance across all combinations. Although CEEMDAN-FMD requires a slightly longer decomposition time (about 29 s), its remarkable gains in accuracy and stability highlight its efficiency at a reasonable computational cost. Overall, the two-stage CEEMDAN-FMD strategy fully capitalizes on CEEMDAN's broadband decomposition and FMD's high-frequency optimization, achieving hierarchical feature enhancement and delivering superior accuracy and robustness in wind power forecasting.

To further assess the behavior of conventional "decomposition + prediction" hybrid frameworks, two additional baseline models were incorporated. The first, VMD-ATT-LSTM, decomposes the raw wind power series using VMD and feeds the resulting modal components as multi-channel inputs into an attention-augmented LSTM network, representing a typical VMD-driven deep learning hybrid. The second, VMD-

SSA-SVR, applies VMD to obtain multi-scale components, trains an individual SVR predictor for each component, and employs the Sparrow Search Algorithm (SSA) for hyperparameter optimization; the final output is produced through linear aggregation of the modal-level predictions. Both hybrid approaches follow the same data split, input window configuration, and evaluation criteria as the proposed framework to ensure fairness and comparability.

Table 5 indicates that VMD-SSA-SVR slightly outperforms VMD-ATT-LSTM in MAE and RMSE, benefiting from SSA's adaptive optimization of SVR hyperparameters. However, the attention mechanism within LSTM improves temporal feature extraction, resulting in only minor differences between the two models. Despite both hybrid frameworks achieving R^2 values above 0.90 and maintaining stable fitting performance, they remain clearly inferior to the proposed CEEMDAN-FMD + MSTCN-BiGRU-NSA + MS-BKA + INCQR framework. This stems from

Table 8

Quality evaluation of prediction intervals of different models.

Model/ Confidence Level	QR- method	Uncertainty Performance Evaluation		Deterministic Performance Evaluation		
		PICP	PINAW	R ²	RMSE	MAE
FEDformer	INCQR	0.8104	0.3198	0.8855	23.9994	17.5043
	NCQR	0.7804	0.3306	0.8644	25.7261	18.9614
	QR	0.7727	0.3629	0.8457	26.1275	19.0534
DLinear	INCQR	0.8787	0.2616	0.9482	16.1365	10.6727
	NCQR	0.8266	0.3008	0.9101	19.2305	11.6241
	QR	0.8198	0.3089	0.9067	20.1137	12.0766
Informer	INCQR	0.8778	0.3447	0.9391	17.5026	11.8183
	NCQR	0.8258	0.3014	0.9017	19.7248	12.6514
	QR	0.8146	0.3107	0.8978	20.6759	13.6667
Transformer	INCQR	0.8849	0.1859	0.9530	12.2438	9.8864
	NCQR	0.8246	0.2961	0.9254	17.6527	11.0051
	QR	0.8211	0.2917	0.9144	18.8354	11.4017
PatchTST	INCQR	0.9107	0.1396	0.9770	9.1072	8.6346
	NCQR	0.8770	0.2037	0.9424	11.9504	10.2472
	QR	0.8739	0.2184	0.9407	11.9908	10.6628
Proposed	INCQR	0.9265	0.1044	0.9895	6.5345	6.0207
	NCQR	0.8864	0.1766	0.9634	8.7588	9.9455
	QR	0.8798	0.1894	0.9506	11.0277	9.2154

the limitations of traditional hybrid strategies, which depend on single-stage decomposition and linear recombination and thus fail to characterize the complex cross-scale non-stationary behaviors in wind power series. Furthermore, their predictors do not explicitly account for noise variation or structural uncertainty, reducing robustness under complex operating conditions. By contrast, the proposed framework integrates a hierarchical macro-/micro-scale analysis and a synergistic deep modeling design that jointly address local dynamics, global dependencies, and non-stationary sensitivity.

(2) Validation of model structure and NSA attention mechanism

To assess the performance of the proposed MSTCN-BiGRU model and its non-stationary self-attention (NSA) mechanism for wind power prediction, experiments were designed as shown in Table 5. The study compares multiple temporal modeling architectures and attention mechanisms to evaluate feature extraction ability, predictive accuracy, and computational efficiency. All models were trained on the same dataset, constructed using the proposed two-stage modal decoupling method, and their hyperparameters were tuned via the improved MSBKA algorithm to ensure fair comparison. The configurations of each model are summarized in Table 6.

Table 7 presents the comparative experimental results of the model under different experimental conditions. In Table 6, M1, M2, and M3 denote three distinct configurations: without any attention mechanism, with a conventional self-attention mechanism, and with the proposed non-stationary self-attention mechanism, respectively.

According to Table 7, under the M1 condition, the MSTCN-BiGRU model outperforms CNN-BiGRU, TCN-BiGRU, and LSTM-BiGRU, reducing MAE by 64.06 %, 29.20 %, and 48.78 %, and improving R² by 14.74 %, 2.50 %, and 9.94 %, respectively. Under both M2 and M3 settings, MSTCN-BiGRU still attains the lowest MAE and highest R², confirming that the multi-scale TCN architecture enables effective extraction of dynamic features at different temporal scales and enhances trend prediction accuracy. Under the M3 setting, compared with M1, the CNN-BiGRU, TCN-BiGRU, LSTM-BiGRU, and MSTCN-BiGRU models reduce MAE by 35.72 %, 30.72 %, 23.31 %, and 22.13 %, while improving R² by 9.68 %, 3.35 %, 7.56 %, and 2.21 %, respectively. These results confirm that integrating Non-Stationary Self-Attention (NSA) enhances prediction accuracy compared with models lacking attention mechanisms. Moreover, models using NSA (M3) outperform those with traditional self-attention (M2) in both MAE and R², proving NSA's superior capability in modeling non-stationary wind power

dynamics. By adaptively redistributing attention weights, NSA effectively captures the unstable correlations between wind speed and power, thereby improving robustness against abrupt changes and fluctuations.

