mathematics

MDPI

Article

Carbon Dioxide Emission Forecasting Using BiLSTM Network Based on Variational Mode Decomposition and Improved Black-Winged Kite Algorithm

Yueqiao Yang [1], Shichuang Li [2,*], Haijun Liu [1] and Jidong Guo [1]

[1] School of Emergency Management, Institute of Disaster Prevention, Langfang 065201, China; yangyueqiao@cidp.edu.cn (Y.Y.); liuhaijun@cidp.edu.cn (H.L.); guojidong@cidp.edu.cn (J.G.)
[2] School of Information Management, Institute of Disaster Prevention, Langfang 065201, China
* Correspondence: 23661601@st.cidp.edu.cn

Abstract: With the growing severity of global climate change, forecasting and managing carbon dioxide (CO_2) emissions has become one of the critical tasks in addressing climate change. To improve the accuracy of CO_2 emission forecasting, an innovative framework based on variational mode decomposition (VMD), improved black-winged kite algorithm (IBKA), and BiLSTM networks is proposed. This framework aims to address the challenges associated with predicting non-stationary data and optimizing model hyperparameters. Initially, experiments were conducted on 29 benchmark functions using the IBKA algorithm, demonstrating its superior performance in highly nonlinear and complex environments. Subsequently, the BiLSTM model optimized by IBKA was employed to predict CO_2 emission trends across four major industries in China, confirming its enhanced prediction accuracy. Finally, a comparative analysis with other mainstream machine learning and deep learning models revealed that the BiLSTM model consistently achieved the best predictive performance across all industries. This research proposes an efficient and practical technical pathway for intelligent carbon emission prediction under the "dual-carbon" strategic goals, offering scientific support for policy formulation and the low-carbon transition.

Keywords: improved black-winged kite algorithm; carbon dioxide emission forecasting; variational mode decomposition; hyperparameter optimization

MSC: 68T07

check for updates

Academic Editor: Ioannis Tsoulos

Received: 24 April 2025
Revised: 21 May 2025
Accepted: 30 May 2025
Published: 5 June 2025

Citation: Yang, Y.; Li, S.; Liu, H.; Guo, J. Carbon Dioxide Emission Forecasting Using BiLSTM Network Based on Variational Mode Decomposition and Improved Black-Winged Kite Algorithm. *Mathematics* **2025**, *13*, 1895. https://doi.org/10.3390/math13111895

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Anthropogenic carbon dioxide (CO_2) emissions have emerged as the dominant contributor to the intensification of the greenhouse effect, thereby driving global climate change. The rising global temperatures have triggered a spectrum of adverse environmental consequences, including the increased frequency of extreme weather events, accelerated glacial melt, sea level rise, and ecosystem degradation. These phenomena pose significant risks to ecological integrity and threaten the sustainable development of human societies [1]. As the world's largest developing country and the foremost emitter of CO_2, China plays a pivotal role in the global climate system. Its long-standing reliance on coal-dominated, carbon-intensive energy sources has underpinned rapid economic development but has simultaneously led to considerable environmental burdens [2]. Against this backdrop, China's position in international climate governance has become increasingly critical. CO_2, as the principal anthropogenic greenhouse gas, exerts profound and multifaceted effects

by altering the climate system and exacerbating challenges related to food security, water resources, biodiversity, and public health—the urgency and complexity of climate change demand robust, science-based policy frameworks for effective mitigation [3]. In response, China has articulated an ambitious dual-carbon strategy, committing to peak carbon emissions by 2030 and to achieve carbon neutrality by 2060. This strategic vision reflects China's proactive approach to climate governance and its commitment to long-term ecological sustainability. Carbon peaking constitutes a prerequisite for carbon neutrality, with both objectives fundamentally reliant on the precise quantification and management of emissions. Accurate forecasting of China's CO_2 emission trajectories is therefore critical for informing evidence-based policymaking and ensuring the effective implementation of decarbonization measures [4,5]. Furthermore, China's dual-carbon framework offers a model for integrating ecological civilization into governance, fostering a symbiotic relationship between socioeconomic development and environmental protection, and contributing constructively to global climate mitigation efforts.

The prediction of carbon dioxide (CO_2) emissions is subject to the influence of a complex interplay of multidimensional factors, including, but not limited to, macroeconomic dynamics, the structural composition of energy consumption, and seasonal variability. CO_2 emission data across various sectors in China, such as aviation, industry, and residential sectors, typically exhibit pronounced non-stationary characteristics, posing a formidable challenge to the accuracy of predictive models. To address the non-stationary characteristics of the data, widely used preprocessing methods include traditional differential methods, logarithmic transformation [6], and exponential smoothing [7]. In addition, signal decomposition techniques that have recently emerged, such as Empirical Mode Decomposition (EMD) [8], Ensemble Empirical Mode Decomposition (EEMD) [9], and Variational Mode Decomposition (VMD) [10], have also provided practical approaches for data rationalization. Among these, VMD has demonstrated superior decomposition accuracy and greater stability, primarily due to its ability to mitigate the mode mixing problem that often affects EMD. Although VMD has been widely applied in areas such as water resources prediction and energy consumption, its application to CO_2 emission prediction remains relatively limited [11,12]. Therefore, introducing VMD into carbon dioxide emission prediction is expected to offer new methods for improving prediction accuracy and achieving precise carbon management.

There are various methods for predicting carbon dioxide emissions. For instance, Lotfalipour et al. applied a grey system-based and autoregressive integrated moving average (ARIMA) to forecast Iran's carbon dioxide emissions over a decade [13]; Borisova et al. utilized the ARMA model to assess the CH4 and CO_2 emissions from a harmless waste landfill gas collection system [14]. However, both ARIMA and ARMA, as conventional statistical models based on linear assumptions, exhibit limited capability in effectively capturing the nonlinear dynamics inherent in CO_2 emission data, consequently leading to suboptimal predictive accuracy.

With the rise of machine learning, the limitations of traditional linear models are being increasingly addressed. Wen et al. employed an Improved Chicken Swarm Optimization algorithm (ICSO) to optimize Support Vector Machine (SVM) parameters, subsequently applying the constructed ICSO-SVM model to forecast residential energy-related CO_2 emissions in Shanghai [15]; Sun et al. proposed a hybrid model integrating Ensemble Empirical Mode Decomposition (EEMD) with a Particle Swarm Optimization-Backpropagation Neural Network (PSO-BPNN) to achieve short-term forecasting of carbon dioxide (CO_2) emissions [16]. However, many machine learning approaches insufficiently account for the temporal dependencies and dynamic structures inherent in time series data.

In recent years, deep learning technology has received widespread attention due to its enhanced capabilities in modeling sequential data. Wang et al. employed a Recurrent Neural Network (RNN) to predict carbon dioxide emissions [17]. As a classic time series model, RNN is capable of capturing the temporal dependencies within sequences, but it tends to suffer from the vanishing or exploding gradient problem during long-term predictions, leading to the "forgetting" of early critical information. Singh et al. applied the Long Short-Term Memory (LSTM) model to estimate vehicle carbon dioxide emissions [18]; Kumari et al. utilized the LSTM model for forecasting India's carbon dioxide emissions [19]. By introducing mechanisms of input gates, forget gates, and output gates, LSTM effectively controls the storage and transmission of information and alleviates the long-term dependency flaws of RNNs. It has demonstrated superior performance in temporal modeling tasks related to CO_2 emission forecasting. However, traditional LSTMs only utilize unidirectional information, which restricts their ability to capture comprehensive temporal patterns and bidirectional dependencies within the data.

To address this issue, Xie et al. utilized a Bidirectional Long Short-Term Memory (BiLSTM) network model for temperature data imputation [20]; Siami-Namini et al. conducted a comparative analysis of LSTM and BiLSTM models in time series prediction [21]. BiLSTM consists of forward and backward LSTM units, enabling it to capture contextual dependencies from both past and future time steps, thereby significantly enhancing the model's understanding and memory of complex sequential patterns. Studies have confirmed that BiLSTM consistently outperforms conventional LSTM models in various time series prediction tasks [22]. Therefore, this study adopts BiLSTM as the foundational model for carbon dioxide emission forecasting.

In BiLSTM, certain critical hyperparameters have a significant impact on model performance. Research has demonstrated that hyperparameters considerably impact the performance of deep learning models to a degree that may even exceed the impact of model architecture selection itself [23]. Hyperparameter optimization is finding the best combination of hyperparameters for a deep learning model, which first requires defining a search space, including the hyperparameters to be optimized and their corresponding ranges. Then, an optimization algorithm is defined to find the best solution within this search space [24,25]. Given the high dimensionality of the search space and the wide variability of individual hyperparameters, the number of possible combinations is vast, requiring extensive evaluation to determine the optimal set [26]. Consequently, hyperparameter tuning remains a major challenge in deep learning applications. When using BiLSTM to predict carbon dioxide emissions, there are two types of hyperparameters: (1) hyperparameters related to the BiLSTM structure, such as the number of units in each BiLSTM layer, the number of stacked BiLSTM layers, etc.; (2) hyperparameters related to the training of BiLSTM, such as the step size for parameter updates during model training (i.e., learning rate), the proportion of neurons randomly dropped during training (represented by the dropout rate), the maximum number of iterations allowed during training (epochs), the optimizer used in error backpropagation, loss function, etc. [27–29]. Due to the large number of hyperparameters and the combinatorial complexity of their search space, exhaustive optimization is computationally intensive. To address this, the three most influential hyperparameters—identified through rigorous sensitivity analysis—were selected for systematic optimization in this study, as summarized in Table 8.

In traditional research, these hyperparameters are typically manually determined by researchers based on empirical knowledge, a process that is inherently subjective and often incapable of ensuring optimal solutions [30]. As a result, automated hyperparameter optimization has increasingly emerged as a prominent area of research. Commonly used optimization techniques include Grid Search [31], Random Search [32], and Bayesian

Optimization [33], among others. Grid Search is straightforward and exhaustive but has high computational costs; Random Search offers reduced computational demand but suffers from a lack of consistency and reproducibility; Bayesian Optimization, by constructing a probabilistic surrogate model, efficiently directs the search process, but it is prone to becoming trapped in local optima, particularly in high-dimensional and complex parameter spaces.

In recent years, Swarm Intelligence (SI) algorithms have been successfully applied to various fields, such as wind speed forecasting, image recognition, and medical diagnosis, due to their strong global search capability and high robustness [34–36]. Yet, their application in carbon dioxide emission forecasting remains relatively underexplored. Swarm Intelligence optimization algorithms represent a class of heuristic, population-based search methods inspired by the collective behavior and decentralized coordination observed in natural systems, designed to identify optimal or near-optimal solutions to complex problems [37]. Currently, mainstream Swarm Intelligence optimization algorithms include MVO, SCA, GWO, RIME, ALO, WOA, STOA, DO, and others [38–45]. Despite the extensive proliferation of SI algorithms, the "No Free Lunch" (NFL) Theorem asserts that no single optimization algorithm can consistently outperform others across all types of problem domains [46]. In other words, if an algorithm performs well in certain situations, it may not perform as well in others. Put differently, no algorithm can excel at all types of problems [47]. Therefore, continual refinement and adaptation of existing algorithms remain essential for enhancing their performance in addressing problem-specific challenges.

The Black Kite Algorithm (BKA), proposed by Jun Wang et al. in 2024 [48], is a swarm intelligence optimization algorithm characterized by strong global search capability, fast convergence speed, and high adaptability and robustness. Benchmark function experiments have shown that the BKA algorithm significantly outperforms several mainstream metaheuristic algorithms such as MVO, SCA, GWO, RIME, ALO, WOA, STOA, and DO in optimization performance. However, the original BKA algorithm has two issues: (1) it is prone to becoming trapped in local optima during the attack phase; (2) the population may prematurely converge to suboptimal solutions during the migration phase. To address these two problems, we propose an improved Black Kite Algorithm (IBKA), which integrates a Lévy flight-based prey escape and population cooperation strategy [49,50] and a nonlinear simplex strategy [51]. The improved algorithm is called IBKA. To verify the effectiveness of the proposed IBKA algorithm, we conducted experiments on multiple benchmark test functions and compared them with mainstream algorithms, including MVO, SCA, GWO, RIME, ALO, WOA, STOA, DO, and the original BKA algorithm. The results demonstrate that IBKA consistently achieves superior optimization performance across all tested functions.

The contributions of this study are as follows:

1. The variational mode decomposition method is applied to carbon dioxide emission prediction, incorporating the nonlinear characteristics of sample data, with the aim of mitigating the impact of inherent non-stationarity in raw data on forecasting accuracy. This approach has significantly enhanced prediction precision and provides a novel perspective for the exploration of carbon dioxide emission forecasting.
2. A VMD-IBKA-BiLSTM framework is proposed for carbon dioxide emission prediction, where BiLSTM is established as a deep learning model specifically designed for carbon dioxide emission forecasting, and IBKA is formulated as an enhanced BKA algorithm dedicated to hyperparameter optimization of BiLSTM.
3. The VMD-IBKA-BiLSTM model proposed in this study was comparatively evaluated against models ARMA, ARIMA, SVM, ANN, and LSTM in predicting carbon diox-

ide emissions from four sectors in China. The results demonstrate that significant superiority of the proposed model over the comparative models is observed.

This paper is structured as follows: Section 2 is dedicated to the introduction of the signal decomposition method VMD, the carbon dioxide emission prediction model BiLSTM, and the improved optimization algorithm IBKA. Section 3 presents the comparison experiments of IBKA, BKA, and IBKA with nine swarm intelligence optimization algorithms on 29 benchmark functions. It also introduces the application of IBKA and BiLSTM in predicting carbon dioxide emissions in China. Finally, Section 4 concludes this paper.

2. Materials and Methods

A framework for predicting and optimizing carbon dioxide emissions is proposed in this study, which is named VMD-IBKA-BiLSTM. In the framework, VMD is a signal decomposition method for handling non-stationary data, which is used for processing carbon dioxide data. IBKA is a swarm intelligence optimization algorithm that is employed to optimize three critical hyperparameters of BiLSTM. BiLSTM is a deep-learning predictive model for forecasting carbon dioxide emissions. Signal decomposition, the model, and the optimization algorithm are systematically elaborated on in this section.

2.1. VMD

Variational Mode Decomposition (VMD), introduced by Dragomiretskiy and Zosso in 2014 [10], is a signal decomposition technique based on variational optimization with frequency-domain bandwidth constraints. Unlike the EMD family, VMD avoids reliance on local extrema and envelope interpolation. It offers improved stability and precise parameter control, and it effectively mitigates mode mixing.

First, a variational problem is formulated under the assumption that the carbon dioxide data f is decomposed into k components. The constraint condition is defined as the equivalence between the summation of all modes and the original signal, leading to the corresponding constrained variational expression presented in Equation (1):

$$\begin{aligned} &\min_{\{u_k,\omega_k\}}\left\{\sum_k\left|\sigma_l\left[\left(\delta(t)+\frac{j}{\pi t}\right)*u_k(t)\right]e^{-j\omega_k t}\right|_2^2\right\} \\ &\sum_k u_k = f(t)n \end{aligned} \tag{1}$$

here, $\{u_k\}$ is the kth modal component, $\{\omega_k\}$ is the central frequency of the kth mode, $\delta(t)$ is the Dirac delta function, and * denotes the convolution operator.

