symmetry

MDPI

Article

Failure-Aware Bidirectional Evolutionary Knowledge Assimilation with Dynamic Regulation for Adaptive Optimization

Hongmei Shao [1,*], Rongguo Qu [2] and Qinwei Fan [2,3]

[1] College of Science, China University of Petroleum (East China), Qingdao 266580, China
[2] School of Science, Xi'an Polytechnic University, Xi'an 710048, China; 220811002@stu.xpu.edu.cn (R.Q.); qinweifan@xpu.edu.cn (Q.F.)
[3] School of Mathematics and Information Science, Guangzhou University, Guangzhou 510006, China
* Correspondence: hmshao@upc.edu.cn

Abstract

Efficient exploitation of evolutionary knowledge while preserving population diversity remains a central challenge in optimization. Existing knowledge-learning evolutionary algorithms primarily rely on successful experiences, overlooking structural information embedded in failed search attempts. This asymmetric learning limits adaptability and may cause premature convergence in high-dimensional landscapes. To address this issue, a failure-aware bidirectional evolutionary knowledge assimilation framework is developed within the honey badger optimization algorithm. Unsuccessful offspring are treated as negative knowledge carriers and transformed through symmetric adversarial reflection, enabling simultaneous extraction of positive and negative structural information. A time-dependent regulation mechanism dynamically adjusts knowledge assimilation intensity across evolutionary phases to balance exploration and exploitation. In addition, a continuous mutation spectrum transition strategy adaptively integrates Cauchy and Gaussian perturbations, facilitating smooth migration from global exploration to local refinement. Comprehensive experiments conducted on the CEC 2017 benchmark suite across 10, 30, and 50 dimensions validate the proposed framework, establishing a novel failure-aware bidirectional evolutionary learning paradigm for knowledge-driven optimization. The results demonstrate that our method achieves statistically significant and consistent performance improvements over classical baseline algorithms. Furthermore, its robustness and cross-domain adaptability are corroborated through successful application to a real-world constrained engineering problem: welded beam design.

Keywords: Honey Badger Algorithm; knowledge learning; opposite learning; adaptive mutation

Check for updates

Academic Editor: Shi Cheng

Received: 19 March 2026
Revised: 6 April 2026
Accepted: 8 April 2026
Published: 25 May 2026

Copyright: © 2026 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.

1. Introduction

In the contemporary world, where complexity and change are the norm, optimization problems are pervasive, spanning a diverse array of application domains. From the daily time management and financial planning of individuals to the scheduling of urban transportation and the optimization of corporate production, optimization methods are employed extensively to enhance efficiency, reduce costs, and maximize resource utilization. Even in the realm of global climate modeling and international financial risk management, optimization techniques are utilized to improve outcomes.

Nevertheless, conventional optimization techniques encounter numerous obstacles when confronted with actual-world issues. To illustrate, the gradient descent method [1] is a prevalent optimization algorithm that reduces the loss function by modifying the parameters in a manner that aligns with the negative gradient direction of the objective function. However, gradient descent algorithms are prone to converging to local optimal solutions, particularly when confronted with non-convex problems. In such cases, the gradient may not be directly computable at specific points, rendering it challenging to identify the global optimal solution. Furthermore, the sensitivity of gradient descent algorithms to the selection of the learning rate may result in slow or unstable convergence. Another traditional optimization method is the exhaustive method [2], which can effectively circumvent the issue of gradient computation. However, due to its reliance on traversing the solution space to determine the optimal solution, the computational complexity increases exponentially with the size of the solution space. This renders the algorithm impractical due to its high time consumption.

To address these challenges, an increasing number of scholars are dedicated to developing optimization algorithms that do not rely on gradient and have low computational complexity. In the 1950s, Metropolis et al. proposed the simulated annealing algorithm [3], which constructs a stochastic search rule by simulating the annealing process of a solid and is used to identify the global optimal solution within the solution space. The simulated annealing algorithm has prompted considerable research by scholars, who have sought to further study optimization algorithms based on the simulation of natural phenomena or the intelligent behavior of groups. These collectively constitute what is known as intelligent optimization algorithms [4]. Intelligent optimization algorithms are typically classified into three categories, depending on the underlying mechanism they employ. These categories include algorithms based on evolutionary mechanisms [5], group intelligence optimization algorithms [6], and algorithms based on physical principles [7].