From the joint analysis of model complexity and prediction accuracy, the MSTCN-BiGRU contains about 0.3654 M trainable parameters, reflecting its compact structure. While its FLOPs (0.1111G) and training time (2567s) are slightly higher than competing models, the computational overhead remains reasonable. Additionally, the model's online inference time of 0.489 s confirms strong real-time performance. In summary, the MSTCN-BiGRU achieves a favorable balance between accuracy, computational efficiency, and practical deployability.

3.3.3. Model comparison experiment

(1) Comparison of the prediction performance of multiple models

To verify the proposed model's generalization across different wind farms and years, four independent subsets were established using 2023–2024 monitoring data from Wind Farms A and B, denoted as A-2023, A-2024, B-2023, and B-2024. All subsets followed a uniform preprocessing pipeline, and the original signals were subjected to two-stage CEEMDAN-FMD decomposition, with decomposed modes provided as auxiliary features for comparison models. The proposed method was evaluated against DLinear [65], Autoformer [66], Informer [67], and PatchTST [68] under identical hyperparameter configurations and a fixed random seed. Each experiment was repeated five times for statistical reliability, and the results are shown in Fig. 16.

According to Fig. 16, conventional models (DLinear, Informer, and Autoformer) show a clear lag in regions of sharp power variation, while PatchTST ranks second to the proposed model, achieving smooth overall trends but missing local detail. The proposed model, however, fits the actual power curves of both Wind Farms A and B more accurately over two years, reflecting stronger dynamic adaptability and generalization. Analysis of the eight subplots further reveals that in rapid power-rising segments, other models display large amplitude deviations, whereas the proposed model precisely captures peak values and trends. In high-frequency oscillation areas, traditional models over-smooth their predictions, losing fine variation patterns, while the proposed model almost overlaps with the true curve, retaining detailed fluctuations effectively. Overall, the proposed model excels in dynamic feature extraction and detail preservation under non-stationary conditions, verifying its accuracy in modeling complex wind power variations. The comparative results of MAE, MSE, and R² distributions are shown in Fig. 17.

(2) Ablation experiment of model prediction performance in different seasons

To assess the proposed model's robustness and generalization across seasonal and operational variations, experiments were conducted using continuous two-year monitoring data from Wind Farm A (2023–2024). The data were divided into four seasonal subsets—spring, summer, autumn, and winter—and tested separately. Benchmark models included state-of-the-art time-series forecasting frameworks: DLinear, Informer, Transformer [69], PatchTST, FEDformer [70], and TimeMixer [71]. For fair comparison, all models were evaluated under the same experimental settings described in Section (1). The corresponding results are shown in Figs. 18 and 19.

Figs. 18 and 19 show the seasonal forecasting results for Wind Farm A (2023–2024). The blue dots denote actual power, the red dashed line represents the proposed model, and other lines indicate baseline models. In high-fluctuation periods (spring and summer), frequent wind speed changes and strong turbulence cause evident high-frequency oscillations. The proposed MSTCN-BiGRU-NSA model, integrating two-stage modal decomposition and multi-band modeling, accurately tracks the actual dynamics with no visible lag at peaks, highlighting its strong