Then, the constructed variational problem is solved by introducing the Lagrange multiplier λ. The original constrained variational problem was converted into an unconstrained variational problem. The augmented Lagrange expression was obtained and shown in Equation (2):

$$\begin{aligned} L(\{u_k\},\{\omega_k\},\lambda) &= \alpha\sum_k\left\|\sigma_l\left[\left(\delta(t)+\frac{j}{\pi t}\right)*u_k(t)\right]e^{-j\omega_k t}\right\|_2^2 \\ &+\left\|f(t)-\sum_k u_k(t)\right\|_2^2+\left\langle\lambda(t),f(t)-\sum_k u_k(t)\right\rangle \end{aligned} \tag{2}$$

where α is the quadratic penalty factor for reducing the interference of Gaussian noise, and $\lambda(t)$ represents the Lagrange operator.

Through the alternate direction multiplier iteration algorithm and Fourier isometric transformation optimization, the modal components and central frequencies are obtained. Among them, the range of the central frequency is $[0, +\infty)$. This range is strictly controlled by the Fourier isometric transformation optimization, which operates as follows: First,

leveraging the energy isometry of the Fourier transform (Parseval's theorem), the variational optimization problem is transferred to the frequency domain, ensuring energy consistency between time and frequency domains. This allows precise constraint of each mode's bandwidth (regulated by the penalty factor α), confining the central frequency ω_k to the dominant frequency range of the actual signal components and avoiding frequency shifts caused by energy leakage. Second, due to the inherent symmetry of the Fourier transform, the transformation yields negative frequencies that lack physical significance. To address this, the Fourier isometric transformation optimization only updates the functionals for non-negative frequencies ($\omega \geq 0$) and performs a "double lift" (energy concentration constraint) on these $\omega \geq 0$, directly filtering out non-physical negative frequencies. Therefore, for all $\omega \geq 0$ (non-negative central frequencies), the functionals $\hat{u}_k$ and ω_k are updated. All $\omega \geq 0$ are performed a double lift. The detailed expressions can be found in Equations (3)–(5):

$$\hat{u}_k^{n+1}(\omega) = \frac{\hat{f}(\omega) - \sum\limits_{i \neq k} u_i(\omega) + \hat{\lambda}(\omega)/2}{1 + 2\alpha(\omega - \omega_k)^2} \tag{3}$$

$$\omega_k^{n+1} = \frac{\int_0^\infty \omega \left| \hat{u}_k^{n+1}(\omega) \right|^2 d\omega}{\int_0^\infty \left| \hat{u}_k^{n+1}(\omega) \right|^2 d\omega} \tag{4}$$

$$\hat{\lambda}_k^{n+1}(\omega) = \hat{\lambda}_k^n(\omega) + \gamma \left(\hat{f}(\omega) - \sum_k \hat{u}_k^{n+1}(\omega) \right) \tag{5}$$

Among these variables, $\hat{u}_k^{n+1}$, $u_i(\omega)$, and $\hat{f}(\omega)$ represent the Fourier transforms of $u_k^{(n+1)}(t)$, $u_i(t)$, and $f(t)$, respectively. The constraints are satisfied until the condition in Equation (6) is met:

$$\sum_{k=1}^{K} \left(\frac{\| \hat{u}_k^{(n+1)} - \hat{u}_k^n \|_2^2}{\| \hat{u}_k^n \|_2^2} \right) < \in \tag{6}$$

2.2. Carbon Dioxide Emission Forecasting Model: BiLSTM

The BiLSTM network utilized in this study is a bidirectional extension of the standard Long Short-Term Memory (LSTM) architecture. It employs three gating mechanisms—forget gate, input gate, and output gate—to regulate the flow of information and effectively capture temporal dependencies in time-series data. Specifically, the forget gate controls the retention of past cell state information, the input gate manages the incorporation of new input data into the cell state, and the output gate determines the portion of the cell state to be propagated to the hidden state. The structure of the LSTM unit is illustrated in Figure 1.

The calculation of LSTM is shown in Equations (7)–(12):

$$f_t = \sigma(w_f \times h_{t-1} + u_f \times x_t + b_f) \tag{7}$$

$$i_t = \sigma(w_i \times h_{t-1} + u_i \times x_t + b_i) \tag{8}$$

$$\widetilde{c}_t = \tanh(w_c \times h_{t-1} + u_c \times x_t + b_c) \tag{9}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c}_t \tag{10}$$

$$o_t = \sigma(w_o \times h_{t-1} + u_o \times x_t + b_o) \tag{11}$$

$$h_t = o_t \odot \tanh(c_t) \tag{12}$$

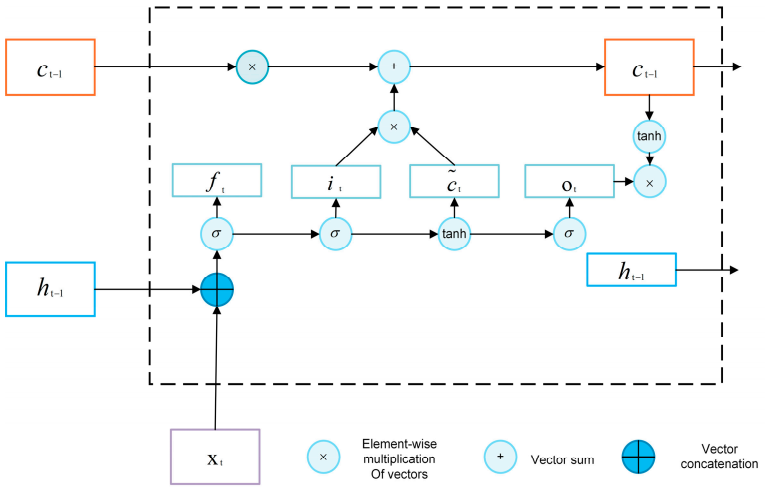


Figure 1. LSTM state gating structure.

Among them, represents the forget gate, represents the input gate, represents the output gate, the new state is the additional new information learned at the current moment t, and the current information is how much information is extracted from previous moments plus how much information is extracted from the current moment.

A BiLSTM unit consists of a forward LSTM unit and a backward LSTM unit, which extract forward and backward temporal features from the sequence, respectively. BiLSTM has stronger sequence modeling capabilities than the basic LSTM; therefore, it is chosen as the basic unit of the prediction model. Its specific structure is shown in Figure 2.

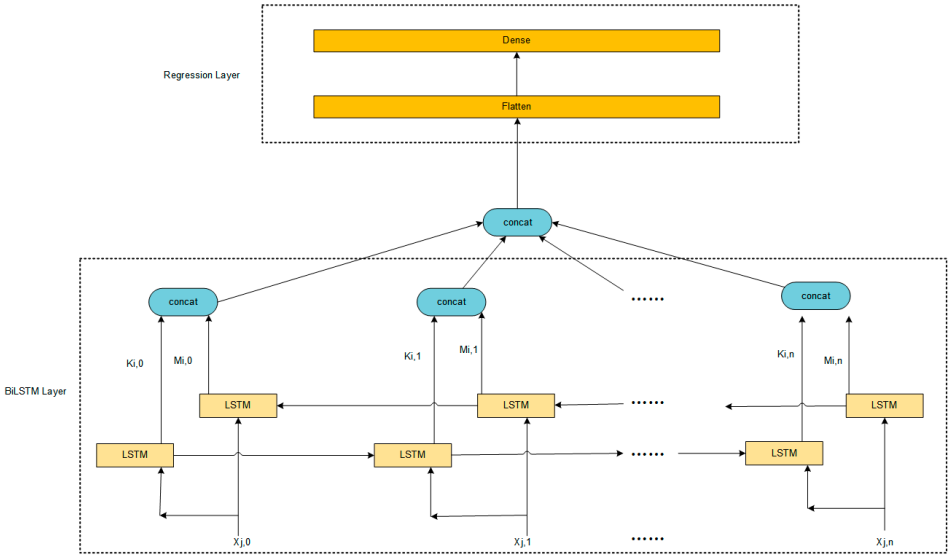


Figure 2. BiLSTM model structure diagram.

2.3. IBKA

The BiLSTM model proposed in the study contains many hyperparameters that significantly influence its performance. Therefore, an optimization algorithm is needed to find the optimal hyperparameters. The Black-winged Kite Algorithm (BKA) is improved by adding prey escape and population cooperation strategies based on Lévy flight behavior and nonlinear simplex strategies. The IBKA is proposed. Then, IBKA is used to optimize BiLSTM. The original BKA and our improved IBKA are introduced separately in the section.

2.3.1. The Original BKA

The Black Kite Algorithm (BKA), proposed by Jun Wang et al. in 2024 [48], is a metaheuristic optimization algorithm inspired by the black kite's predatory and migratory behaviors and strong hovering ability. BKA simulates these behaviors through three main stages: population initialization, predatory behavior, and migratory behavior. The following section presents each of these stages separately:

1. Population initialization phase

$$\boldsymbol{BK} = \begin{bmatrix} \boldsymbol{BK}_{1,1} & \boldsymbol{BK}_{1,2} & \ldots & \boldsymbol{BK}_{1,d} \\ \boldsymbol{BK}_{2,1} & \boldsymbol{BK}_{2,2} & \ldots & \boldsymbol{BK}_{2,d} \\ \boldsymbol{BK}_{3,1} & \boldsymbol{BK}_{3,2} & \ldots & \boldsymbol{BK}_{3,d} \\ \cdot\cdot & \cdot\cdot & \cdot\cdot & \\ \vdots & \cdot\cdot & \cdot\cdot & \\ \boldsymbol{BK}_{n,1} & \boldsymbol{BK}_{n,2} & \ldots & \boldsymbol{BK}_{n,d} \end{bmatrix} \tag{13}$$

where n is the number of hyperparameters, d is the dimension size of the given problem, and is the j-th dimension of the i-th black-winged kite. The positions of each black-winged kite are distributed uniformly.

$$X_i = \boldsymbol{BK}_{lb} + rand(\boldsymbol{BK}_{ub} - \boldsymbol{BK}_{lb}) \tag{14}$$

where represents the coordinate of the i-th Black-winged Kite in the j-th dimension, and are the lower and upper bounds of the dimension, respectively, and the rand is a random value with a range of [0, 1].

During the initialization phase, BKA selects the individual with the best fitness value in the initial population as the leader, representing the black kite's optimal position. The mathematical formulation for identifying the initial leader (based on fitness minimization) is given as follows:

$$f_{best} = \min(f(X_i)) \tag{15}$$

$$X_L = X(find(f_{best} == f(X_i))) \tag{16}$$

2. Aggressive behavior

The Black-winged Kite initiates its attack by diving toward its prey at high speed. Figure 3a illustrates the kite in an attack state, while Figure 3b depicts its hovering posture. The mathematical model of the predatory behavior is presented below:

$$y_{t+1}^{i,j} = \begin{cases} y_t^{i,j} + n(1 + \sin(r)) \times y_t^{i,j} & p < r \\ y_t^{i,j} + n(2r - 1) \times y_t^{i,j} & else \end{cases} \tag{17}$$

$$n = 0.05 \times e^{-2 \times (\frac{t}{T})^2} \tag{18}$$

The following is a detailed description of the formula:

- $y_t^{i,j}$ and $y_{t+1}^{i,j}$ represent the position of the i-th black-winged kite in the j-th dimension at the t-th and $(t + 1)^{th}$ iteration steps, respectively.
- r is a random number between 0 and 1, while p is a constant value equal to 0.9.
- T is the total number of iterations, and t is the number of iterations that have been completed.
- n is the dynamic perturbation coefficient.

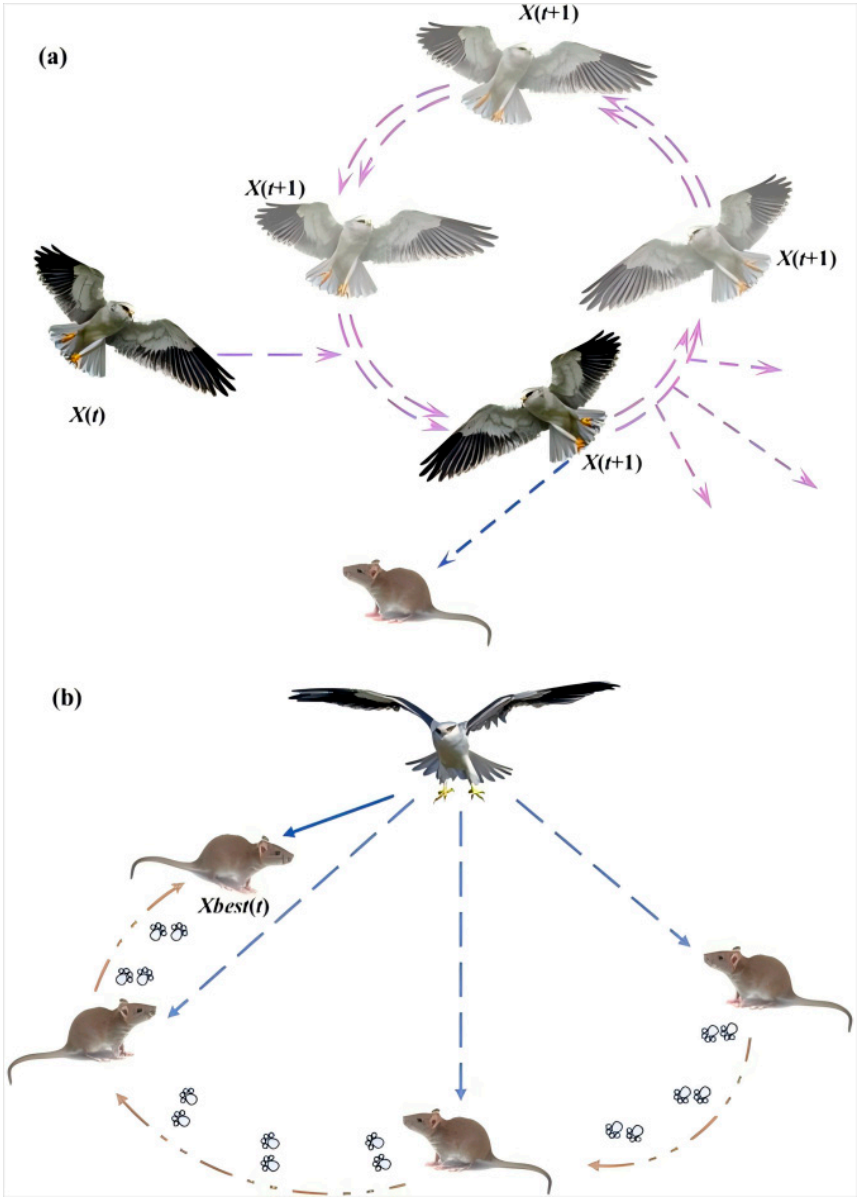


Figure 3. Black-winged kites have two attack strategies: (**a**) hovering in the air, waiting to attack, and (**b**) hovering in the air, searching for prey.