Intelligent optimization algorithms based on evolutionary mechanisms simulate the biological evolution process for optimization search. Representative algorithms include genetic algorithms [8] and differential evolution algorithms [9]. Group intelligent optimization algorithms, on the other hand, perform a global search by simulating group behavior. For example, the ant colony algorithm [10] simulates the foraging behavior of ants, and the particle swarm optimization algorithm [11] simulates the group movement behavior of bird flocks. These algorithms achieve global search through the collaborative and competitive interactions among individuals. In contrast, intelligent optimization algorithms based on physical principles optimize by simulating the behavior of physical systems, as exemplified by the aforementioned simulated annealing algorithm. Despite the advantages of intelligent optimization algorithms, including the ability to adapt to high-dimensional complex spaces and the lack of reliance on gradient, they still face challenges in practical applications. These include issues related to parameter setting, initial population selection, local optimal traps, and convergence speed.

To address these issues, numerous scholars have put forth a variety of potential enhancements, with a primary emphasis on parameter tuning [12–14], the introduction of novel search strategies [15,16], the refinement of population initialization methods [17–20], and the integration of these algorithms with other algorithms [21–25]. Nevertheless, few scholars have considered the experiences generated by the algorithms during the iteration process. Nevertheless, these experiences are crucial for the algorithms to more effectively explore the optimal solution regions. It is therefore imperative to consider how these experiences can be effectively learned and utilized within the optimization process, in order to enhance the performance of intelligent optimization algorithms. The majority of current research concentrates on the collection and utilization of historical success experiences

of individuals, with minimal consideration given to the inherent differences between individuals. The disparate locations of the solution space for different individuals may render the success experiences of others ineffective for some, potentially leading them to pursue suboptimal search paths.

Optimization algorithms can be viewed as adaptive dynamical systems evolving over complex fitness landscapes. Traditional metaheuristics rely on stochastic operators and population interactions to approximate global optima, yet their performance is fundamentally constrained by how efficiently information is extracted and reused during the evolutionary process.

Recent advances in knowledge-learning evolutionary computation have introduced supervised learning mechanisms into metaheuristic frameworks, enabling algorithms to model historical successful experiences and guide individual updates accordingly. However, existing approaches exhibit two fundamental theoretical limitations:

1. Unidirectional knowledge acquisition—only successful evolutionary transitions are modeled, while failed trajectories are discarded.
2. Static knowledge assimilation intensity—learning strength is typically controlled by fixed parameters, ignoring phase-dependent search dynamics.

From an information-theoretic perspective, failed individuals contain implicit boundary and gradient information about infeasible or suboptimal regions of the search space. Discarding such information leads to incomplete landscape modeling and reduced structural awareness.

Therefore, a natural question arises:

Can failed evolutionary attempts be transformed into constructive knowledge to improve adaptive search dynamics?

To address this issue, Jiang [26] et al. proposed the knowledge-learning evolutionary computation (KLEC) paradigm, in which neural networks are trained using historical successes and then used to customize the evolutionary direction for each individual, thus preventing individuals from blindly relying on the historical experiences of others. The experimental results demonstrate that the implementation of KLEC in differential evolutionary algorithms and particle swarm optimization algorithms markedly enhances the algorithms' performance. However, KLEC's approach to knowledge acquisition is limited to the learning of historical successful experiences, with minimal consideration given to the potentially useful knowledge embedded within failure experiences. This may result in the algorithms becoming trapped in local optima or exhibiting slow convergence.

This paper proposes a bidirectional evolutionary knowledge assimilation (BEKA) paradigm to address the above question. Within this framework, successful trajectories provide directional reinforcement information, while failed trajectories extract complementary structural information through a symmetric reflection mechanism. The intensity of knowledge assimilation is dynamically regulated over time.

Unlike conventional hybrid improvement strategies, this study redefines the role of "failure" in evolutionary optimization, elevating it from discarded noise to a structured source of information.