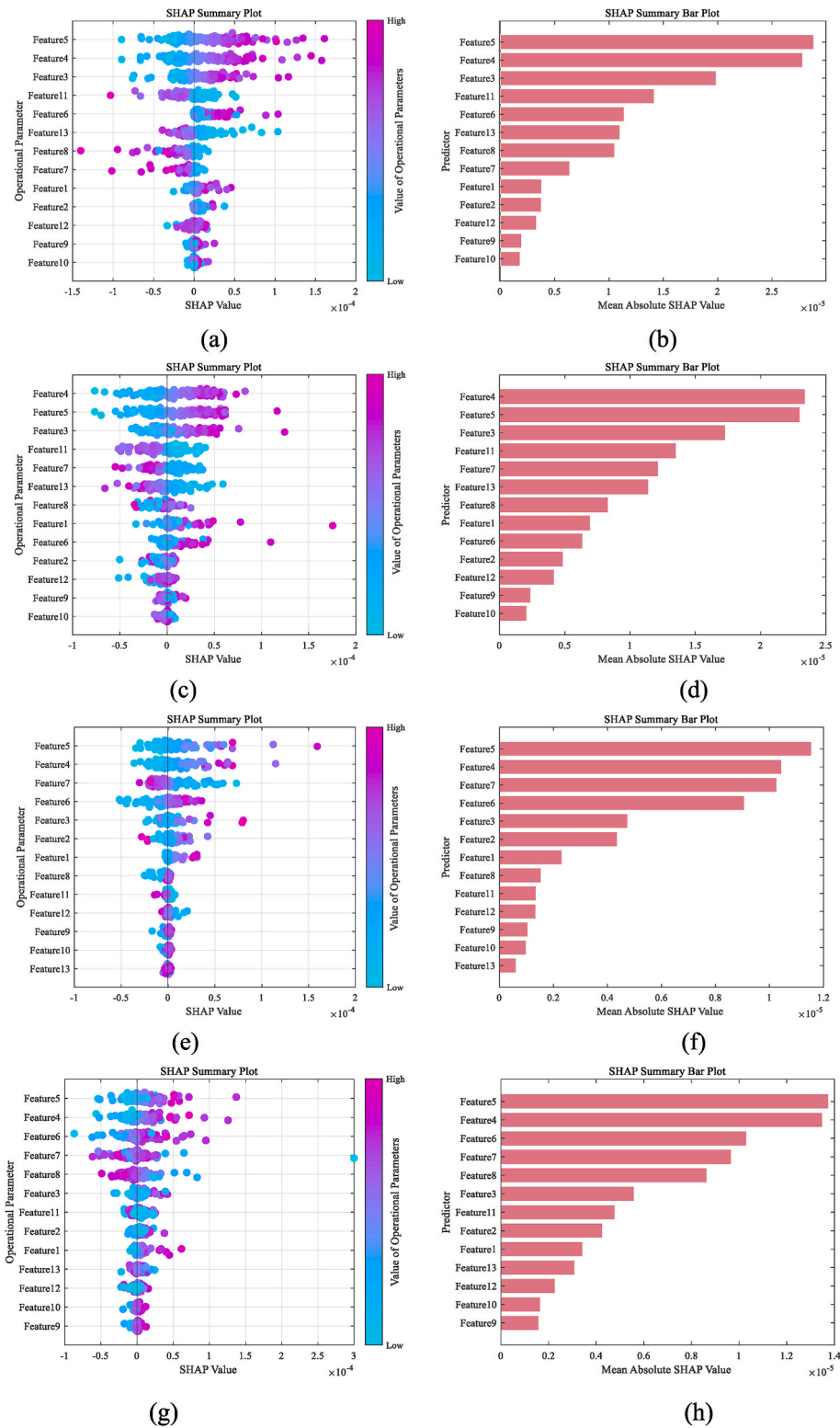


Fig. 22. The impact of different seasonal characteristics on the model: (a) and (b) are spring experimental results; (c) and (d) are summer experimental results; (e) and (f) are autumn experimental results; (g) and (h) are winter experimental results.

responsiveness to rapid fluctuations. By contrast, FEDformer and Informer tend to over- or under-estimate at peaks, while PatchTST and TimeMixer reduce oscillations but still lag during multi-peak intervals. In stable periods (autumn and winter), when the wind field is relatively steady, the proposed model yields smooth predictions consistent with actual values, indicating superior low-frequency fitting and noise

robustness. DLinear and Transformer capture general trends but show minor deviations in low-power ranges.

As shown by the cross-year prediction results, the 2023 wind power curve clearly captures the periodic characteristics of turbine operation. The proposed model performs best in spring and summer, accurately capturing peaks and valleys during rapid power variations, and achieves

Table 9

Results of encoding each feature.

Feature name	Code	Feature name	Code
Wind speed at 10m on wind tower (m/s)	Feature 1	The wind direction of the wind tower 50m (°)	Feature 8
Wind speed at 30m on wind tower (m/s)	Feature 2	The wind direction of the wind tower 70m (°)	Feature 9
Wind speed at 50m on wind tower (m/s)	Feature 3	Hub height wind direction (°)	Feature 10
Wind speed at 70m on wind tower (m/s)	Feature 4	Temperature(°)	Feature 11
Hub height wind speed (m/s)	Feature 5	Atmospheric pressure(hPa)	Feature 12
The wind direction of wind tower 10m (°)	Feature 6	Humidity (%)	Feature 13
The wind direction of the wind tower 30m (°)	Feature 7		

the highest fitting precision in autumn and winter, highlighting its strong cross-frequency adaptability. Despite the increased fluctuation amplitude and meteorological complexity in 2024, the model continues to exhibit stable prediction trends and minimal peak errors, confirming its excellent robustness and generalization across years. In contrast, other models, especially FEDformer and Informer, show excessive smoothing or local distortions in certain seasons, indicating weaker modeling capability for complex non-stationary signals. In summary, the MSTCN-BiGRU-NSA framework based on two-stage modal decomposition effectively integrates high-frequency fluctuation capture with low-frequency trend modeling, reducing the negative impact of seasonal and random disturbances. Compared with DLinear, Informer, Transformer, PatchTST, FEDformer, and TimeMixer, the proposed method demonstrates superior adaptability and stability across both seasonal and yearly variations, making it well-suited for accurate forecasting in complex wind environments.

To evaluate the seasonal forecasting performance of different models, raincloud plots were used to visualize the residual distributions. The width of each “cloud” represents the probability density of the prediction errors, depicting their dispersion and distributional characteristics. The box section illustrates the interquartile range (IQR), describing the concentration of the residuals, while the black line within the box marks the median, indicating the central bias of the model predictions. The dots display the residuals of individual samples, providing insights into the sample-level variability. The results are presented in Fig. 20.