3. Migration behavior

Inspired by bird migration behavior, the following hypothesis is proposed: if the fitness of the current population is lower than that of a randomly generated population, the current leader relinquishes its role and joins the migrating population, indicating its unsuitability to continue leading. Conversely, if the current population's fitness is higher, the leader continues to guide the group toward the objective. This strategy enables dynamic leader selection to ensure effective migration. The mathematical model for the migration behavior of the Black-winged Kite is given below:

$$y_{t+1}^{ij} = \begin{cases} y_t^{ij} + C(0,1) \times \left(y_t^{ij} - L_t^j\right), & F_i < F_{ri} \\ y_t^{ij} + C(0,1) \times \left(L_t^j - m \times y_t^{ij}\right), & else \end{cases} \tag{19}$$

$$m = 2 \times \sin\left(r + \frac{\pi}{2}\right) \tag{20}$$

- L_t^j represents the leading scorer in the *j*-th dimension for the black-winged kite at the *t*-th iteration so far.
- $y_t^{i,j}$ and represent the position of the *i*-th black-winged kite in the *j*-th dimension at the *t*-th and $(t + 1)^{th}$ iteration steps, respectively.
- F_i represents the current position of any black-winged kite in the *j*-th dimension at the *t*-th iteration.
- F_{ri} represents the fitness value of the random position in the j-th dimension obtained from any black-winged kite at the t-th iteration.
- C (0,1) represents Cauchy mutation. It is defined as follows:

The one-dimensional Cauchy distribution is a continuous probability distribution with two parameters. The following equation shows the probability density function of the univariate Cauchy distribution:

$$f(x, \delta, \mu) = \frac{1}{\pi} \times \frac{\delta}{\delta^2 + (x - \mu)^2}, -\infty < x < \infty \tag{21}$$

when $\delta = 1$, $\mu = 0$, its probability density function will become the standard form. Here is the exact formula:

$$f(x, \delta, \mu) = \frac{1}{\pi} \times \frac{1}{x^2 + 1}, -\infty < x < \infty \tag{22}$$

2.3.2. Proposed IBKA

The original BKA algorithm exhibits significant advantages in global search, convergence speed, and robustness. However, it also has some drawbacks. For instance, it tends to fall into local optima during attack behaviors, and the population may converge prematurely to suboptimal solutions during migration behaviors. To address these two issues, we have added two mechanisms to the basic BKA algorithm. The following outlines our improvements in detail.

1. Lévy Flight-Inspired Prey Escape and Collective Cooperation Strategies

The original BKA algorithm includes global search and various attack behaviors during the search process in attack behaviors. However, it overlooks the situation where the prey escapes, resulting in the failure to find the global optimal solution. The algorithm is prone to falling into local optimal solutions and lacks information exchange during the attack behaviors, which limits the global search capability. Inspired by the Harris hawk optimization algorithm [50] in the exploration and exploitation phases, prey escape, and group cooperation, and the Lévy flight behavior in biological migration [49], we propose a prey escape and population cooperation strategy based on Lévy flight behavior. When the prey senses a threat, it takes escape behavior to evade predators. This behavior can be considered as a mechanism to avoid falling into local optimal solutions in optimization algorithms. In the improved BKA algorithm, we define the formula for prey escape energy as follows:

$$E = \|X_i - X_L\|_2 = \sqrt{\sum_{k=1}^{d} (X_{i,k} - X_{L,k})^2} \tag{23}$$

where E represents the energy of prey escape, denotes the position vector of the *i*-th individual in the kth dimension, and represents the position vector of the current leader individual in the kth dimension.

When abs (E) < 0.1, it indicates that the solution is close to the optimal one. Introduce a population collaboration strategy; the formula is as follows:

$$X_{new} = X_i + n \times E \times (1 + \sin(r)) \times X_i \tag{24}$$

where n refers to the dynamic disturbance coefficient, X_i represents the current individual's position vector, r is a random number between 0 and 1, and X_{new} denotes the position after updating the individual.

When abs (E) $\geq$ 0.1, it indicates that the current individual is far from the optimal solution, posing a risk of falling into a local optimum or search stagnation. We have introduced a Lévy flight global perturbation strategy, and its updated formula is as follows:

$$X_{new} = X + r \times \mathrm{Le'vy}(d) \times (X - X_L) \tag{25}$$

where Lévy (d) is a random step size generated based on the Lévy distribution, which involves the following formula:

$$step \sim \frac{u}{|v|^{1/\beta}} \tag{26}$$

where u $\sim N(0, \sigma^2)$ and v $\sim N(0,1)$ are random variables from standard normal distributions; is the Lévy index, controlling the heavy-tailed property of the step size (we set $\beta = 1.5$ in this paper).

2. Nonlinear Simplex Strategy

In the migration behavior design of the original Black Kite Algorithm (BKA), although the combination of the dynamic leader selection mechanism and the Cauchy mutation strategy effectively maintains the migratory directionality of the population, the mechanism still possesses potential for further optimization. Particularly when addressing complex nonlinear optimization problems, the algorithm tends to prematurely converge to local optima rather than the global optimum. The nonlinear simplex strategy is introduced to address this issue [51]. The nonlinear simplex method is a classical direct search approach frequently employed for solving extremum problems of multivariate functions. Its core idea is to gradually approximate the optimal solution of the function by continuously adjusting the positions of the simplex vertices. In the improved BKA algorithm, the nonlinear simplex strategy is applied during the migration behavior phase. Specifically, a portion of individuals with poorer fitness values are selected from the current population as the simplex vertices to be optimized. Then, the adjustment mechanism of the nonlinear simplex method is used. The positions of these vertices are updated in the hope of finding better solutions iteratively. During the updating process, operations such as reflection, expansion, contraction, and shrinking are employed to adjust the shape and position of the simplex. Through multiple iterations, when the shape and position of the simplex tend to stabilize, the algorithm has converged to the optimal or near-optimal solution within the current search space. In this paper, the reflection operation is selected to adjust the shape and position of the simplex, and its reflection formula is as follows:

$$X_r = c + r(t) \times (c - X_w) \times \cos(\theta) \tag{27}$$

where c is the simplex's optimal vertex, is the simplex's worst vertex, and θ is a random angle in U [0, 2π].

The final proposed IBKA flowchart is shown in Figure 4, where the water-green section represents our improvements.

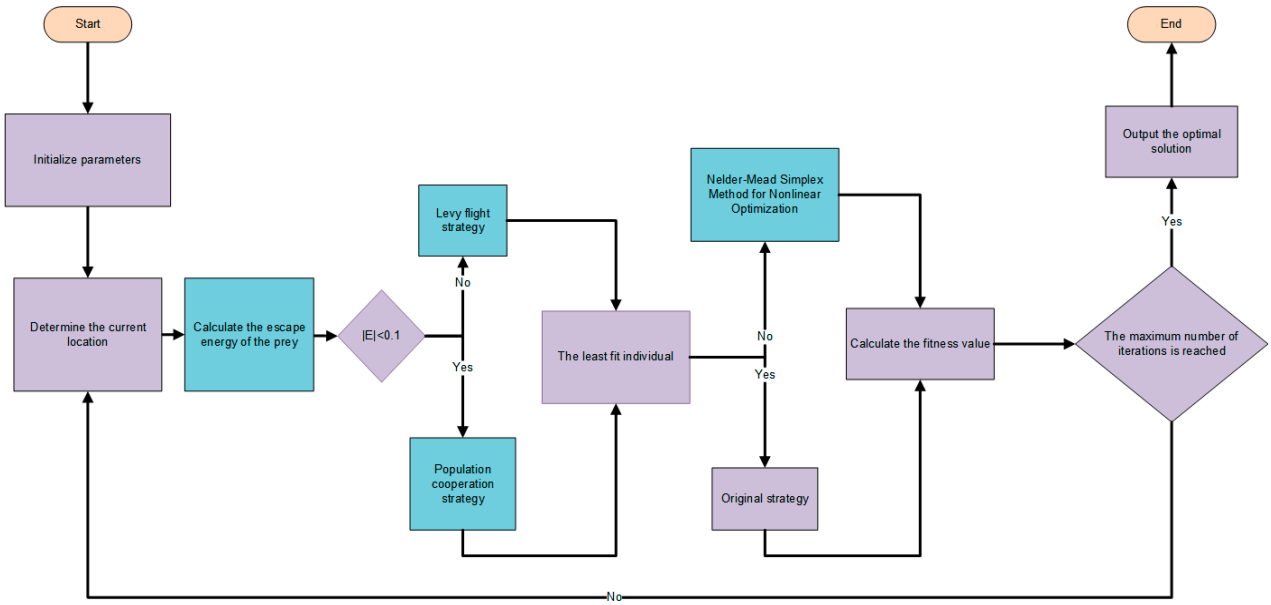


Figure 4. Flowchart of IBKA.

3. Results and Discussion

3.1. IBKA Performance Verification Experiments

In this section, experiments were conducted to evaluate the optimization performance of the proposed IBKA, including its development, exploration, and local optimum avoidance capabilities. These included ablation studies and comparisons with nine state-of-the-art metaheuristic algorithms. All algorithms were tested under identical conditions on a system with a 12th Gen Intel i5-12500H processor and NVIDIA GeForce RTX 4060 GPU (8 GB VRAM), using 500 iterations and a population size of 30.

3.1.1. Benchmark Functions

To evaluate the performance of IBKA, 29 different benchmark functions are selected from the CEC2017 test suite for comparative experiments [52]. Four types are categorized into 29 benchmark functions. Among them, F1–F3 are unimodal benchmark functions used to test the development ability of the algorithm (after the official deletion of the original F2, it was adjusted to F1–F3); F4–F10 are multimodal benchmark functions used to test the exploration ability of the algorithm; F11¬–F20 are hybrid functions used to test the adaptability in heterogeneous problems; F21–F30 are composition functions used to assess the algorithm's ability to avoid local optima. The range represents the boundaries of the variables, and F (min) is the optimal value. The CEC2017 benchmark functions are shown in Table 1.

In this section, each algorithm is executed 30 times on each benchmark function to mitigate the impact of random factors. In this paper, the Friedman test method is used to rank and evaluate the results of all algorithms on the benchmark functions.

Table 1. CEC2017 Benchmark Functions.

Type	No.	Functions	Fi * = Fi (x*)
Unimodal Functions	1	Shifted and Rotated Bent Cigar Function	100
	3	Shifted and Rotated Zakharov Function	200

Table 1. *Cont.*

Type	No.	Functions	Fi * = Fi (x*)
Simple Multimodal Functions	4	Shifted and Rotated Rosenbrock's Function	300
	5	Shifted and Rotated Rastrigin's Function	400
	6	Shifted and Rotated Expanded Scaffer's F6 Function	500
	7	Shifted and Rotated Lunacek Bi _ Rastrigin Function	600
	8	Shifted and Rotated Non-Continuous Rastrigin's Function	700
	9	Shifted and Rotated Levy Function	800
	10	Shifted and Rotated Schwefel's Function	900
Hybrid Functions	11	Hybrid Function 1 (N = 3)	1000
	12	Hybrid Function 2 (N = 3)	1100
	13	Hybrid Function 3 (N = 3)	1200
	14	Hybrid Function 4 (N = 4)	1300
	15	Hybrid Function 5 (N = 4)	1400
	16	Hybrid Function 5 (N = 4)	1500
	17	Hybrid Function 6 (N = 5)	1600
	18	Hybrid Function 6 (N = 5)	1700
	19	Hybrid Function 6 (N = 5)	1800
	20	Hybrid Function 6 (N = 6)	1900
Composition Functions	21	Composition Function 1 (N = 3)	2000
	22	Composition Function 2 (N = 3)	2100
	23	Composition Function 3 (N = 4)	2200
	24	Composition Function 4 (N = 4)	2300
	25	Composition Function 5 (N = 5)	2400
	26	Composition Function 6 (N = 5)	2500
	27	Composition Function 7 (N = 6)	2600
	28	Composition Function 8 (N = 6)	2700
	29	Composition Function 9 (N = 3)	2800
	30	Composition Function 10 (N = 3)	2900
Search Range: [−100, 100]D			

3.1.2. Ablation Analysis of the IBKA

We integrate prey escape and population cooperation strategies based on Lévy flight behavior, as well as a nonlinear simplex strategy, into the basic BKA algorithm and propose the IBKA algorithm. This section verifies the performance enhancement of the basic BKA through ablation experiments. Table 2 shows the BKA variant algorithms with one or more integrated mechanisms, where "1" indicates that the mechanism is added, and "0" indicates the opposite. These BKA variant algorithms are compared and tested on the previously mentioned 29 benchmark functions.

Table 2. Various BKA variants with two mechanisms.

Model	LFPECCS [1]	NSS [2]
IBKA	1	1
LBKA	1	0
NBKA	0	1
BKA	0	0

[1] LFPECCS: Lévy Flight-Inspired Prey Escape and Collective Cooperation Strategies.[2] NSS: Nonlinear simplex strategy.

The experimental results are given in Table 2, where Avg represents the Friedman test average ranking of the algorithm on 29 benchmark functions. The lower the average value, the better the algorithm's performance. "+/−/=" represents that IBKA is better, worse, or equal to other BKA variants; rank represents the final ranking of the algorithm.

From Table 3, the performance of the basic BKA is the worst. Whether adding a prey escape and population cooperation strategy based on Lévy flight behavior or a nonlinear

simplex strategy, the performance of BKA can be improved. IBKA ranks first, which means adding both the prey escape and population cooperation strategy based on Lévy flight behavior and the nonlinear simplex strategy to BKA can effectively enhance its optimization capability.

Table 3. The results of the ablation experiment.

Algorithm	Rank	+/−/=	Avg
IBKA	1	~	1.6552
LBKA	2	18/0/12	2.0690
NBKA	3	25/0/5	2.5862
BKA	4	28/0/2	3.6897

3.1.3. Comparison with Other Algorithms

A comprehensive performance evaluation of the proposed IBKA was conducted by comparing it with eight state-of-the-art algorithms, along with the original BKA algorithm. The comparison algorithms used in this study are as follows:

- Multi-verse optimization algorithm (MVO) [38]
- Sine cosine algorithm (SCA) [39]
- Grey wolf optimization (GWO) [40]
- Rime optimization algorithm (RIME) [41]
- Ant lion optimization (ALO) [42]
- The whale optimization algorithm (WOA) [43]
- Sooty tern optimization algorithm (STOA) [44]
- Dandelion optimization (DO) [45]
- Black-winged kites algorithm (BKA) [48]

The parameter settings of the aforementioned comparison algorithm are shown in Table 4:

Table 4. Parameter settings of comparison algorithms.

Algorithm	Parameter	Value
MVO	Coefficient of wormhole expansion w	w∈(0.1–0.5)
SCA	Convergence parameter spiral factor a	a = 2
GWO	Area vector a, random vector r1, r2	A ∈ [0, 2], r1 ∈ [0, 1], r2 ∈ [0, 1]
RIME	Ice crystal growth rate α	α∈(0.5–2.0)
ALO	Wandering step decay rate c	c∈(1E−5–1E−2)
WOA	Convergence factor decay rate a	a∈(2→0)
STOA	Migration step factor	α∈(0.5–2.0)
DO	Enveloping contraction factor γ, Track the step factor α	γ∈(1.0–3.0), α∈(0.5–2.0)
BKA	Hover step factor α, Diving intensity factor γ	γ∈(1.0–3.0), α∈(0.5–2.0)

All algorithms are tested under the same environment with a maximum of 500 iterations and a population size of 30. To minimize the impact of randomness on the experimental results, each algorithm was independently executed 30 times on each benchmark function, and the average (Aver) and standard deviation (Std) of the 30 independent runs were calculated for each algorithm on each test function (presented in Table 5, with the best test results highlighted in bold). A smaller average indicates better performance.