By embedding this paradigm into the honey badger optimization mechanism [27], a closed-loop system that integrates the following components is constructed: a population-level stochastic search mechanism, an individual-level knowledge modeling mechanism, a time-dependent learning regulation mechanism, and a distribution-adaptive mutation spectrum control mechanism. This unified structure transforms the optimizer into a "partially learnable dynamical system," establishing a bridge between stochastic search theory and data-driven evolutionary modeling. The main contributions are summarized as follows:

(a) A bidirectional knowledge evolution mechanism with synergistic positive–negative experience learning is constructed. The traditional KLEC framework utilizes only successful evolutionary samples for knowledge modeling, neglecting the latent structural information contained in failed samples. In this work, failed individuals are treated, for the first time, as "negative knowledge carriers." Through an adversarial generation mechanism, their dual structural representations are mapped and incorporated into the learning process, enabling cooperative evolutionary learning from both positive and negative experiences.

Theoretically, this mechanism expands the sample space of knowledge learning and transforms knowledge updating from unilateral statistical learning into bidirectional structural inference learning, thereby improving the efficiency of information utilization in the search space.

(b) A dynamic knowledge assimilation regulation model is proposed to achieve phase-adaptive search control. Specifically, a time-varying learning parameter is designed to evolve along the iteration process. This mechanism essentially establishes a time-driven regulation model for knowledge assimilation intensity.

As a result, the algorithm emphasizes structural exploration in the early stage, maintains a balanced exploration–exploitation trade-off in the middle stage, and strengthens knowledge-guided exploitation in the later stage. This forms a three-phase evolutionary scheduling mechanism characterized by early exploration dominance, intermediate balance, and late-stage knowledge reinforcement.

(c) A Cauchy–Gaussian fusion-based adaptive mutation spectrum modulation mechanism is proposed. Unlike fixed-mutation-intensity strategies, this study constructs a dual-distribution fusion mutation model regulated by sinusoidal weighting.

In the early stage, the heavy-tailed Cauchy distribution enhances global jump capability, while in the later stage, the Gaussian distribution reinforces local convergence precision. This mechanism constitutes a continuous mutation spectrum transition model, enabling smooth migration from exploration-oriented perturbation to exploitation-oriented refinement.

(d) A novel optimization framework coupling knowledge-driven learning with swarm intelligence mechanisms is constructed. Rather than simply superimposing a neural network module onto the algorithm, this work achieves systematic coupling among population behavior modeling, individual knowledge learning, dynamic regulation mechanisms, and adversarial failure-based learning.

This integrated architecture provides a general paradigm for transforming metaheuristic algorithms into "learnable optimizers."

The remainder of this paper is organized as follows: Section 2 presents a review of the literature on KLEC and honey badger optimization algorithms. Section 3 provides a comprehensive description of the honey badger optimization algorithm based on evolutionary knowledge learning, as proposed in this paper. Section 4 presents the experimental results and their analysis. Section 5 presents the application of the KLEC-HBA to the design problem of welded beams. Section 6 provides a summary of the work presented in this paper and offers insights into potential avenues for future research.

2. Related Work

2.1. Honey Badger Algorithm

The honey badger algorithm (HBA) [27] is an intelligent optimization algorithm inspired by the foraging behavior of an animal called the honey badger. Honey badger foraging behavior is usually divided into two types: digging and nectar seeking. Digging

behavior means that the honey badger estimates the approximate location of the prey through its sense of smell and chooses a suitable place to dig and capture the prey after arriving at a location close to the prey. Honey-seeking behavior refers to the honey badger's behavior of following the guidance of honey guide birds to find honey. As mentioned above, the algorithm mimics the honey badger's foraging behavior to find the optimal solution to the optimization problem, which is divided into the "digging phase" and the "honey phase", as described below.

2.1.1. Digging Phase

During this phase, the honey badger is mainly influenced by the intensity of the prey's scent, the distance between itself and the prey, and the time-varying search factor to determine the location of the prey and capture it. The following is a mathematical expression for its foraging in this phase:

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

where x_{new} indicates the new location of the honey badger and x_{prey} is the location of the prey, i.e., the best location found so far. $\beta \geq 1$ (default = 6) is the ability of the honey badger to obtain food. I is the scent strength of the prey. r_3, r_4, and r_5 are three different random numbers between 0 and 1. F serves as a flag to change the direction of the search. α is a density factor to control the time-varying randomization to ensure a smooth transition from exploration to exploitation.