As shown in Fig. 20, the proposed model exhibits the most stable error distributions across all seasons during 2023–2024, with concentrated and symmetric residuals whose medians are close to zero. This indicates small prediction errors, balanced bias, and stable generalization. In contrast, FEDformer and Informer display more dispersed error distributions with pronounced long-tail effects and multiple extreme errors, while their skewness further increases in 2024, suggesting poor robustness to complex meteorological disturbances. PatchTST and TimeMixer show noticeable skewed distributions and clustered outliers in certain seasons (especially summer and autumn), implying systematic over- or underestimation. Although DLinear and Transformer exhibit slightly higher residual concentration than the aforementioned models, their overall dispersion remains greater than that of the proposed model. Overall, the proposed approach maintains compact, balanced residual distributions across both seasonal and annual scales, reflecting superior stability and adaptability under varying operating conditions. In summary, the two-stage modal decomposition-based MSTCN-BiGRU-NSA integrated forecasting framework consistently produces concentrated, symmetric residuals with near-zero medians across different seasons and years, confirming its predictive consistency and robustness under both fluctuating and steady operating conditions. Compared with other models, the proposed method not only significantly reduces residual dispersion and extreme errors but also maintains a stable error

distribution structure across seasonal and annual variations, further verifying its strong generalization capability and practical applicability in complex meteorological environments.

3.3.4. Uncertainty interval assessment

To further evaluate the uncertainty inherent in the forecasting process, this study applies the improved INCQR method to quantify the predictive uncertainty of different models. The model performance is assessed under confidence levels of 70 %, 80 %, and 95 %, and the evaluation results are presented in Fig. 21.

Fig. 21 illustrates the comparison between the prediction intervals, actual values, and predicted results of different models under various confidence levels. The blue line represents the actual power, the red line denotes the predicted values, and the shaded regions in different colors correspond to different confidence intervals. As shown in Fig. 21, FEDformer and DLinear exhibit wider high-confidence intervals (95 %) during periods of sharp power fluctuations, indicating higher uncertainty under extreme conditions. Meanwhile, at a lower confidence level (70 %), their intervals fail to adequately cover the actual values, reflecting weaker model stability. Informer and Transformer show some improvement in interval compactness and trend alignment, but still exhibit noticeable deviations at local peaks and troughs. PatchTST further enhances interval stability, demonstrating stronger temporal feature extraction and convergence capability. In contrast, the proposed model maintains the narrowest and smoothest prediction intervals across all confidence levels, ensuring sufficient coverage at high confidence while significantly reducing interval width at lower confidence. These results highlight the model's superior capability in uncertainty quantification and interval calibration, providing more reliable interval references for wind power forecasting.

Table 8 presents the interval prediction performance of different models using three methods: classical quantile regression (QR), non-crossing quantile regression (NCQR), and the improved non-crossing quantile regression (INCQR) proposed in this study. The experimental results indicate that, compared with the traditional QR approach, all evaluated models (FEDformer, DLinear, Informer, Transformer, PatchTST, and the Proposed model) achieved improvements in the PICP by 4.87 %, 7.18 %, 7.76 %, 7.77 %, 4.21 %, and 5.31 %, respectively. Moreover, after applying the improved INCQR method, the PICP values of all models further increased by 3.84 %, 6.31 %, 6.30 %, 7.31 %, 3.84 %, and 4.52 %, respectively, compared with the traditional NCQR method. These results demonstrate that the INCQR method more accurately captures the dynamic variations of wind power and effectively enhances the coverage capacity of the confidence intervals, particularly under high-confidence conditions. Furthermore, in terms of interval width, the INCQR method significantly reduced the prediction interval normalized average width. For all models, the PINAW values under INCQR were considerably smaller than those obtained with NCQR and QR, confirming that the improved INCQR not only enhances interval coverage but also minimizes interval width, thereby improving the precision of uncertainty estimation. Among all models combined with the INCQR method, the proposed model achieved the highest PICP (0.9265), narrowest PINAW (0.1044), lowest RMSE (6.5345) and MAE (6.0207), as well as the best fitting accuracy ($R^2 = 98.95$). Overall, the experimental findings confirm that the proposed approach substantially enhances both the coverage and accuracy of probabilistic forecasts, while effectively narrowing the prediction intervals, thus outperforming conventional methods and demonstrating strong applicability for high-variability wind power forecasting tasks.

3.3.5. Model interpretability analysis

The interpretability of deep neural networks has long been a crucial research focus in the field of wind power forecasting, aiming to uncover the internal decision-making logic of models and enhance the transparency and credibility of prediction results [72–75]. In this study, the SHAP (Shapley Additive exPlanations) method is introduced at the

post-processing stage to perform a systematic contribution analysis of 13 input feature variables. The SHAP-based feature analysis quantitatively explains the influence of different features on the model's predictive outputs from a result-oriented perspective, revealing both the importance and directional effects of each input variable. Fig. 22 presents the distributions and mean absolute values of SHAP scores under different seasonal conditions, intuitively illustrating the positive and negative impacts and the relative contribution of each feature to the prediction results. A larger absolute SHAP value indicates a stronger influence of that feature in the forecasting process; positive values represent a driving effect on power increase, while negative values correspond to a suppressive effect on power generation. The encoding and descriptions of the input features are summarized in Table 9.