Table 5. Experimental results on 29 benchmark functions (the best are in bold).

Fun	F1		F3		F4	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	**8.1830E+03**	**7.0422E+03**	**1.8912E+03**	**1.0926E+03**	**5.0512E+02**	**1.8824E+01**
MVO	2.0834E+06	5.5248E+05	1.7653E+04	8.1720E+03	5.0848E+02	2.8785E+01
SCA	2.1779E+10	3.7625E+09	8.0469E+04	1.4987E+04	2.9913E+03	9.6940E+02
GWO	3.0973E+09	2.0276E+09	6.5206E+04	1.3765E+04	6.6954E+02	9.4817E+01
RIME	3.9959E+06	1.5350E+06	5.6550E+04	2.0593E+04	5.2317E+02	3.0755E+01
ALO	1.8263E+04	1.4310E+04	2.2269E+05	7.2692E+04	5.5684E+02	3.5674E+01
WOA	5.1528E+09	2.2079E+09	2.6804E+05	6.6301E+04	1.3270E+03	3.3620E+02
STOA	1.1517E+10	3.6744E+09	6.9199E+04	1.1507E+04	1.0874E+03	3.7627E+02
DO	9.0487E+05	5.1165E+05	3.7217E+04	1.4747E+04	5.2275E+02	3.0270E+01
BKA	9.3339E+09	9.5028E+09	3.2213E+04	1.4535E+04	2.4960E+03	3.3167E+03
	F5		**F6**		**F7**	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	6.7015E+02	4.6326E+01	6.3754E+02	1.5282E+01	1.0409E+03	1.0912E+02
MVO	6.2880E+02	4.8084E+01	6.3446E+02	1.4620E+01	8.7723E+02	3.7373E+01
SCA	8.2968E+02	2.665E+01	6.6372E+02	6.6056E+00	1.2586E+03	6.6517E+01
GWO	6.3001E+02	4.4073E+01	6.1339E+02	4.1681E+00	9.0428E+02	5.5862E+01
RIME	**6.2020E+01**	**3.8822E+01**	**6.1278E+02**	**6.0731E+00**	**8.7318E+02**	**4.4449E+01**
ALO	6.7634E+02	4.9636E+01	6.4684E+02	7.4114E+00	1.1289E+03	8.7380E+01
WOA	8.7229E+02	5.1075E+01	6.8024E+02	1.3970E+01	1.3262E+03	7.7212E+01
STOA	7.3505E+02	3.1691E+01	6.4909E+02	7.7178E+00	1.1410E+03	6.6825E+01
DO	6.7471E+02	4.2709E+01	6.4089E+02	1.4148E+01	1.0196E+03	8.4731E+01
BKA	7.4849E+02	5.0098E+01	6.6304E+02	1.0978E+01	1.2209E+03	5.0611E+01
	F8		**F9**		**F10**	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	9.3957E+02	3.2733E+01	3.9205E+03	1.0900E+03	**4.7605E+03**	**7.8266E+02**
MVO	9.2897E+02	3.3092E+01	6.5787E+03	3.6856E+03	4.8820E+03	5.8507E+02
SCA	1.0954E+03	2.8364E+01	8.1511E+03	1.6601E+03	8.7252E+03	4.2182E+02
GWO	**9.0659E+02**	**2.4623E+01**	**2.5647E+03**	**1.1602E+03**	5.5303E+03	1.6354E+03
RIME	9.1213E+02	2.4865E+01	2.8693E+03	1.5267E+03	4.8791E+03	5.3497E+02
ALO	9.4792E+02	3.5205E+01	4.5019E+03	1.2755E+03	5.6546E+03	7.1469E+02
WOA	1.0904E+03	6.5874E+01	1.0858E+04	4.1747E+03	7.6357E+03	5.7563E+02
STOA	9.9967E+02	3.0121E+01	6.5067E+03	1.7392E+03	7.4005E+03	6.6993E+02
DO	9.6499E+02	3.9626E+01	6.1139E+03	1.9411E+03	5.2652E+03	6.1188E+02
BKA	9.7942E+02	5.2221E+01	5.2284E+03	9.7625E+02	5.4685E+03	1.0673E+03
	F11		**F12**		**F13**	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	1.2740E+03	4.8137E+01	**2.1344E+06**	**2.0313E+06**	**3.6575E+04**	**4.4484E+04**
MVO	1.3492E+03	7.3312E+01	1.7126E+07	1.9260E+07	1.4184E+05	8.7242E+04
SCA	3.8055E+03	9.6441E+02	2.6769E+09	1.0277E+09	1.2457E+09	4.5075E+08
GWO	2.4925E+03	1.0861E+03	1.2794E+08	1.4104E+08	4.2027E+07	9.0808E+07
RIME	1.3430E+03	6.4494E+01	1.5445E+07	1.2556E+07	2.0367E+05	2.1819E+05
ALO	1.6072E+03	2.9033E+02	2.9105E+07	2.7789E+07	1.1196E+05	4.8421E+04
WOA	1.1173E+04	4.3832E+03	5.2975E+08	3.5710E+08	1.2064E+07	1.0695E+07
STOA	2.9989E+03	9.8445E+02	7.2588E+08	5.5008E+08	1.6408E+08	1.3323E+08
DO	**1.2507E+03**	**5.0890E+01**	1.0984E+07	6.2977E+06	7.8362E+04	3.7814E+04
BKA	1.4823E+03	2.7425E+02	3.4050E+08	1.4271E+09	1.4742E+08	5.6204E+08

Table 5. *Cont.*

	F14		F15		F16	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	**9.7122E+03**	**8.5039E+03**	**7.5731E+03**	**7.8311E+03**	2.6968E+03	3.2905E+02
MVO	3.4510E+04	2.7848E+04	6.9820E+04	5.2125E+04	2.8851E+03	2.6801E+02
SCA	9.2015E+05	7.0195E+05	6.0567E+07	4.9517E+07	4.2423E+03	3.1656E+02
GWO	5.3408E+05	6.3622E+05	1.1281E+06	1.8568E+06	**2.6205E+03**	**3.4678E+02**
RIME	9.1925E+04	5.6751E+04	1.7103E+04	1.1875E+04	2.6640E+03	3.4723E+02
ALO	3.3436E+05	3.7126E+05	5.0703E+04	4.6511E+04	3.1785E+03	3.0629E+02
WOA	2.5124E+06	2.1772E+06	6.4599E+06	9.1716E+06	4.4090E+03	4.8521E+02
STOA	8.1628E+05	8.1028E+05	2.9379E+07	3.0940E+07	3.1430E+03	3.3971E+02
DO	1.0419E+05	1.2960E+05	6.1593E+04	4.4668E+04	2.8216E+03	3.0688E+02
BKA	5.2316E+04	2.0990E+05	2.5011E+05	1.1217E+06	3.0542E+03	3.4306E+02
	F17		**F18**		**F19**	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	2.1869E+03	1.7382E+02	**1.7548E+05**	**1.8114E+05**	2.4472E+04	7.1039E+04
MVO	2.2538E+03	2.2586E+02	9.6241E+05	6.5250E+05	2.5265E+06	2.2711E+06
SCA	2.7904E+03	1.8499E+02	1.4122E+07	5.5519E+06	9.2923E+06	5.1945E+07
GWO	**2.0888E+03**	**1.4027E+02**	2.3394E+06	4.0224E+06	2.8350E+06	7.5725E+06
RIME	2.1895E+03	1.9572E+02	1.4803E+06	1.3970E+06	**2.2062E+04**	**1.8394E+04**
ALO	2.5163E+03	2.4436E+02	1.3267E+06	1.1844E+06	4.8498E+06	4.1078E+06
WOA	2.7627E+03	3.4280E+02	9.3162E+06	7.5396E+06	2.2632E+07	1.7285E+07
STOA	2.4367E+03	2.9294E+02	4.5471E+06	6.1653E+06	1.9592E+07	2.9041E+07
DO	2.2896E+03	2.5552E+02	1.3907E+06	1.7331E+06	1.6694E+05	1.6972E+05
BKA	2.3278E+03	2.4580E+02	8.7398E+05	3.0872E+06	4.1854E+05	8.5443E+05
	F20		**F21**		**F22**	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	2.5150E+03	2.1790E+02	2.4553E+03	5.3725E+01	**4.1723E+03**	**2.1115E+03**
MVO	2.5910E+03	2.3265E+02	2.4087E+03	2.4179E+01	5.6648E+03	1.5282E+03
SCA	2.9201E+03	1.3859E+02	2.6055E+03	3.3046E+01	9.7246E+03	1.6830E+03
GWO	**2.4847E+03**	**1.7324E+02**	**2.4064E+03**	**2.3014E+01**	5.1814E+03	2.1510E+03
RIME	2.6049E+03	2.2393E+02	2.4114E+03	3.1850E+01	4.9643E+03	1.8501E+03
ALO	2.7646E+03	2.0719E+02	2.4579E+03	3.6581E+01	5.4810E+03	2.1202E+03
WOA	2.9553E+03	2.5455E+02	2.6568E+03	6.8036E+01	8.2539E+03	1.9474E+03
STOA	2.8332E+03	1.9580E+02	2.5038E+03	2.4933E+01	8.7053E+03	1.2683E+03
DO	2.7178E+03	1.9581E+02	2.4670E+03	4.2004E+01	6.0385E+03	1.7841E+03
BKA	2.6231E+03	2.0607E+02	2.5513E+03	4.9758E+01	6.9847E+03	1.6217E+03
	F23		**F24**		**F25**	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	2.8503E+03	8.7286E+01	**2.9399E+03**	**3.6752E+01**	**2.9006E+03**	**2.3501E+01**
MVO	**2.7769E+03**	**3.9708E+01**	3.0205E+03	9.5203E+01	2.9148E+03	2.4447E+01
SCA	3.0867E+03	4.9441E+01	3.2546E+03	3.7694E+01	3.6973E+03	2.3620E+02
GWO	2.7832E+03	4.8088E+01	2.9813E+03	6.6322E+01	3.0294E+03	8.5565E+01
RIME	2.7931E+03	5.0184E+01	2.9554E+03	3.7237E+01	2.9272E+03	3.2391E+01
ALO	2.8834E+03	6.4267E+01	3.0330E+03	6.9358E+01	2.9724E+03	3.0384E+01
WOA	3.1521E+03	1.3010E+02	3.2817E+03	1.3662E+02	3.1981E+03	8.5220E+01
STOA	2.8996E+03	4.5481E+01	3.0322E+03	3.2068E+01	3.2076E+03	1.3902E+02
DO	2.9202E+03	8.0655E+01	3.0863E+03	5.7420E+01	2.9127E+03	1.8573E+01
BKA	3.1457E+03	1.9363E+02	3.2771E+03	1.1239E+02	3.1276E+03	2.1325E+02

Table 5. *Cont.*

	F26		F27		F28	
	Aver	**Std**	**Aver**	**Std**	**Aver**	**Std**
IBKA	5.6976E+03	1.6824E+03	3.2511E+03	2.6541E+01	**3.2634E+03**	**4.3976E+01**
MVO	4.9731E+03	6.8718E+02	**3.2362E+03**	**2.9722E+01**	3.2662E+03	3.2919E+01
SCA	7.9205E+03	4.1810E+02	3.5862E+03	9.2203E+01	4.5569E+03	3.3449E+02
GWO	5.0397E+03	4.9222E+02	3.2744E+03	4.0910E+01	3.5002E+03	1.5594E+02
RIME	**4.7702E+03**	**7.7115E+02**	3.2455E+03	1.7650E+01	3.2993E+03	4.2249E+01
ALO	5.6808E+03	9.2647E+02	3.4391E+03	1.1451E+02	3.3617E+03	3.9583E+01
WOA	8.6925E+03	1.2937E+03	3.5157E+03	1.6520E+02	3.8686E+03	2.4676E+02
STOA	6.1952E+03	4.3746E+02	3.3364E+03	5.8234E+01	5.1750E+03	1.3275E+03
DO	5.9975E+03	1.0339E+03	3.3035E+03	5.2064E+01	3.2731E+03	2.9028E+01
BKA	7.9174E+03	1.4532E+03	3.4438E+03	1.3972E+02	3.9948E+03	9.9809E+02
	F29		**F30**			
	Aver	**Std**	**Aver**	**Std**		
IBKA	4.0294E+03	2.1901E+02	**5.2818E+05**	**1.4079E+06**		
MVO	4.0716E+03	2.4553E+02	5.1215E+06	3.5195E+06		
SCA	5.2651E+03	3.0705E+02	1.8984E+08	5.5931E+07		
GWO	**3.8933E+03**	**1.6979E+02**	1.1620E+07	1.1248E+07		
RIME	4.0837E+03	2.2735E+02	6.2485E+05	5.5801E+05		
ALO	4.8292E+03	4.1913E+02	1.0533E+07	7.3270E+06		
WOA	5.3572E+03	5.6067E+02	7.7638E+07	8.0022E+07		
STOA	4.6813E+03	3.2943E+02	5.3184E+07	3.9854E+07		
DO	4.1918E+03	2.7007E+02	1.7537E+06	9.3124E+05		
BKA	4.7009E+03	3.7168E+02	3.5877E+07	9.0955E+07		

Based on Table 5, it is evident that IBKA ranks first among the 29 benchmark functions in the CEC2017 test set for the most part. All algorithms are ranked using the Friedman test method based on their fitness on the benchmark functions, as shown in Table 5. The results of the Friedman test are presented in Figure 5, where the lowest average rank indicates the best performance. Obviously, IBKA has the lowest average rank on the cec2017 test functions. It suggests that the optimization performance of IBKA outperforms the competing algorithms, whether on unimodal, multimodal, hybrid, or composition functions.

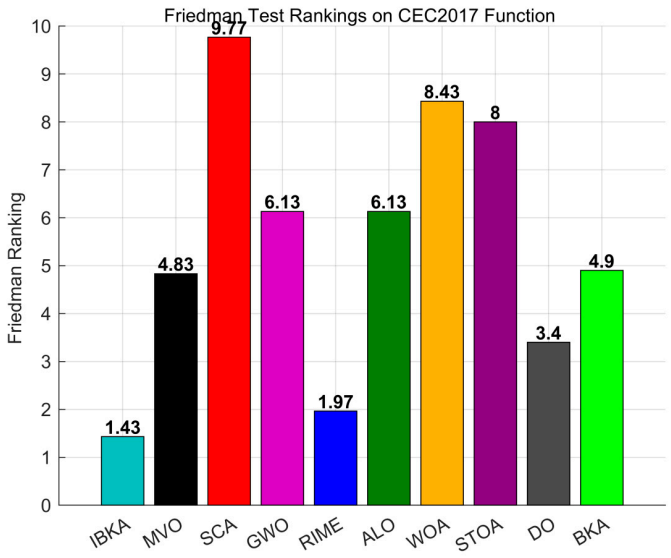


Figure 5. Friedman rankings of different algorithms on 29 benchmark functions.