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

where S is the source strength or concentration strength, d_i is the distance between the prey and the ith badger, and r_2 is a random number between 0 and 1.

$$F = \begin{cases} 1, & r_6 \leq 0.5 \\ -1, & else \end{cases} \tag{3}$$

$$\alpha = C \times exp(\frac{-t}{T}) \tag{4}$$

where T is the maximum number of iterations and C is a constant ≥ 1 (default = 2).

2.1.2. Honey Phase

In contrast to the digging phase, in the honey phase the honey badger is guided to the hive by a honey guide bird and searches for honey around the hive, eventually obtaining honey to meet its energy needs. The above behavior can be expressed in a mathematical formula as follows:

$$x_{new} = x_{prey} + F \times \beta \times r_7 \times d_i \tag{5}$$

where r_7 is a random number between 0 and 1.

The step-by-step procedure of the HBA is presented in Algorithm 1.

2.2. Knowledge Learning of Algorithms

The knowledge-learning framework for evolutionary computation, as proposed by Jiang [26] et al., comprises two stages: the initial stage, which involves learning knowledge

from historical experience, and the subsequent stage, which utilizes the knowledge thus acquired to guide individual evolution. The following section will provide a detailed account of the process of algorithmic knowledge learning from these two stages.

Algorithm 1 Honey badger algorithm

Input: Maximum iterations T, Population size N, Parameters β, C
Output: Best position x_{prey}
Set parameters T, N, β, C
Initialize population with random positions
Evaluate the fitness of each honey badger position x_i using the objective function, and assign the fitness value to f_i, $i \in [1, 2, \ldots, N]$
Save the best position x_{prey} and assign its fitness value to f_{prey}
while $t \leq T$ **do**
  Update the decreasing factor α using Equation (4)
  for $i = 1$ to N **do**
    Calculate the intensity I_i using Equation (2)
    if $r < 0.5$ **then**
      Update the position x_{new} using Equation (1)
    else
      Update the position x_{new} using Equation (5)
    end if
    Compute the objective function value $f_{x_{\text{new}}}$
    Select the best agent x_{prey}
  end for
end while Stop criteria satisfied
return x_{prey}

2.2.1. Learning Knowledge

In each iteration of the update process, the algorithm generates new offspring individuals from parent individuals in accordance with the established updating rules. Subsequently, the algorithm assesses whether the offspring individuals are superior to their parent counterparts by comparing their fitness values before and after the update. Should an offspring individual prove more successful than its parent, the algorithm records this as a successful historical experience, which is then stored in the knowledge base. This process continues until all successful historical experiences of the current iteration have been accumulated. Subsequently, a neural network is employed for the purpose of learning from all of these successful experiences. In particular, the neural network is trained using all successfully evolved individuals and their corresponding successful evolutionary directions as training samples. The parent individuals that underwent successful evolution serve as input for the neural network training samples, while their corresponding successful evolutionary directions represent the ideal outputs for the training samples. By training the neural network in this manner, the parameters of the neural network are optimized and saved for future utilization in the subsequent phase of "applying the knowledge acquired through the algorithm".

2.2.2. Utilizing Knowledge

In the preceding phase, the paper provided an analysis of the effective learning of algorithmic successful experiences (see Figure 1). In this subsequent phase, the paper will elaborate on the effective utilization of acquired knowledge. Typically, intelligent optimization algorithms update individuals based solely on their corresponding learning rules. However, in the context of algorithmic knowledge acquisition, a novel updating rule, tailored to individuals, is devised and integrated into the existing updating mechanisms of the algorithm. This rule employs the acquired algorithmic knowledge to establish a

novel learning trajectory, tailored to the specific characteristics of each individual, thereby facilitating the generation of new individuals along this trajectory. Specifically, the individual is introduced into the previously trained neural network as a testing sample, and the neural network then outputs a learning direction that is tailored to that specific individual. Subsequently, the aforementioned learning direction is employed for the generation of a new individual. As the offspring is derived from the successful historical experiences of the preceding generation, it is more likely to succeed than an offspring generated solely by its inherent learning rules. The following section provides a detailed account of the aforementioned process.