Fig. 22 illustrates the SHAP-based feature interpretability analysis, which reveals the seasonal variations in the model's response to different input features. In spring, the hub-height wind speed (Feature 5) and the 70 m tower wind speed (Feature 4) exert the most significant influence on wind power prediction, indicating that variations in high-altitude wind velocity serve as the primary drivers of power output. During summer, these two features remain highly influential, while the impact of lower-level wind speeds (Feature 1 and Feature 2) further increases, reflecting the combined effects of multi-layer wind fluctuations under intensified turbulence. In autumn, wind-speed features continue to dominate, but the contribution of mid- and low-level wind-direction features (Feature 6 and Feature 7) rises, suggesting that airflow directional stability plays a critical role in power regulation. In winter, a pattern of "wind-speed dominance, wind-direction coordination, and meteorological modulation" emerges, with the influence of wind-direction features (Feature 6 and Feature 8) further enhanced, while meteorological factors such as temperature, pressure, and humidity regulate power output through air density variation. Overall, wind-speed features remain the dominant contributors throughout the year, whereas wind-direction and meteorological variables exhibit complementary effects across seasons. The proposed model effectively captures the seasonal evolution of key feature importance, elucidating the coupling mechanism among wind speed, wind direction, and meteorological factors that govern wind power generation, thereby providing interpretable theoretical guidance for operational optimization and seasonal dispatching of wind farms.

4. Conclusion

This study proposes a wind power forecasting method that integrates a two-stage CEEMDAN-FMD modal decomposition with an MSTCN-BiGRU-NSA prediction model, further enhanced by an improved BKA optimization algorithm and the INCQR-based uncertainty modeling mechanism. Experimental results demonstrate that the proposed approach significantly improves forecasting accuracy, strengthens temporal modeling capabilities, and enhances uncertainty quantification. Specifically, CEEMDAN-FMD effectively decomposes non-stationary signals, the enhanced BKA algorithm improves model stability, and INCQR substantially increases the consistency and credibility of prediction intervals. Despite its strong performance in accuracy and robustness, the proposed method still presents challenges in terms of computational complexity and lightweight deployment. Future research will focus on model simplification and multi-scenario adaptability to improve its practical applicability and scalability.

CRedit authorship contribution statement

Wenchao Jia: Writing – review & editing, Writing – original draft, Methodology. **Aimin An:** Supervision, Methodology, Investigation, Funding acquisition. **Bin Gong:** Writing – review & editing. **Yaoke Shi:** Project administration, Methodology. **Zheming Yan:** Investigation.

Declaration of competing interest

All authors declare that they have NO affiliations with or involvement in any organization or entity with any financial interest in the subject matter or materials discussed in this manuscript.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61563032, 61963025, 62203196, 62463018) and the Gansu Provincial Science and Technology Program (22YF7GA164, 22CX8GA131, 22CX3GD005).

Data availability

Data will be made available on request.

References

- [1] Neshat M, Nezhad MM, Abbasnejad E, et al. A deep learning-based evolutionary model for short-term wind speed forecasting: a case study of the Lillgrund offshore wind farm. *Energy Convers Manag* 2021;236:114002. <https://doi.org/10.1016/j.enconman.2021.114002>.
- [2] Liu H, Chen C, Lv X, et al. Deterministic wind energy forecasting: a review of intelligent predictors and auxiliary methods. *Energy Convers Manag* 2019;195(1): 328–45. <https://doi.org/10.1016/j.enconman.2019.05.020>.
- [3] Wang Y, Zou R, Liu F, et al. A review of wind speed and wind power forecasting with deep neural networks. *Appl Energy* 2021;304(1):117766. <https://doi.org/10.1016/j.apenergy.2021.117766>.
- [4] Hou G, Wang J, Fan Y. Wind power forecasting method of large-scale wind turbine clusters based on DBSCAN clustering and an enhanced hunter-prey optimization algorithm. *Energy Convers Manag* 2024;307:118341. <https://doi.org/10.1016/j.enconman.2024.118341>.
- [5] Lopez-Villalobos CA, Rodriguez-Hernandez O, Martínez-Alvarado O, et al. Effects of wind power spectrum analysis over resource assessment. *Renew Energy* 2021; 167. <https://doi.org/10.1016/j.renene.2020.11.147>.
- [6] Parri S, Teeparthi K, Kosana V. A hybrid VMD based contextual feature representation approach for wind speed forecasting. *Renew Energy* 2023;219: 119391. <https://doi.org/10.1016/j.renene.2019.112254>.
- [7] Zhang Y, Zhao Y, Kong C, et al. A new prediction method based on VMD-PRBF-ARMA-E model considering wind speed characteristic. *Energy Convers Manag* 2020;203:112254. <https://doi.org/10.1016/j.enconman.2019.112254>.
- [8] Abdel-Aty AH, Nisar KS, Alharbi WR, et al. Boosting wind turbine performance with advanced smart power prediction: employing a hybrid ARMA-LSTM technique. *Alex Eng J* 2024;96:58–71. <https://doi.org/10.1016/j.aej.2024.03.078>.
- [9] Zhang B, Dou C, Yuan D, et al. Interval forecasting strategy of photovoltaic generation considering multi-factor self-fluctuation. *Elec Power Syst Res* 2024;234: 110495. <https://doi.org/10.1016/j.epsr.2024.110495>.
- [10] Hu H, Wang L, Zhang D, et al. Rolling decomposition method in fusion with echo state network for wind speed forecasting. *Renew Energy* 2023;216:119101. <https://doi.org/10.1016/j.renene.2023.119101>.
- [11] Wang Z, Wang L, Revanesh M, et al. Short-term wind speed and power forecasting for smart city power grid with a hybrid machine learning framework. *IEEE Internet Things J* 2023;10(21):18754–65. <https://doi.org/10.1109/JIOT.2023.3286568>.
- [12] Yang H, Niu W, Wang X, et al. Wind turbine output power forecasting based on temporal convolutional neural network and complete ensemble empirical mode decomposition with adaptive noise. *Int J Green Energy* 2023;20(14):1612–27. <https://doi.org/10.1080/15435075.2023.2169577>.
- [13] Hanifi S, Zare-Behtash H, Cammarano A, et al. Offshore wind power forecasting based on WPD and optimised deep learning methods. *Renew Energy* 2023;218: 119241. <https://doi.org/10.1016/j.renene.2023.119241>.
- [14] Liu W, Bai Y, Yue X, et al. A wind speed forecasting model based on rime optimization based VMD and multi-headed self-attention-LSTM. *Energy* 2024;294: 130726. <https://doi.org/10.1016/j.energy.2024.130726>.
- [15] Zhao Z, Yun S, Jia L, et al. Hybrid VMD-CNN-GRU-based model for short-term forecasting of wind power considering spatio-temporal features. *Eng Appl Artif Intell* 2023;121:105982. <https://doi.org/10.1016/j.engappai.2023.105982>.
- [16] Li N, Dong J, Liu L, et al. A novel EMD and causal convolutional network integrated with transformer for ultra short-term wind power forecasting. *Int J Electr Power Energy Syst* 2023;154:109470. <https://doi.org/10.1016/j.ijepes.2023.109470>.
- [17] Yuzgec U, Dokur E, Balci M. A novel hybrid model based on empirical mode decomposition and echo state network for wind power forecasting. *Energy* 2024; 300:131546. <https://doi.org/10.1016/j.energy.2024.131546>.
- [18] Li K, Huang W, Hu G, et al. Ultra-short term power load forecasting based on CEEMDAN-SE and LSTM neural network. *Energy Build* 2023;279:112666. <https://doi.org/10.1016/j.enbuild.2022.112666>.
- [19] Li Q, Wang G, Wu X, et al. Arctic short-term wind speed forecasting based on CNN-LSTM model with CEEMDAN. *Energy* 2024;299:131448. <https://doi.org/10.1016/j.energy.2024.131448>.