Furthermore, the Wilcoxon rank-sum test is used to illustrate the significant differences between IBKA and the competing algorithms. Table 6 reports the *p*-values of the Wilcoxon rank-sum test, where a *p*-value less than 0.05 indicates that IBKA has a statistically significant advantage over its competitors (results with *p*-values greater than 0.05 are displayed in bold). Significant results are highlighted in bold. Obviously, most *p*-values are below 0.05, which indicates that the superiority of IBKA on the benchmark functions is statistically significant.

Table 6. *p*-values obtained from the Wilcoxon rank-sum test (F1–F30).

Fun	IBKA vs. MVO	IBKA vs. SCA	IBKA vs. GWO	IBKA vs. RIME	IBKA vs. ALO	IBKA vs. WOA	IBKA vs. STOA	IBKA vs. DO	IBKA vs. BKA
F1	3.01E−11	3.01E−11	3.01E−11	3.01E−11	5.87E−04	3.01E−11	3.01E−11	3.01E−11	3.01E−11
F3	3.33E−11	3.01E−11	3.01E−11	3.01E−11	3.01E−11	3.01E−11	3.01E−11	3.01E−11	3.01E−11
F4	3.77E−04	3.01E−11	5.18E−07	9.70E−01	9.06E−03	3.01E−11	3.33E−11	3.71E−01	1.46E−10
F5	8.66E−05	3.01E−11	3.14E−02	2.38E−04	5.99E−01	4.07E−11	8.19E−07	1.76E−02	8.48E−09
F6	8.76E−01	7.38E−10	4.18E−09	1.01E−08	5.09E−06	4.07E−11	6.28E−06	2.15E−03	3.49E−09
F7	3.49E−09	2.22E−09	1.49E−06	1.07E−09	3.56E−04	1.20E−10	4.11E−06	4.20E−01	1.15E−07
F8	1.45E−01	3.01E−11	8.31E−03	1.37E−01	2.32E−02	3.01E−11	3.35E−08	3.87E−01	3.52E−07
F9	7.65E−05	1.20E−10	3.18E−04	4.42E−03	1.18E−01	4.97E−11	6.04E−07	2.59E−05	9.51E−06
F10	7.28E−01	3.01E−11	1.76E−01	3.04E−01	1.83E−02	8.99E−11	5.49E−11	9.62E−02	4.22E−03
F11	1.10E−06	3.01E−11	3.01E−11	1.24E−04	6.12E−10	3.01E−11	3.01E−11	2.92E−02	3.35E−08
F12	1.58E−04	3.01E−11	2.37E−10	6.76E−05	2.57E−07	3.01E−11	3.01E−11	3.83E−05	6.52E−07
F13	2.78E−07	3.01E−11	4.61E−10	1.38E−06	3.35E−08	3.01E−11	3.01E−11	6.52E−07	5.46E−09
F14	3.25E−07	3.01E−11	6.69E−11	6.72E−10	8.15E−11	3.01E−11	3.01E−11	2.37E−10	1.95E−01
F15	1.07E−09	3.01E−11	4.19E−10	2.53E−04	3.96E−08	3.01E−11	3.33E−11	7.69E−08	2.87E−06
F16	6.52E−01	5.49E−11	8.23E−02	7.95E−01	5.09E−06	4.07E−11	3.25E−07	1.76E−02	2.26E−03
F17	6.30E−01	1.10E−08	3.51E−02	3.25E−01	3.00E−04	1.60E−06	5.08E−03	1.80E−01	1.22E−02
F18	5.96E−09	3.01E−11	3.49E−09	8.89E−10	4.31E−08	4.97E−11	3.33E−11	3.49E−09	1.45E−01
F19	3.01E−11	3.01E−11	1.07E−09	1.17E−03	3.01E−11	3.01E−11	3.01E−11	2.66E−09	4.19E−10
F20	8.41E−01	1.54E−09	7.48E−02	6.52E−01	6.35E−05	1.10E−08	2.12E−04	1.91E−02	7.39E−01
F21	6.66E−03	3.15E−10	4.63E−03	5.82E−03	9.70E−01	8.15E−11	9.79E−05	1.76E−02	5.09E−08
F22	6.14E−02	4.18E−09	1.85E−01	1.95E−01	5.55E−02	1.06E−07	3.15E−10	4.84E−02	6.09E−03
F23	9.51E−06	1.46E−10	2.49E−03	1.32E−02	1.27E−02	4.50E−11	4.08E−05	1.04E−04	3.15E−10
F24	3.15E−05	3.68E−11	2.23E−02	1.37E−03	2.28E−01	1.61E−10	1.02E−01	6.54E−04	2.66E−09
F25	8.23E−02	3.01E−11	4.97E−11	8.18E−01	2.13E−05	3.01E−11	3.01E−11	3.32E−01	3.68E−11
F26	2.17E−01	9.75E−10	1.29E−01	2.28E−01	9.88E−03	2.22E−09	1.07E−02	2.32E−02	9.06E−08
F27	4.05E−02	3.82E−10	3.18E−03	8.88E−01	1.41E−09	4.19E−10	6.52E−07	7.69E−04	3.49E−09
F28	6.20E−01	3.01E−11	6.06E−11	3.67E−03	2.37E−07	3.01E−11	3.01E−11	1.80E−01	6.69E−11
F29	2.51E−01	3.33E−11	1.85E−03	7.06E−01	9.53E−07	5.49E−11	9.06E−08	7.28E−01	6.52E−07
F30	3.01E−11	3.01E−11	3.68E−11	4.99E−09	3.33E−11	3.01E−11	3.01E−11	1.77E−10	3.68E−11

The experimental results demonstrate that IBKA consistently achieves solutions closest to the theoretical optimum on most benchmark functions. To further illustrate its convergence behavior, Figure 6 shows the convergence curves of IBKA and the compared algorithms on six representative functions (F3, F10, F15, F18, F22, and F30) with a dimension of 30. For F3, IBKA performs similarly to other algorithms in the early stages but converges faster and continues improving toward the end. On F10, IBKA converges more slowly than RIME but achieves higher final accuracy than BKA. For F15, although initially inferior to BKA, IBKA surpasses it as iterations progress. On F18, IBKA maintains both high accuracy and fast convergence throughout. For F22, IBKA outperforms all others in convergence speed and final precision. On F30, its early performance is comparable to BKA, but the final accuracy is better. These results confirm that IBKA avoids premature convergence and demonstrates strong global exploration and local exploitation capabilities.

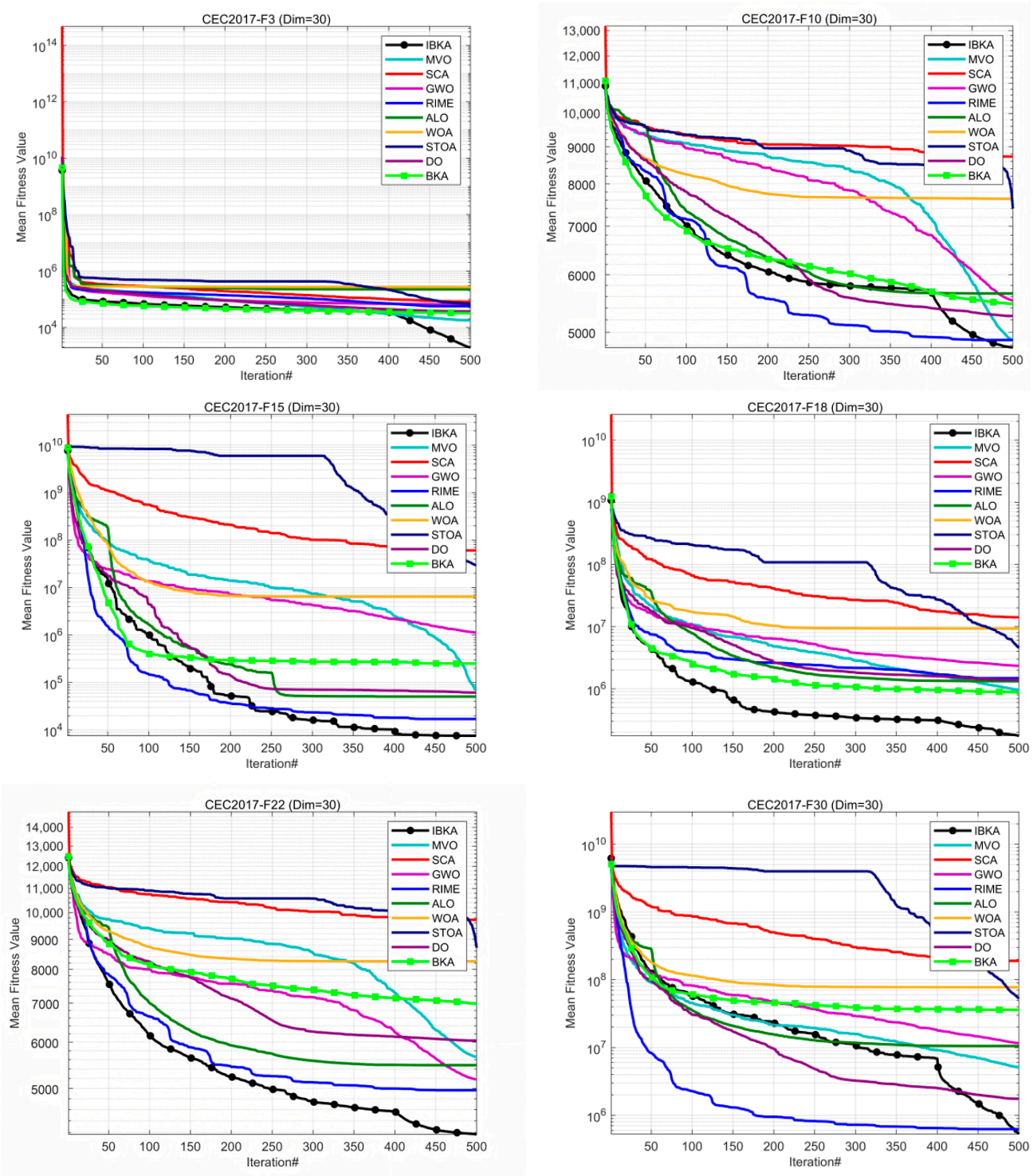


Figure 6. Comparison of Fitness Curves Among Algorithms During Iteration.

3.2. VMD-IBKA-BiLSTM Framework for CO_2 Emission Forecasting

Previously, experiments on benchmark functions were completed. In this section, we apply the proposed IBKA algorithm to practical optimization tasks, constructing a CO_2 emission prediction framework named VMD-IBKA-BiLSTM. Within this framework, BiLSTM serves as the deep learning model for CO_2 emission prediction, while IBKA is utilized to optimize five hyperparameters of the BiLSTM.

In hyperparameter optimization for deep learning models, the optimization space is characterized by its combinatorial and non-convex nature; traditional grid search, bayesian

optimization, and random search methods have limitations when dealing with such problems. The IBKA algorithm proposed in this study is particularly suitable for deep learning model optimization. The key reason is that the nonlinear simplex strategy avoids premature convergence to suboptimal solutions through continuous searching during migration behavior. Additionally, the prey escapes and population cooperation strategies based on Lévy flight behavior in the attack behavior exhibit the capability to enhance the algorithm's search range and avoid falling into local optimal solutions. The BKA algorithm, which combines these two strategies, can accelerate the convergence of the model training process and find the global optimal solution.

3.2.1. CO_2 Emission Data

The dataset used in this work is the CO_2 emissions from various departments in China from 2019 to June 2023. The experiment selected CO_2 emissions data from four major sectors: domestic aviation, international aviation, industry, and residents. The time range of the data is from 1 January 2019 to 31 June 2023. During this period, each sector contains 1612 sampling points. Figure 7 shows all the data used in this section. The x-axis represents the date sequence starting from 1 January 2019, and the y-axis represents the amount of carbon dioxide emissions measured in millions of tons (MM·T−1).

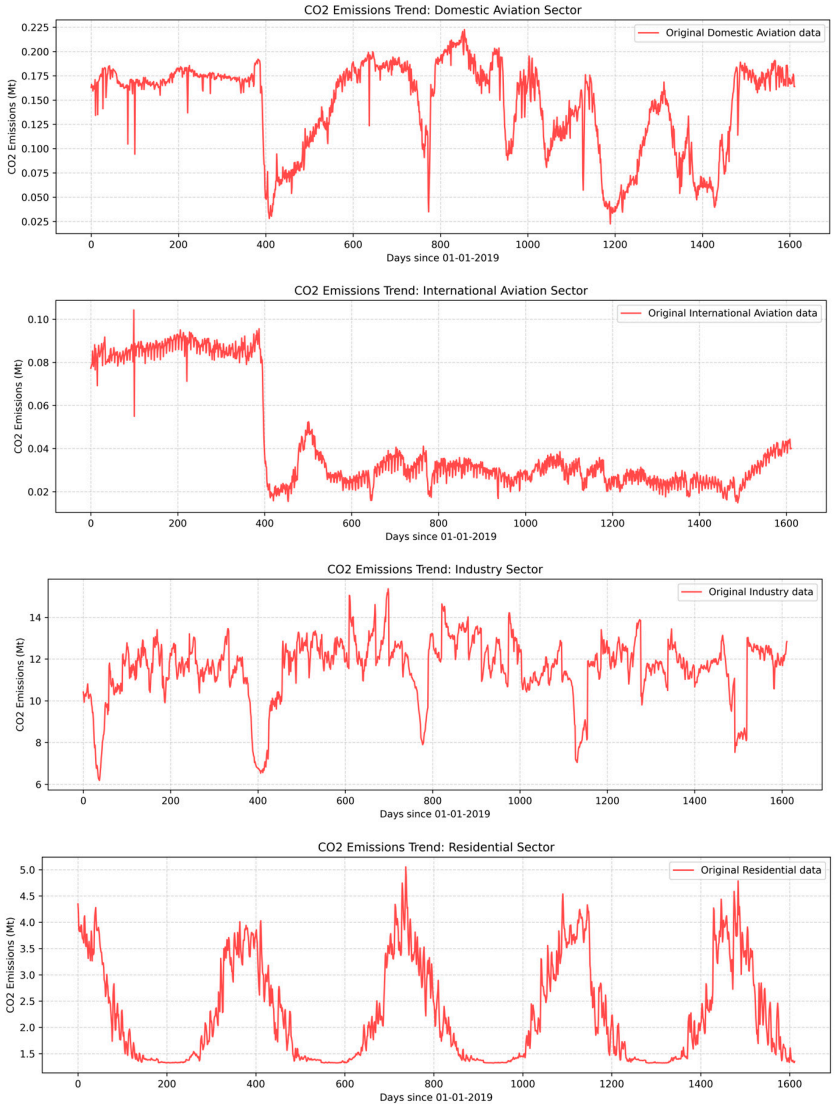


Figure 7. CO_2 emissions from different sectors.