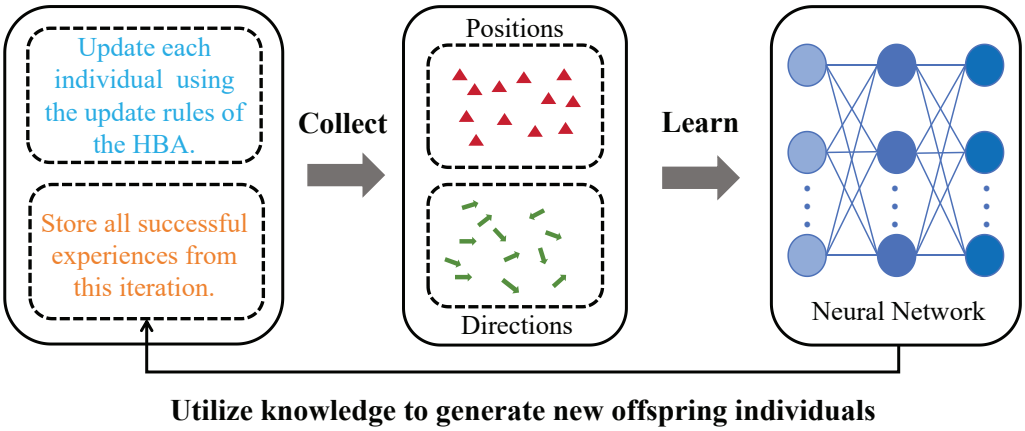


Figure 1. Algorithmic knowledge-learning framework diagram.

3. The Proposed Improved Honey Badger Algorithm

3.1. Algorithm Flowchart

The flowchart of the KLEC-HBA is shown in Figure 2 below.

3.2. Dynamic Learning Parameters

In the context of algorithmic knowledge acquisition, the learning parameter represents a crucial element that influences the algorithm's ability to integrate historical successful experiences. In order to maintain equilibrium between the learning from experiences and the algorithm's inherent search strategies, Jiang et al. defined the range of the learning parameter between 0 and 0.5. Through the sensitivity testing of parameters, the optimal value for the learning parameter was determined to be 0.2. Nevertheless, in practical applications, the optimal value for the learning parameter is not a fixed constant. The optimal numerical value of the learning parameter may require adjustment in light of the specific characteristics of the datasets and types of problems under consideration. Accordingly, this paper proposes the concept of dynamic learning parameters as a means to enhance the algorithm's learning efficacy regarding historical experiences. The purpose of these dynamic learning parameters is to concentrate the algorithm's efforts on optimizing its search strategy during the initial stages of the search process. As the number of iterations increases, the algorithm gradually shifts its focus towards reinforcing the learning derived from historical experiences, thereby providing more effective guidance for the problem-solving process. The precise formula is as follows:

$$LR = 0.2 + 0.3 \times \frac{t}{T} \tag{6}$$

where the variable t represents the current number of iterations, whereas T denotes the maximum number of iterations.