- [20] He Y, Zhang L, Guan T, et al. An integrated CEEMDAN to optimize deep long short-term memory model for wind speed forecasting. *Energies* 2024;17(18). <https://doi.org/10.3390/en17184615>.
- [21] Karijadi I, Chou SY, Dewabharata A. Wind power forecasting based on hybrid CEEMDAN-EWT deep learning method. *Renew Energy* 2023;218:119357. <https://doi.org/10.1016/j.renene.2023.119357>.
- [22] Chen N, Sun H, Zhang Q, et al. A short-term wind speed forecasting model based on EMD/CEEMD and ARIMA-SVM algorithms. *Appl Sci* 2022;12(12):6085. <https://doi.org/10.3390/app12126085>.
- [23] Khan M, He C, Liu T, et al. A new hybrid approach of clustering based probabilistic decision tree to forecast wind power on large scales. *Journal of Electrical Engineering & Technology* 2021;16:697–710. <https://doi.org/10.1007/s42835-020-00616-1>.
- [24] Ponkumar G, Jayaprakash S, Nagarathinam K. Advanced machine learning techniques for accurate very-short-term wind power forecasting in wind energy systems using historical data analysis. *Energies* 2023;16(14):5459. <https://doi.org/10.3390/en16145459>.
- [25] Huang CM, Chen SJ, Yang SP, et al. One-day-ahead hourly wind power forecasting using optimized ensemble prediction methods. *Energies* 2023;16(6):2688. <https://doi.org/10.3390/en16062688>.
- [26] Jamii J, Mansouri M, Trabelsi M, et al. Effective artificial neural network-based wind power generation and load demand forecasting for optimum energy management. *Front Energy Res* 2022;10:898413. <https://doi.org/10.3389/fenrg.2022.898413>.
- [27] Hong YY, Rioflorida CLPP. A hybrid deep learning-based neural network for 24-h ahead wind power forecasting. *Appl Energy* 2019;250:530–9. <https://doi.org/10.1016/j.apenergy.2019.05.044>.
- [28] Yildiz C, Acikgoz H, Korkmaz D, et al. An improved residual-based convolutional neural network for very short-term wind power forecasting. *Energy Convers Manag* 2021;228:113731. <https://doi.org/10.1016/j.enconman.2020.113731>.
- [29] Li M, Yang M, Yu Y, et al. Short-term wind power forecast based on continuous conditional random field. *IEEE Trans Power Syst* 2023;39(1):2185–97. <https://doi.org/10.1109/TPWRS.2023.3270662>.
- [30] Mo Y, Wang H, Yang C, et al. FNet: frequency filter enhanced dual LSTM network for wind power forecasting. *Energy* 2024;312:133514. <https://doi.org/10.1016/j.energy.2024.133514>.
- [31] Quan H, Zhang W, Zhang W, et al. An interval prediction approach of wind power based on Skip-GRU and block-bootstrap techniques. *IEEE Trans Ind Appl* 2023;59(4):4710–9. <https://doi.org/10.1109/TIA.2023.3270114>.
- [32] Yaghoubirad M, Azizi N, Farajollahi M, et al. Deep learning-based multistep ahead wind speed and power generation forecasting using direct method. *Energy Convers Manag* 2023;281:116760. <https://doi.org/10.1016/j.enconman.2023.116760>.
- [33] Nascimento EGS, de Melo TAC, Moreira DM. A transformer-based deep neural network with wavelet transform for forecasting wind speed and wind energy. *Energy* 2023;278:127678. <https://doi.org/10.1016/j.energy.2023.127678>.
- [34] Sun S, Liu Y, Li Q, et al. Short-term multi-step wind power forecasting based on spatio-temporal correlations and transformer neural networks. *Energy Convers Manag* 2023;283:116916. <https://doi.org/10.1016/j.enconman.2023.116916>.
- [35] Wei J, Wu X, Yang T, et al. Ultra-short-term forecasting of wind power based on multi-task learning and LSTM. *Int J Electr Power Energy Syst* 2023;149:109073. <https://doi.org/10.1016/j.ijepes.2023.109073>.
- [36] Xiao Y, Zou C, Chi H, et al. Boosted GRU model for short-term forecasting of wind power with feature-weighted principal component analysis. *Energy* 2023;267:126503. <https://doi.org/10.1016/j.energy.2022.126503>.
- [37] Jiang Z, Tan Q, Li N, et al. A novel BiGRU multi-step wind power forecasting approach based on multi-label integration random forest feature selection and neural network clustering. *Energy Convers Manag* 2024;319:118904. <https://doi.org/10.1016/j.enconman.2024.118904>.
- [38] Xiong J, Peng T, Tao Z, et al. A dual-scale deep learning model based on ELM-BiLSTM and improved reptile search algorithm for wind power prediction. *Energy* 2023;266:126419. <https://doi.org/10.1016/j.energy.2022.126419>.
- [39] Gao J, Ye X, Lei X, et al. A multichannel-based cnn and gru method for short-term wind power prediction. *Electronics* 2023;12(21):4479. <https://doi.org/10.3390/electronics12214479>.
- [40] Sun Y, Zhou Q, Sun L, et al. CNN–LSTM–AM: a power prediction model for offshore wind turbines. *Ocean Eng* 2024;301:117598. <https://doi.org/10.1016/j.oceaneng.2024.117598>.
- [41] Deng J, Xiao Z, Zhao Q, et al. Wind turbine short-term power forecasting method based on hybrid probabilistic neural network. *Energy* 2024;313:134042. <https://doi.org/10.1016/j.energy.2024.134042>.
- [42] Hu J, Hu W, Cao D, et al. Bayesian averaging-enabled transfer learning method for probabilistic wind power forecasting of newly built wind farms. *Appl Energy* 2024;355:122185. <https://doi.org/10.1016/j.apenergy.2023.122185>.
- [43] Yang L, Zheng Z, Zhang Z. An improved mixture density network via wasserstein distance based adversarial learning for probabilistic wind speed predictions. *IEEE Trans Sustain Energy* 2021;13(2):755–66. <https://doi.org/10.1109/TSTE.2021.3131522>.