3.2.2. VMD Parameters

The VMD algorithm is utilized to decompose complex nonlinear and non-stationary carbon dioxide data into a series of Intrinsic Mode Function (IMF) components with clear physical significance. These components reflect the intrinsic characteristics of the data and effectively separate the high-frequency and low-frequency information of carbon dioxide emissions. Accurately extracting these features helps to construct more accurate prediction models. VMD exhibits the capability to adaptively match the optimal central frequency and bandwidth for each mode, precisely capturing the dynamic characteristics of the data. It is crucial for revealing the essential characteristics of the data and lays a solid foundation for subsequent predictions. The parameter settings for VMD are shown in Table 7:

Table 7. VMD Parameters.

Mode Number K	Penalty Factor α	Noise Tolerance τ	Convergence Tolerance tol	DC Component
8	1800	0	1E−7	0

3.2.3. Sample Making

In this study, we adopt historical data from a continuous 12-day period to predict the next day's carbon dioxide emissions. The data samples are divided using a sliding window method, where each sample consists of 12 data points as input and 1 data point as output (as shown in Figure 8), with green data points representing the input and red data points representing the output. We obtained a total of 1600 training samples. The first 80% of the samples are used as the training set, and the remaining 20% as the test set. For model hyperparameter tuning, we perform cross-validation on the training set, with each fold using 20% of the training data as the validation subset. The final model is evaluated independently on the test set.

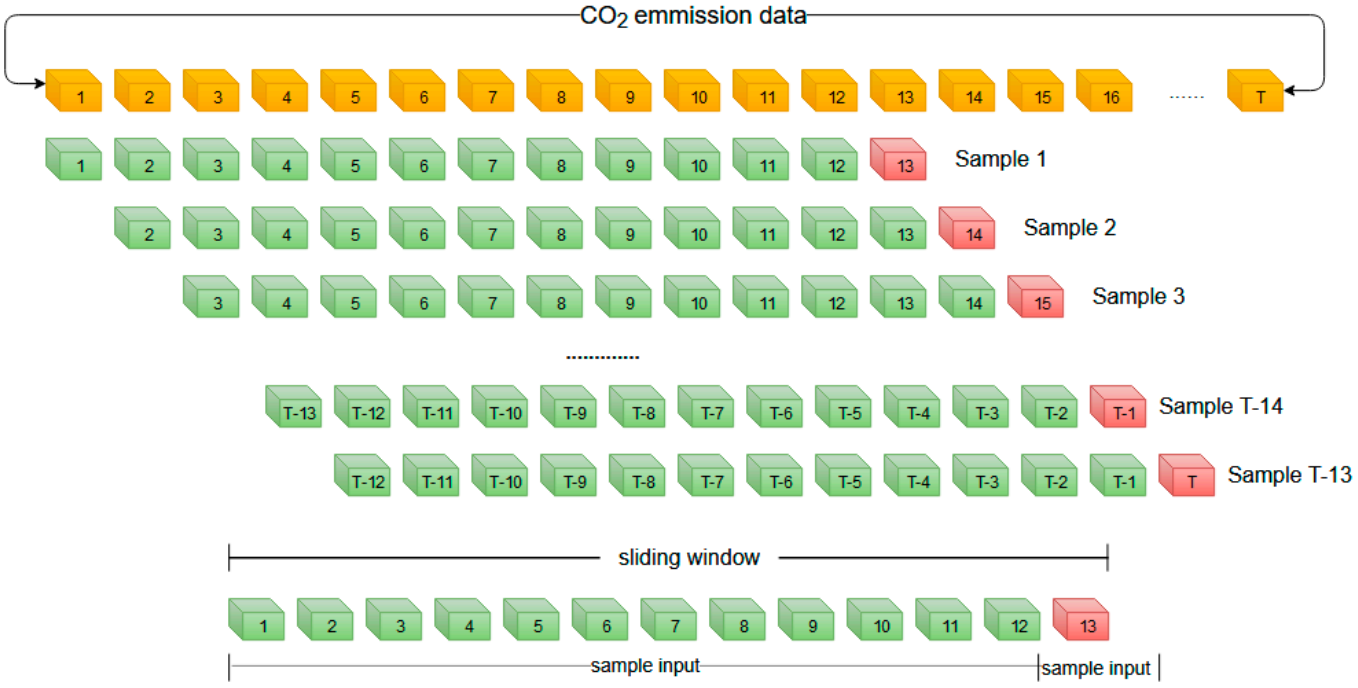


Figure 8. Sample making map.

3.2.4. VMD-IBKA-BiLSTM Flowchart

When utilizing the BiLSTM model for CO_2 emission prediction, the following three hyperparameters have the most significant impact on its predictive performance, including (1) the number of units in each BiLSTM layer (unit); (2) the parameter update step size during model training (learning_rate); (3) the maximum number of training epochs for the model (max_epochs).

The improved IBKA is employed to optimize the aforementioned three hyperparameters. First, it is necessary to define the search space boundaries for each hyperparameter with specific value ranges, as shown in Table 8.

Table 8. The value range of the hyperparameter.

Hyperparameter Name	Description	Lower Bounds	Upper Bounds
unit	The number of units in the BiLSTM layer	50	300
learning_rate	Parameter update step during model training	0.001	0.01
max_epochs	Maximum number of cycles for model training	50	300

Secondly, initialize IBKA. Among them, the maximum number of iterations is 200, the dimension d is 3, and the population size n is 30. Then, the loss of BiLSTM is used as the fitness function (the loss function is RMSE in the study). The solver of BiLSTM is set to AdaGrad. Finally, the IBKA algorithm is used to search for the optimal hyperparameters of BiLSTM. Figure 9 shows the flowchart of the entire VMD-IBKA-BiLSTM framework.

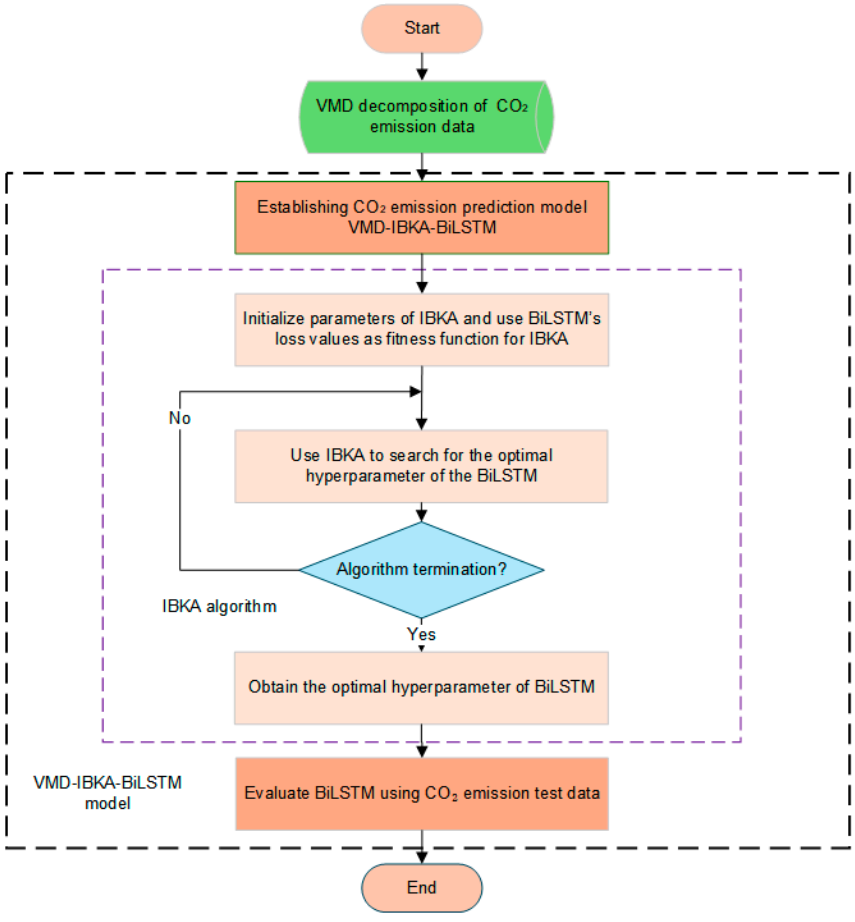


Figure 9. VMD-IBKA-BiLSTM framework.

3.2.5. Evaluation Metrics

Three indicators are adopted to evaluate model performance: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE) in this section. The calculation formulas are as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{28}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{29}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{y_i}\right) \times 100\% \tag{30}$$

where n is the number of test samples, is the actual measured value of the sample data, is the predicted value of the sample data, and is the average measured value of the sample data.

3.2.6. Comparison of BiLSTM Optimized by Various Algorithms

In addition to IBKA, we also employed 12 optimization algorithms to optimize our proposed CO_2 emission prediction model, BiLSTM, and compared the optimization performance of different algorithms in this section. These comparative algorithms include three traditional optimization algorithms (namely Grid Search, Random Search, and Bayesian Optimization Algorithm), eight state-of-the-art swarm intelligence optimization algorithms (namely MVO, SCA, GWO, RIME, ALO, WOA, STOA, DO), and the original BKA algorithm. To reduce the error caused by experimental randomness, each set of comparative experiments was repeated 30 times.

The study utilizes CO_2 emission data from four Chinese government agencies, with results displayed in Tables 9–12. Among them, the best results are displayed in bold. The terms unit, lr, and mp represent the optimal hyperparameters obtained through algorithmic optimization, corresponding to the number of network units, learning rate, and maximum training epochs, respectively, as defined in Table 8.

Table 9. Comparison of various models in the domestic aviation sector.

Optimization Algorithm	Optimal Solution from the Optimization Algorithm			Evaluation Indicators		
	unit	lr	mp	MAE (MM·T−1)	RMSE (MM·T−1)	MAPE
Grid Search	300	0.0090	50	0.0029	0.0039	1.96%
Random Search	81	0.0094	157	0.0030	0.0041	2.09%
Bayesian	73	0.0061	300	0.0028	0.0037	1.93%
MVO	283	0.0091	127	0.0027	0.0036	1.88%
SCA	185	0.0058	299	0.0026	0.0035	1.83%
GWO	151	0.0024	298	0.0028	0.0037	1.90%
RIME	158	0.0076	299	0.0025	0.0035	1.74%
ALO	298	0.0046	300	0.0027	0.0036	1.86%
WOA	151	0.0099	296	0.0028	0.0040	1.90%
STOA	92	0.0034	300	0.0027	0.0035	1.84%
DO	80	0.0046	299	0.0027	0.0035	1.86%
BKA	117	0.0065	298	0.0026	0.0037	1.83%
IBKA	**126**	**0.0082**	**296**	**0.0023**	**0.0033**	**1.59%**

Table 10. Comparison of various models in the international aviation sector.

Optimization Algorithm	Optimal Solution from the Optimization Algorithm			Evaluation Indicators		
	unit	lr	mp	MAE (MM·T−1)	RMSE (MM·T−1)	MAPE
Grid Search	224	0.0049	263	0.0019	0.0022	6.03%
Random Search	259	0.0031	294	0.0016	0.0019	5.52%
Bayesian	109	0.0027	237	0.0014	0.0017	5.02%
MVO	55	0.0015	256	0.0008	0.0010	2.86%
SCA	203	0.0015	67	0.0011	0.0013	3.90%
GWO	89	0.0029	268	0.0010	0.0012	3.41%
RIME	67	0.0010	297	0.0008	0.0009	2.71%
ALO	54	0.0019	275	0.0009	0.0011	3.31%
WOA	51	0.0033	278	0.0011	0.0013	3.86%
STOA	56	0.0023	268	0.0009	0.0011	3.21%
DO	259	0.0030	294	0.0012	0.0015	3.95%
BKA	55	0.0012	272	0.0008	0.0010	2.81%
IBKA	**124**	**0.0011**	**221**	**0.0007**	**0.0008**	**2.41%**

Table 11. Comparison of various models in the industry sector.

Optimization Algorithm	Optimal Solution from the Optimization Algorithm			Evaluation Indicators		
	unit	lr	mp	MAE (MM·T−1)	RMSE (MM·T−1)	MAPE
Grid Search	300	0.0060	50	0.1075	0.1428	0.89%
Random Search	82	0.0011	300	0.0936	0.1260	0.79%
Bayesian	51	0.0049	240	0.0911	0.1227	0.76%
MVO	208	0.0035	282	0.0856	0.1194	0.72%
SCA	264	0.0099	127	0.0903	0.1223	0.75%
GWO	56	0.0062	300	0.0857	0.1208	0.72%
RIME	149	0.0045	299	0.0828	0.1181	0.71%
ALO	187	0.0016	281	0.0831	0.1126	0.70%
WOA	120	0.0032	256	0.0845	0.1132	0.72%
STOA	77	0.0024	265	0.0896	0.1175	0.72%
DO	180	0.0017	118	0.0865	0.1137	0.75%
BKA	125	0.0099	299	0.0839	0.1148	0.71%
IBKA	**300**	**0.0100**	**300**	**0.0816**	**0.1117**	**0.68%**

Table 12. Comparison of various models in the resident sector.

Optimization Algorithm	Optimal Solution from the Optimization Algorithm			Evaluation Indicators		
	unit	lr	mp	MAE (MM·T−1)	RMSE (MM·T−1)	MAPE
Grid Search	300	0.0040	50	0.0749	0.0765	3.87%
Random Search	291	0.0069	135	0.0643	0.0830	3.24%
Bayesian	262	0.0038	67	0.0542	0.7653	2.41%
MVO	279	0.0096	227	0.0534	0.0741	2.15%
SCA	248	0.0024	58	0.0736	0.1014	3.22%
GWO	143	0.0045	299	0.0463	0.0635	2.01%
RIME	227	0.0031	282	0.0422	0.0580	1.85%
ALO	243	0.0011	300	0.0452	0.0642	2.04%
WOA	54	0.0096	72	0.0595	0.0594	2.47%
STOA	56	0.0032	243	0.0469	0.0650	2.04%
DO	52	0.0049	268	0.0456	0.0638	1.91%
BKA	300	0.0100	300	0.0552	0.0739	2.30%
IBKA	**174**	**0.0035**	**283**	**0.0385**	**0.0550**	**1.76%**

According to the results in Tables 9–12, it is evident that the performance of traditional optimization algorithms (such as grid search, random search, and Bayesian optimization)

is significantly lower than that of swarm intelligence optimization algorithms. Among the 10 swarm intelligence optimization algorithms, our improved IBKA achieved the best performance.

Furthermore, different optimization algorithms significantly enhance the performance of the same prediction model, BiLSTM. This indicates that model optimization plays a crucial role and may be even more critical than model selection in deep learning applications.