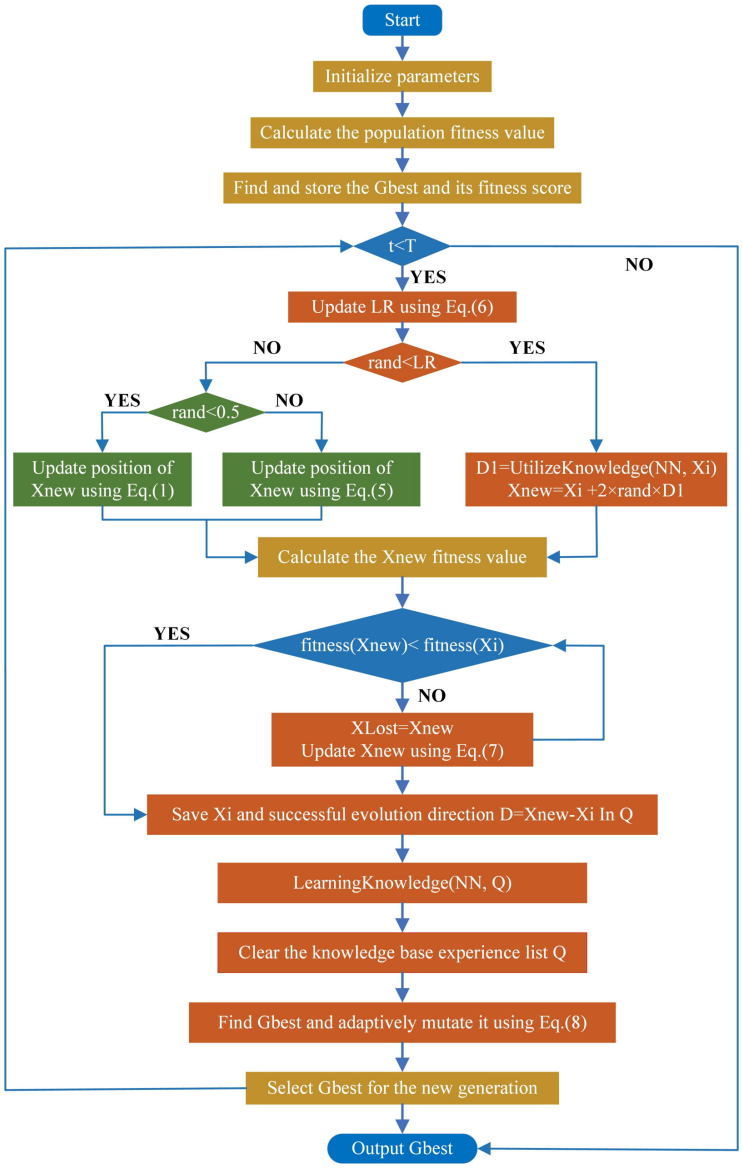


Figure 2. Flowchart of KLEC-HBA.

3.3. Collecting Success Stories of Potential Individuals

In the original framework of algorithmic knowledge acquisition, researchers concentrated exclusively on the collection of historical experiences from successful individuals within the population, thereby overlooking the possibility of outstanding individuals among those who had not succeeded. This may result in the loss of valuable knowledge during the population evolution process. To address this issue, this paper employs antagonistic learning on the offspring individuals that did not successfully evolve. Subsequently, the newly generated offspring individuals are subjected to comparison with the non-evolved parent individuals. Should this evolutionary process prove successful, the experience will be collated in order to address the potential loss of useful knowledge during population evolution. To address this issue, this paper employs antagonistic learning on the offspring individuals that did not successfully evolve. Let X_{lost} denote such an offspring individual that fails to evolve.

$$X_{new} = (1 - \frac{t}{T}) \times (lb - ub) - m \times X_{lost} \tag{7}$$

where lb and ub denote the lower and upper bounds, respectively; X_{lost} represents the failed offspring individual; X_{new} is the newly generated offspring; and m is a random vector following a standard normal distribution.

If the fitness value of the new offspring individual X_{new}, produced after dyadic learning, is superior to that of the unevolved parent individual X, the position of individual X and its corresponding evolutionary direction $D = X_{new} - X$ is preserved as a successful historical experience.

3.4. Adaptive Mutation Methods

This study incorporates adaptive mutation of the current global best solution in each iteration, with the objective of preventing the algorithm from prematurely converging to local optima during the search process. This approach differs from conventional mutation strategies in that it effectively integrates characteristics of Gaussian mutation [28] and Cauchy mutation [29], thereby offering a mutation method that is tailored to the current environment and which is designed to yield the best contemporary solution globally.

In particular, Gaussian mutation typically features a relatively small standard deviation, concentrating mutation values around the mean. In contrast, the Cauchy distribution possesses a longer tail, resulting in a wider range of mutation amplitudes. Furthermore, by adaptively adjusting weights, the algorithm autonomously tunes the mutation strategy, promoting extensive exploration of the solution space in the early stages and fine-tuning solution quality later on. This enhancement bolsters the algorithm's global search capability and local convergence potential.

$$best_{new} = (1 - sin(\frac{\pi t}{2T})) \times K + sin(\frac{\pi t}{2T}) \times G \tag{8}$$

Here, $best_{new}$ represents the offspring global best individual generated through adaptive mutation of the parent global best individual. t denotes the current iteration number, and T signifies the maximum iteration count. K stands for the global best offspring individual produced by Cauchy mutation of the parent global best individual. G refers to the global best offspring individual generated by Gaussian mutation of the parent global best individual.