- [44] Sun Y, Huang Y, Yang M. Ultra-short-term wind power interval prediction based on fluctuating process partitioning and quantile regression forest. *Front Energy Res* 2022;10:867719. <https://doi.org/10.3389/fenrg.2022.867719>.
- [45] Wen H. Probabilistic wind power forecasting resilient to missing values: an adaptive quantile regression approach. *Energy* 2024;300:131544. <https://doi.org/10.1016/j.energy.2024.131544>.
- [46] Zhang F, Leng Z, Chen L, et al. Joint probabilistic forecasting of wind and solar power exploiting spatiotemporal complementarity. *Sustainability* 2025;17(8):3584. <https://doi.org/10.3390/su17083584>.
- [47] Jonkers J, Avendano DN, Van Wallendael G, et al. A novel day-ahead regional and probabilistic wind power forecasting framework using deep CNNs and conformalized regression forests. *Appl Energy* 2024;361:122900. <https://doi.org/10.1016/j.apenergy.2024.122900>.
- [48] Hu J, Luo Q, Tang J, et al. Conformalized temporal convolutional quantile regression networks for wind power interval forecasting. *Energy* 2022;248:123497. <https://doi.org/10.1016/j.energy.2022.123497>.
- [49] Zou R, Song M, Wang Y, et al. Deep non-crossing probabilistic wind speed forecasting with multi-scale features. *Energy Convers Manag* 2022;257:115433. <https://doi.org/10.1016/j.enconman.2022.115433>.
- [50] Zhu J, He Y, Yang X, et al. Ultra-short-term wind power probabilistic forecasting based on an evolutionary non-crossing multi-output quantile regression deep neural network. *Energy Convers Manag* 2024;301:118062. <https://doi.org/10.1016/j.enconman.2024.118062>.
- [51] Yang T, Yang Z, Li F, et al. A short-term wind power forecasting method based on multivariate signal decomposition and variable selection. *Appl Energy* 2024;360:122759. <https://doi.org/10.1016/j.apenergy.2024.122759>.
- [52] Wang Y, Pei L, Li W, et al. Short-term wind power prediction method based on multivariate signal decomposition and RIME optimization algorithm. *Expert Syst Appl* 2025;259:125376. <https://doi.org/10.1016/j.eswa.2024.125376>.
- [53] Abou Houran M, Bukhari SMS, Zafar MH, et al. COA-CNN-LSTM: coati optimization algorithm-based hybrid deep learning model for PV/wind power forecasting in smart grid applications. *Appl Energy* 2023;349:121638. <https://doi.org/10.1016/j.apenergy.2023.121638>.
- [54] Chen Y, Xiao JW, Wang YW, et al. Non-crossing quantile probabilistic forecasting of cluster wind power considering spatio-temporal correlation. *Appl Energy* 2025;377:124356. <https://doi.org/10.1016/j.apenergy.2024.124356>.
- [55] Ran P, Dong K, Liu X, et al. Short-term load forecasting based on CEEMDAN and transformer. *Electr Power Syst Res* 2023;214:108885. <https://doi.org/10.1016/j.epr.2022.108885>.
- [56] Miao Y, Zhang B, Li C, et al. Feature mode decomposition: new decomposition theory for rotating machinery fault diagnosis. *IEEE Trans Ind Electron* 2022;70(2):1949–60. <https://doi.org/10.1109/TIE.2022.3156156>.
- [57] Hewage P, Behera A, Trovati M, et al. Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station. *Soft Comput* 2020;24:16453–82. <https://doi.org/10.1007/s00500-020-04954-0>.
- [58] Wu X, Lai X, Hu J, et al. Multiscale temporal convolutional network-based end-to-end recognition of drill-string stick-slip vibration in drilling process. *IEEE Trans Ind Inf* 2024. <https://doi.org/10.1109/TII.2024.3498096>.
- [59] Wu X, Zhan H, Hu J, et al. Non-stationary GNNCrossformer: transformer with graph information for non-stationary multivariate spatio-temporal wind power data forecasting. *Appl Energy* 2025;377:124492. <https://doi.org/10.1016/j.apenergy.2024.124492>.
- [60] Cui W, Wan C, Song Y. Ensemble deep learning-based non-crossing quantile regression for nonparametric probabilistic forecasting of wind power generation. *IEEE Trans Power Syst* 2022;38(4):3163–78. <https://doi.org/10.1109/TPWRS.2022.3202236>.
- [61] Wang J, Wang W, Hu X, et al. Black-winged kite algorithm: a nature-inspired meta-heuristic for solving benchmark functions and engineering problems. *Artif Intell Rev* 2024;57(4):98. <https://doi.org/10.1007/s10462-024-10723-4>.
- [62] Hou Y, Gao H, Wang Z, et al. Improved grey wolf optimization algorithm and application. *Sensors* 2022;22(10):3810. <https://doi.org/10.3390/s22103810>.
- [63] Jia W, An A, Gong B, et al. An efficient transfer fault diagnosis method integrating feature redundancy selection and multi-strategy parameter optimization. *Expert Syst Appl* 2025;279:127267. <https://doi.org/10.1016/j.eswa.2025.127267>.
- [64] Gong B, An A, Shi Y, et al. Fast fault detection method for photovoltaic arrays with adaptive deep multiscale feature enhancement. *Appl Energy* 2024;353:122071. <https://doi.org/10.1016/j.apenergy.2023.122071>.
- [65] Zeng A, Chen M, Zhang L, et al. Are transformers effective for time series forecasting? *Proc AAAI Conf Artif Intell* 2023;37(9):11121–8. <https://doi.org/10.48550/arXiv.2205.13504>.
- [66] Wu H, Xu J, Wang J, et al. Autoformer: decomposition transformers with auto-correlation for long-term series forecasting. *Adv Neural Inf Process Syst* 2021;34:22419–30. <https://doi.org/10.48550/arXiv.2106.13008>.
- [67] Zhou H, Zhang S, Peng J, et al. Informer: beyond efficient transformer for long sequence time-series forecasting. *Proc AAAI Conf Artif Intell* 2021;35(12):11106–15. <https://doi.org/10.1609/aaai.v35i12.17325>.