3.2.7. Comparison with Other Models

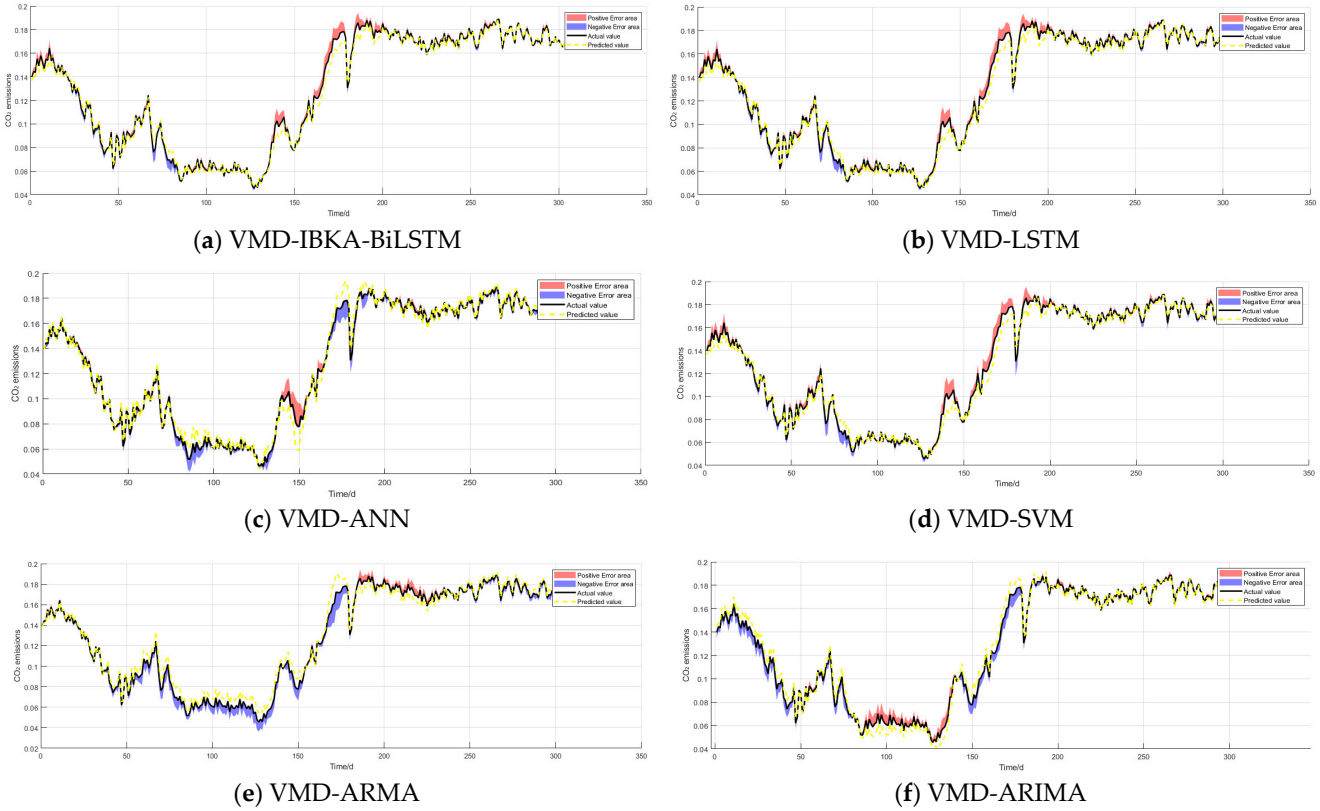
In the previous section, the performance of BiLSTM was analyzed under different optimization algorithms, and it was found that the model optimized by IBKA performed the best. In this section, we compare the optimized BiLSTM with five other machine learning and deep learning models, including SVM, ANN, and LSTM, as well as with the two statistical models, ARMA and ARIMA.

To ensure fairness, all models were optimized using IBKA. Each experiment was repeated 30 times, and Table 13 summarizes the average evaluation metric values from the 30 experiments. Table 13 also displays the experimental results of each model (with the best results in bold), where Rank indicates the final ranking of the model. The results demonstrate that BiLSTM significantly outperforms the comparative models in all four evaluation domains.

Table 13. Comparison of different models in four sectors.

Sector	Model	MAE	RMSE	MAPE	Rank
		Aver (30 Times)			
Aviation (domestic aviation)	ARMA	0.0048	0.0065	3.1701%	6
	ARIMA	0.0039	0.0050	2.3482%	5
	SVM	0.0032	0.0047	2.2108%	4
	ANN	0.0031	0.0046	1.9845%	2
	LSTM	0.0029	0.0040	2.0343%	3
	BiLSTM	**0.0025**	**0.0034**	**1.5995%**	**1**
Aviation (international aviation)	ARMA	0.0015	0.0018	5.2502%	6
	ARIMA	0.0009	0.0010	3.0177%	4
	SVM	0.0008	0.0009	2.7802%	3
	ANN	0.0011	0.0013	3.7620%	5
	LSTM	0.0008	0.0009	2.6425%	2
	BiLSTM	**0.0007**	**0.0008**	**2.4182%**	**1**
Industry	ARMA	0.2103	0.2814	1.7709%	6
	ARIMA	0.1423	0.2286	1.2075%	5
	SVM	0.0901	0.1346	0.7571%	3
	ANN	0.1297	0.1994	1.1108%	4
	LSTM	0.0865	0.1198	0.7220%	2
	BiLSTM	**0.0833**	**0.1149**	**0.6979%**	**1**
Resident	ARMA	0.0977	0.1323	4.1540%	6
	ARIMA	0.0789	0.1064	3.3183%	5
	SVM	0.0559	0.0757	2.3712%	4
	ANN	0.0554	0.0722	2.2670%	3
	LSTM	0.0462	0.0633	1.9967%	2
	BiLSTM	**0.0392**	**0.0554**	**1.7638%**	**1**

Figures 10–13 visually display the comparison between the predicted values and actual values of various models in the test set, covering four different domains. Here, the x-axis represents time, and the y-axis represents CO_2 emissions. From these charts, it is evident that the proposed BiLSTM model has the best predictive performance, with the most minor error, further proving the superiority of this model.

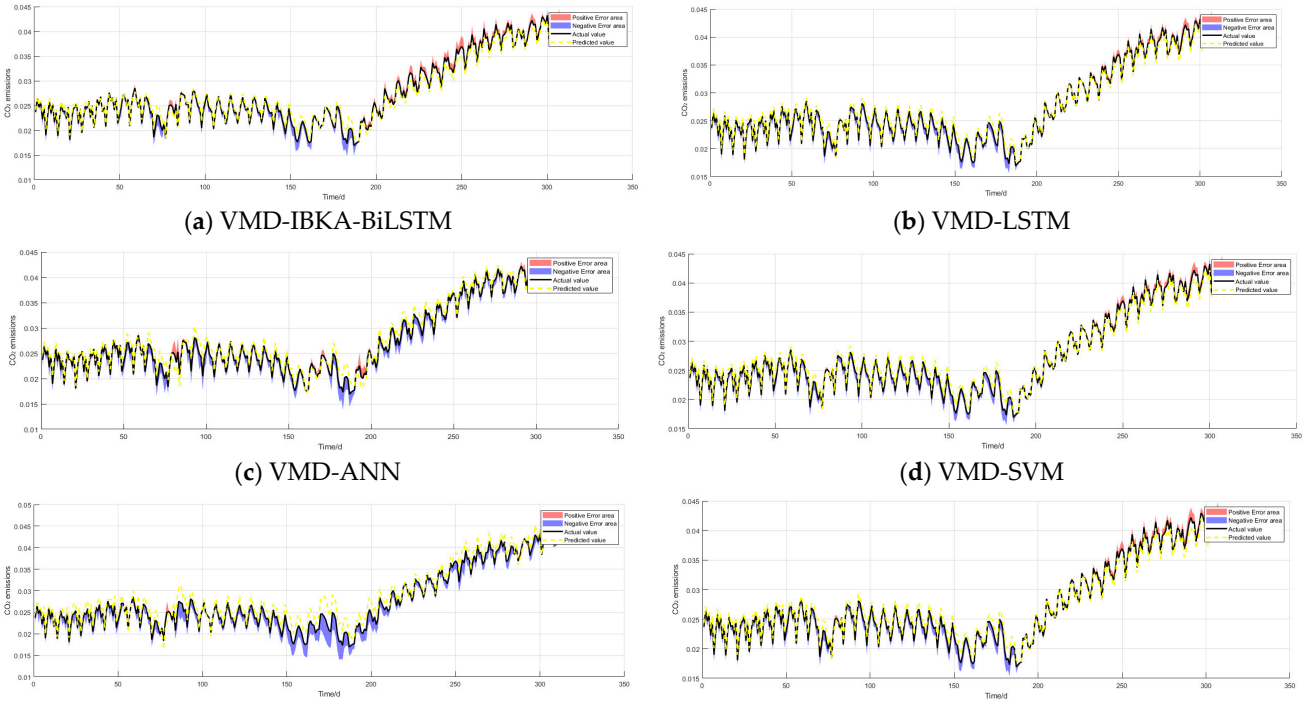


(**a**) VMD-IBKA-BiLSTM (**b**) VMD-LSTM

(**c**) VMD-ANN (**d**) VMD-SVM

(**e**) VMD-ARMA (**f**) VMD-ARIMA

Figure 10. Error curves of different models in the domestic aviation sector.

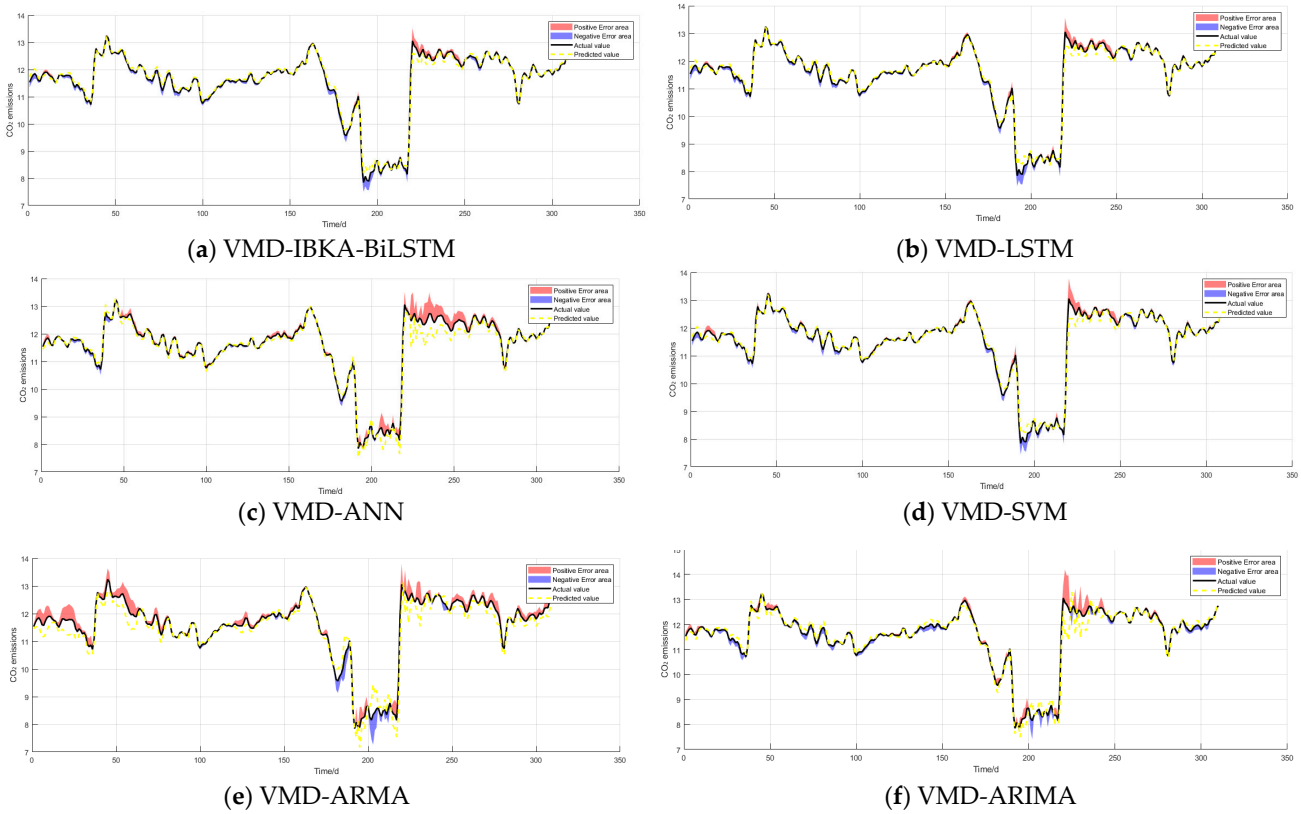


(**a**) VMD-IBKA-BiLSTM (**b**) VMD-LSTM

(**c**) VMD-ANN (**d**) VMD-SVM

(**e**) VMD-ARMA (**f**) VMD-ARIMA

Figure 11. Error curves of different models in the international aviation sector.

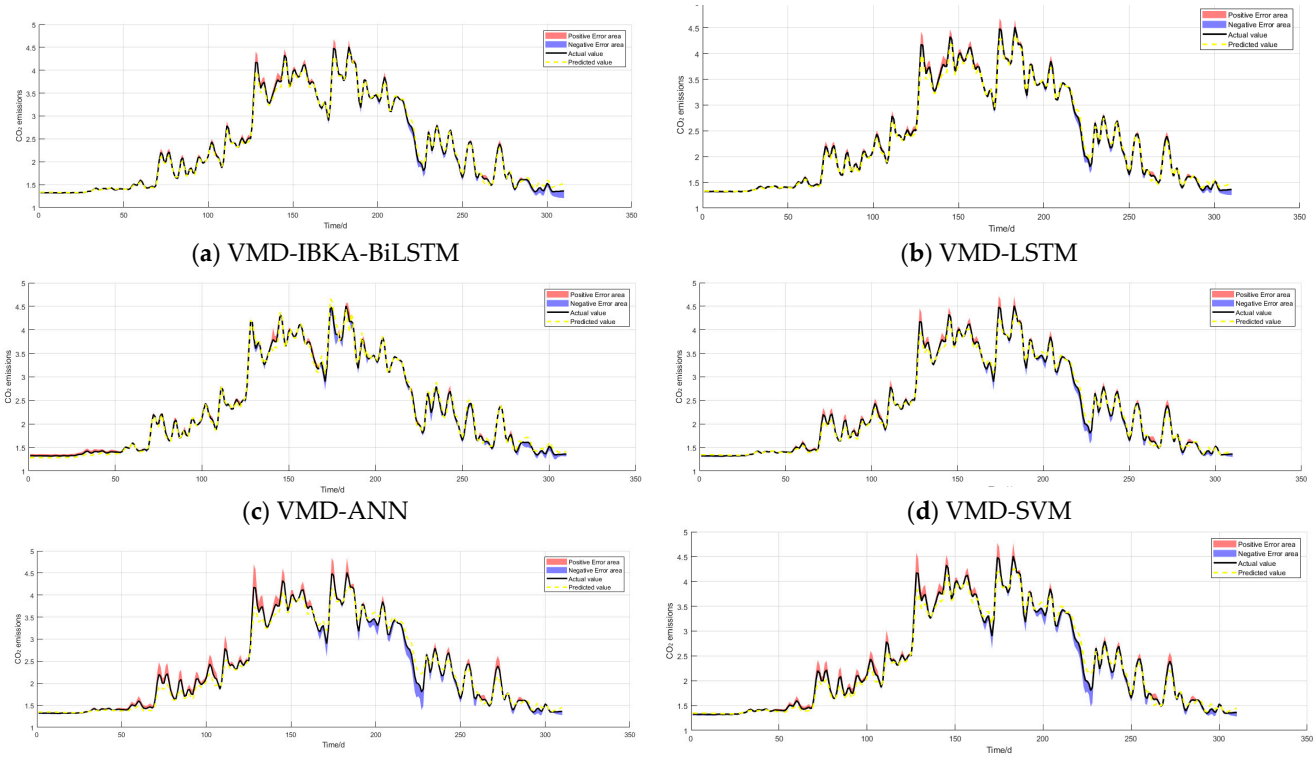


(**a**) VMD-IBKA-BiLSTM (**b**) VMD-LSTM

(**c**) VMD-ANN (**d**) VMD-SVM

(**e**) VMD-ARMA (**f**) VMD-ARIMA

Figure 12. Error curves of different models in the industry sector.