3.5. Algorithmic Pseudo-Code

The step-by-step procedure of the KLEC-HBA is presented in Algorithm 2. The knowledge base Q is defined as a set of experience tuples, where each entry is represented as (X, D), with X denoting the individual position and $D = X_{\text{new}} - X$ representing the corresponding evolutionary direction.

3.6. Computational Complexity of KLEC-HBA

The computational complexity of the KLEC-HBA is primarily determined by its core improvement modules. The time complexity of the bidirectional knowledge-learning mechanism (i.e., the training and inference of the lightweight neural network) in each iteration is $O(N \times dim)$, and the complexity of the population update and fitness evaluation loop is also $O(N \times dim)$. The overhead of operations such as dynamic learning rate adjustment and adaptive mutation is relatively low. Considering the main operations of the algorithm over T iterations, its overall time complexity is $O(T \times N \times dim)$, where N, dim, and T represent the population size, problem dimension, and maximum number of iterations, respectively. The space complexity of the algorithm is $O(N \times dim)$, primarily used to store the population positions and knowledge base experience. This analysis demonstrates that despite the introduction of the knowledge-learning module, the KLEC-HBA main-

tains computational and storage efficiency of the same order as classical population-based metaheuristic algorithms.

Algorithm 2 KLEC-HBA

Input: Population P, Experience list Q, Neural network NN parameters, Maximum iterations T
Output: Global best individual
Initialize P, Q, set NN parameters, T
Calculate the population fitness value and save the global optimal individual and its fitness value
for $t = 1$ to T **do**
  Update LR using Equation (6)
  for each individual X_i **do**
    if rand $<$ LR **then**
      $S = \text{UtilizeKnowledge}(NN, X_i)$
      $X_{\text{new}} = X_i + 2 * rand * S$
    else
      if rand < 0.5 **then**
        Update position of X_{new} using Equation (1)
      else
        Update position of X_{new} using Equation (5)
      end if
    end if
    if fitness(X_{new}) $<$ fitness(X_i) **then**
      Save the position of X_i and successful evolution direction $D = X_{\text{new}} - X_i$ in Q
      $X_i = X_{\text{new}}$
    else
      $X_{\text{lost}} = X_{\text{new}}$
      Update position of X_{new} using Equation (7)
      if fitness(X_{new}) $<$ fitness(X_i) **then**
        Save the position of X_i and successful evolution direction $D = X_{\text{new}} - X_i$ in Q
        $X_i = X_{\text{new}}$
      end if
    end if
  end for
  LearningKnowledge(NN, Q);
  Clear the knowledge base experience list Q
  Find the current global best individual *Gbest* and adaptively mutate it using Equation (8)
  Select the globally best individual with higher fitness as *Gbest* for the new generation
end for
return *Gbest*

4. Experimental Results and Discussion

To assess the efficacy of the proposed algorithm, we conducted an empirical investigation by testing it on 29 functions from the CEC2017 test suite [30] and comparing the outcomes with several established algorithms. All experiments were conducted on a computer with a 1.7 GHz Intel i5-1240 CPU and 16.00 GB of RAM, using Matlab R2024a as the computational platform.

4.1. Parameter Setting

The list of parameters and settings for each algorithm utilized in this study is presented in Table 1.

Table 1. Parameters and settings for each algorithm used in this study.

Algorithms	Parameter Settings
IHBA	$\beta = 6$.
HHO	$\beta = 1.5$.
HBA	$C = 2$, $\beta = 6$.
PSO	$c_1 = 2$, $c_2 = 2$, $w = 0.9$.
WOA	α decreases linearly from 2 to 0.
SSA	c_1 and c_2 are randomly distributed.
ABC	ϕ is a random number between -1 and 1.
KLEC-HBA	$C = 2$, $\beta = 6$, *epoch* = 25, *hidden* number = 16.