- [68] Nie Y. A time series is worth 64Words: long-term forecasting with transformers. arXiv preprint arXiv:2211.14730 2022. <https://doi.org/10.48550/arXiv.2211.14730>. 2022.
- [69] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. Adv Neural Inf Process Syst 2017;30. <https://doi.org/10.48550/arXiv.1706.03762>.
- [70] Zhou T, Ma Z, Wen Q, et al. Fedformer: frequency enhanced decomposed transformer for long-term series forecasting. In: International conference on machine learning. PMLR; 2022. p. 27268–86. <https://doi.org/10.48550/arXiv.2201.12740>.
- [71] Wang S, Wu H, Shi X, et al. Timemixer: decomposable multiscale mixing for time series forecasting. arXiv preprint arXiv:2405.14616 2024. <https://doi.org/10.48550/arXiv.2405.14616>.
- [72] Van Zyl C, Ye X, Naidoo R. Harnessing eXplainable artificial intelligence for feature selection in time series energy forecasting: a comparative analysis of Grad-CAM and SHAP. Appl Energy 2024;353:122079. <https://doi.org/10.1016/j.apenergy.2023.122079>.
- [73] Liao W, Fang J, Ye L, et al. Can we trust explainable artificial intelligence in wind power forecasting? Appl Energy 2024;376:124273. <https://doi.org/10.1016/j.apenergy.2024.124273>.
- [74] Liu L, Wang X, Dong X, et al. Interpretable feature-temporal transformer for short-term wind power forecasting with multivariate time series. Appl Energy 2024;374:124035. <https://doi.org/10.1016/j.apenergy.2024.124035>.
- [75] Cakiroglu C, Demir S, Ozdemir MH, et al. Data-driven interpretable ensemble learning methods for the prediction of wind turbine power incorporating SHAP analysis. Expert Syst Appl 2024;237:121464. <https://doi.org/10.1016/j.eswa.2023.121464>.