(**a**) VMD-IBKA-BiLSTM (**b**) VMD-LSTM

(**c**) VMD-ANN (**d**) VMD-SVM

(**e**) VMD-ARMA (**f**) VMD-ARIMA

Figure 13. Error curves of different models in the resident sector.

4. Conclusions

In the context of "dual carbon" (carbon peak and carbon neutrality), accurate CO_2 emission forecasting plays a crucial role in formulating emission reduction strategies and addressing climate change. To tackle this issue, this study proposes an innovative deep learning framework—VMD-IBKA-BiLSTM—to enhance the accuracy of CO_2 emission forecasting. The framework integrates the BiLSTM model with the IBKA optimization algorithm to improve the predictive ability for complex carbon emission data. Initially, experiments are conducted on 29 benchmark test functions to verify the performance of the IBKA optimization algorithm on different datasets. Through ablation experiments and comparison with nine mainstream metaheuristic optimization algorithms, the experimental results indicate that IBKA outperforms existing methods in all 29 benchmark tasks, demonstrating its advantages in nonlinear and high-complexity environments. Subsequently, the BiLSTM model is used to predict the CO_2 emission trends of four industries in China and compare the performance of IBKA with other optimization algorithms (including nine metaheuristic optimization algorithms and three traditional optimization algorithms) during this process. The results show that the BiLSTM optimized by IBKA exhibited the best predictive effect in all experiments, significantly improving the accuracy of CO_2 emission forecasting. Furthermore, we also compared the optimized BiLSTM with five mainstream machine learning and deep learning models, namely ARMA, ARIMA, ANN, SVM, and LSTM. The result indicates that BiLSTM exhibits the best performance in predicting all industries. In summary, the VMD-IBKA-BiLSTM framework demonstrates outstanding performance in dealing with complex carbon emission data, providing strong technical support for China's carbon emission monitoring and policymaking. This study not only provides a scientific approach to achieving Chinese "dual carbon" goals but also contributes a scalable Chinese solution to global climate governance, promoting the construction of an ecological civilization system and aiding in achieving the sustainable development vision of harmonious coexistence between humanity and nature.

Author Contributions: Conceptualization, J.G.; data curation, Y.Y. and H.L.; formal analysis, Y.Y., S.L. and J.G.; funding acquisition, Y.Y.; investigation, S.L.; methodology, H.L.; project administration, J.G.; resources, S.L.; software, S.L.; supervision, Y.Y. and J.G; validation, Y.Y., S.L. and H.L.; visualization, S.L.; writing—original draft, Y.Y., S.L. and H.L.; writing—review and editing, Y.Y., S.L. and H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to express their sincere gratitude to Yang Yueqiao from the Institute of Disaster Prevention Science and Technology for her valuable assistance in language editing and research guidance. Additionally, we would like to thank Liu Haijun and Guo Jidong for their support and help in data analysis, experimental design, and other aspects. We also appreciate all the individuals involved in this research and the peer experts who provided valuable suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Alizadeh, O. A review of ENSO teleconnections at present and under future global warming. *Wiley Interdiscip. Rev. Clim. Change* **2024**, *15*, e861. [CrossRef]
2. Mehmood, K.; Tauseef Hassan, S.; Qiu, X.; Ali, S. Comparative analysis of CO_2 emissions and economic performance in the United States and China: Navigating sustainable development in the climate change era. *Geosci. Front.* **2024**, *15*, 101843. [CrossRef]
3. Zhang, Z.; Wang, G. China's role in global climate governance. *Clim. Policy* **2017**, *17* (Suppl. S1), 32–47.
4. IEA. *An Energy Sector Roadmap to Carbon Neutrality in China. Energy Technology Policy Division*; IEA: Paris, France, 2021.

5. MEE. *List of Chemicals Under Priority Control (Second Batch)*; Ministry of Ecology and Environment of the People's Republic of China: Beijing, China, 2020.
6. Benoit, K. *Linear Regression Models with Logarithmic Transformations*; London School of Economics: London, UK, 2011; Volume 22, pp. 23–36.
7. De Livera, A.M.; Hyndman, R.J.; Snyder, R.D. Forecasting time series with complex seasonal patterns using exponential smoothing. *J. Am. Stat. Assoc.* **2011**, *106*, 1513–1527. [CrossRef]
8. Ge, H.; Chen, G.; Yu, H.; Chen, H.; An, F. Theoretical analysis of empirical mode decomposition. *Symmetry* **2018**, *10*, 623. [CrossRef]
9. Faysal, A.; Ngui, W.K.; Lim, M.H. Noise eliminated ensemble empirical mode decomposition for bearing fault diagnosis. *J. Vib. Eng. Technol.* **2021**, *9*, 2229–2245. [CrossRef]
10. Dragomiretskiy, K.; Zosso, D. Variational mode decomposition. *IEEE Trans. Signal Process.* **2013**, *62*, 531–544. [CrossRef]
11. Ahmadi, F.; Tohidi, M.; Sadrianzade, M. Streamflow prediction using a hybrid methodology based on variational mode decomposition (VMD) and machine learning approaches. *Appl. Water Sci.* **2023**, *13*, 135. [CrossRef]
12. Qin, Y.; Zhao, M.; Lin, Q.; Li, X.; Ji, J. Data-driven building energy consumption prediction model based on VMD-SA-DBN. *Mathematics* **2022**, *10*, 3058. [CrossRef]
13. Lotfalipour, M.R.; Falahi, M.A.; Bastam, M. Prediction of CO_2 emissions in Iran using grey and ARIMA models. *Int. J. Energy Econ. Policy* **2013**, *3*, 229–237.
14. Borisova, D.; Kostadinova, G.; Petkov, G.; Dospatliev, L.; Ivanova, M.; Dermendzhieva, D.; Beev, G. Assessment of CH_4 and CO_2 Emissions from a Gas Collection System of a Regional Non-Hazardous Waste Landfill, Harmanli, Bulgaria, Using the Interrupted Time Series ARMA Model. *Atmosphere* **2023**, *14*, 1089. [CrossRef]
15. Wen, L.; Cao, Y. Influencing factors analysis and forecasting of residential energy-related CO_2 emissions utilizing optimized support vector machine. *J. Clean. Prod.* **2020**, *250*, 119492. [CrossRef]
16. Sun, W.; Ren, C. Short-term prediction of carbon emissions based on the EEMD-PSOBP model. *Environ. Sci. Pollut. Res.* **2021**, *28*, 56580–56594. [CrossRef] [PubMed]
17. Wang, H.; Liang, W.; Liang, S.; Chen, B. Research on Carbon Dioxide Concentration Prediction Based on RNN Model in Deep Learning. *Highlights Sci. Eng. Technol.* **2023**, *48*, 281–287. [CrossRef]
18. Singh, M.; Dubey, R.K. Deep learning model based CO_2 emissions prediction using vehicle telematics sensors data. *IEEE Trans. Intell. Veh.* **2021**, *8*, 768–777. [CrossRef]
19. Kumari, S.; Singh, S.K. Machine learning-based time series models for effective CO_2 emission prediction in India. *Environ. Sci. Pollut. Res.* **2023**, *30*, 116601–116616. [CrossRef]
20. Xie, C.; Huang, C.; Zhang, D.; He, W. BiLSTM-I: A deep learning-based long interval gap-filling method for meteorological observation data. *Int. J. Environ. Res. Public Health* **2021**, *18*, 10321. [CrossRef]
21. Siami-Namini, S.; Tavakoli, N.; Namin, A.S. The performance of LSTM and BiLSTM in forecasting time series. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019; IEEE: Los Angeles, CA, USA, 2019; pp. 3285–3292.
22. Siami-Namini, S.; Tavakoli, N.; Namin, A.S. A comparative analysis of forecasting financial time series using arima, lstm, and bilstm. *arXiv* **2019**, arXiv:1911.09512.
23. Khalid, R.; Javaid, N. A survey on hyperparameters optimization algorithms of forecasting models in smart grid. *Sustain. Cities Soc.* **2020**, *61*, 102275. [CrossRef]
24. Bergstra, J.; Yamins, D.; Cox, D. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In Proceedings of the International Conference on Machine Learning, PMLR, Atlanta, GA, USA, 17–19 June 2013; pp. 115–123.
25. Probst, P.; Boulesteix, A.L.; Bischl, B. Tunability: Importance of hyperparameters of machine learning algorithms. *J. Mach. Learn. Res.* **2019**, *20*, 1934–1965.
26. Bergstra, J.; Komer, B.; Eliasmith, C.; Yamins, D.; Cox, D.D. Hyperopt: A python library for model selection and hyperparameter optimization. *Comput. Sci. Discov.* **2015**, *8*, 014008. [CrossRef]
27. Reimers, N.; Gurevych, I. Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. *arXiv* **2017**, arXiv:1707.06799.
28. Zhang, B.; Rajan, R.; Pineda, L.; Lambert, N.; Biedenkapp, A.; Chua, K.; Hutter, F.; Calandra, R. On the importance of hyperparameter optimization for model-based reinforcement learning. In Proceedings of the International Conference on Artificial Intelligence and Statistics, San Diego, CA, USA, 13–15 April 2021; PMLR: Cambridge, MA, USA, 2021; pp. 4015–4023.
29. Wei, Y.; Chen, Z.; Zhao, C.; Tu, Y.; Chen, X.; Yang, R. A BiLSTM hybrid model for ship roll multi-step forecasting based on decomposition and hyperparameter optimization. *Ocean Eng.* **2021**, *242*, 110138. [CrossRef]
30. Kaur, S.; Aggarwal, H.; Rani, R. Hyper-parameter optimization of deep learning model for prediction of Parkinson's disease. *Mach. Vis. Appl.* **2020**, *31*, 32. [CrossRef]

31. Sidana, S. Grid Search Optimized Machine Learning based Modeling of CO_2 Emissions Prediction from Cars for Sustainable Environment. *Int. J. Curr. Sci. Res. Rev.* **2024**, *7*. [CrossRef]
32. Toro-Molina, C.; Rivera-Tinoco, R.; Bouallou, C. Hybrid adaptive random search and genetic method for reaction kinetics modelling: CO_2 absorption systems. *J. Clean. Prod.* **2012**, *34*, 110–115. [CrossRef]
33. Bian, C.; Zhang, S.; Yang, J.; Liu, H.; Zhao, F.; Wang, X. Bayesian optimization design of inlet volute for supercritical carbon dioxide radial-flow turbine. *Machines* **2021**, *9*, 218. [CrossRef]
34. Tao, H.; Salih, S.Q.; Saggi, M.K.; Dodangeh, E.; Voyant, C.; Al-Ansari, N.; Yaseen, Z.M.; Shahid, S. A newly developed integrative bio-inspired artificial intelligence model for wind speed prediction. *IEEE Access* **2020**, *8*, 83347–83358. [CrossRef]
35. Dimov, D.T. Rotation-invariant NCC for 2D color matching of arbitrary shaped fragments of a fresco. *Pattern Recognit. Lett.* **2020**, *138*, 431–438. [CrossRef]
36. Eshtay, M.; Faris, H.; Obeid, N. Improving extreme learning machine by competitive swarm optimization and its application for medical diagnosis problems. *Expert Syst. Appl.* **2018**, *104*, 134–152. [CrossRef]
37. Bacanin, N.; Bezdan, T.; Tuba, E.; Strumberger, I.; Tuba, M. Optimizing convolutional neural network hyperparameters by enhanced swarm intelligence metaheuristics. *Algorithms* **2020**, *13*, 67. [CrossRef]
38. Mirjalili, S.; Jangir, P.; Mirjalili, S.Z.; Saremi, S.; Trivedi, I.N. Optimization of problems with multiple objectives using the multi-verse optimization algorithm. *Knowl.-Based Syst.* **2017**, *134*, 50–71. [CrossRef]
39. Mirjalili, S. SCA: A sine cosine algorithm for solving optimization problems. *Knowl.-Based Syst.* **2016**, *96*, 120–133. [CrossRef]
40. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
41. Su, H.; Zhao, D.; Heidari, A.A.; Liu, L.; Zhang, X.; Mafarja, M.; Chen, H. RIME: A physics-based optimization. *Neurocomputing* **2023**, *532*, 183–214. [CrossRef]
42. Mirjalili, S. The ant lion optimizer. *Adv. Eng. Softw.* **2015**, *83*, 80–98. [CrossRef]
43. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2015**, *95*, 51–67. [CrossRef]
44. Dhiman, G.; Kaur, A. STOA: A bio-inspired based optimization algorithm for industrial engineering problems. *Eng. Appl. Artif. Intell.* **2019**, *82*, 148–174. [CrossRef]
45. Zhao, S.; Zhang, T.; Ma, S.; Chen, M. Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications. *Eng. Appl. Artif. Intell.* **2022**, *114*, 105075. [CrossRef]
46. Wolpert, D.H. What is important about the no free lunch theorems? In *Black Box Optimization, Machine Learning, and No-Free Lunch Theorems*; Springer International Publishing: Cham, Switzerland, 2021; pp. 373–388.
47. Adam, S.P.; Alexandropoulos, S.A.N.; Pardalos, P.M.; Vrahatis, M.N. *No Free Lunch Theorem: A Review. Approximation and Optimization: Algorithms, Complexity and Applications*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 57–82.
48. Wang, J.; Wang, W.C.; Hu, X.X.; Qiu, L.; Zang, H.-F. Black-winged kite algorithm: A nature-inspired meta-heuristic for solving benchmark functions and engineering problems. *Artif. Intell. Rev.* **2024**, *57*, 98. [CrossRef]
49. Liu, Y.; Cao, B. A novel ant colony optimization algorithm with Levy flight. *IEEE Access* **2020**, *8*, 67205–67213. [CrossRef]
50. Mohammad, S.; Ibrahim, M.; Zaid, M.; Yousef, S.M.K.; Anas, A.L.B.; Saja, A.D.; Laith, A. Harris Hawks Optimization Algorithm: Variants and Applications. *Arch. Comput. Methods Eng.* **2022**, *7*, 5579–5603.
51. Martínez, S.Z.; Montano, A.A.; Coello, C.A.C. A nonlinear simplex search approach for multi-objective optimization. In Proceedings of the 2011 IEEE Congress of Evolutionary Computation (CEC), New Orleans, LA, USA, 5–8 June 2011; IEEE: New Orleans, LA, USA, 2011; pp. 2367–2374.
52. Wu, G.; Mallipeddi, R.; Suganthan, P.N. *Problem Definitions and Evaluation Criteria for the CEC 2017 Competition on Constrained Real-Parameter Optimization*; Technical Report; National University of Defense Technology: Changsha, China; Kyungpook National University: Daegu, Republic of Korea; Nanyang Technological University: Singapore, 2017.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.