4.2. Sensitivity Analysis

Given the utilization of two user-defined parameters *epoch* and *hidden* in the KLEC-HBA, a meticulous selection of appropriate values becomes pivotal, as they exert a substantial influence on the performance of the KLEC-HBA. *epoch* denotes the number of iterations for the neural network to update the network weights using the gradient descent method, *hidden* denotes the number of neurons in the hidden layer of the neural network. To evaluate the impact of these parameters, a sensitivity analysis is conducted on CEC Composition Function 1 (population size = 30, iteration number = 1000). Figure 3 presents the parameter sensitivity analysis of the proposed method on four representative functions from the CEC2017 benchmark suite, namely, F1, F6, F15, and F25. These functions cover diverse landscape characteristics, including unimodal, multimodal, and hybrid optimization problems. The performance values are displayed in logarithmic scale to improve visual discrimination.

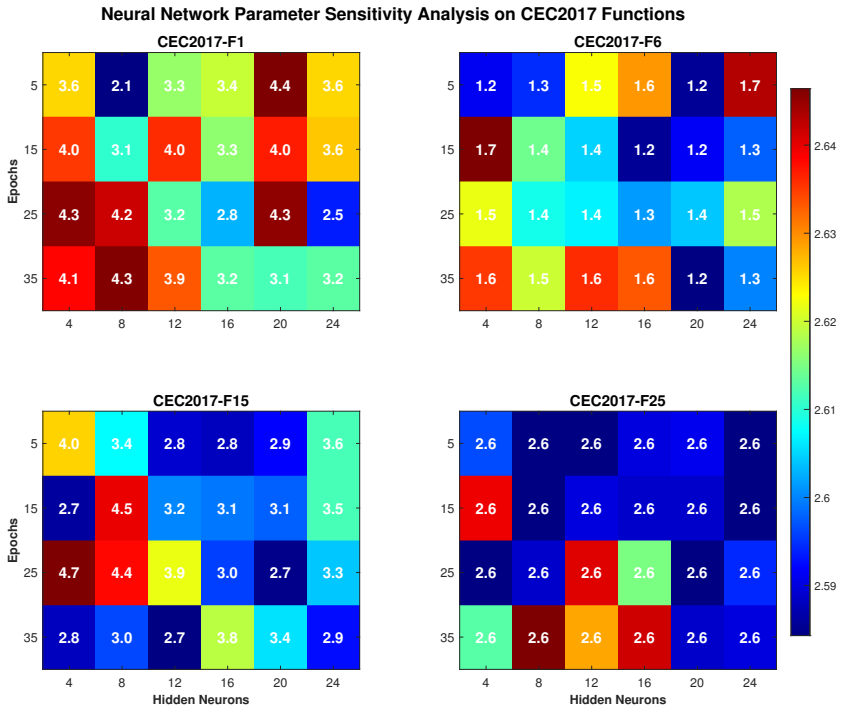


Figure 3. The parameters *epoch* and *hidden* tune the visualization.

From the overall observation, both the number of training epochs and hidden neurons significantly affect the optimization performance. For F1 and F15, the performance exhibits clear sensitivity to parameter settings, where moderate configurations tend to achieve better results. In particular, the setting with 25 epochs and 16 hidden neurons consistently yields competitive or optimal performance, indicating a good balance between training

sufficiency and model complexity. For F6, the performance variation across different parameter settings is relatively small, suggesting that the problem is less sensitive to network configuration and can be solved robustly under a wide range of settings. In contrast, F15 shows stronger fluctuations, implying that more careful parameter tuning is required for complex landscapes. For F25, the performance remains almost unchanged across all configurations, indicating that the proposed method is highly robust on this type of problem and less dependent on parameter selection.

Overall, the experimental results demonstrate that the configuration (epochs = 25, hidden = 16) achieves stable and competitive performance across different types of optimization problems, highlighting its strong generalization capability.

4.3. Experimental Description

This section presents a comparative analysis of the performance of the proposed algorithm in this paper with other intelligent optimization algorithms, as well as an ablation study on the CEC2017 test suite [30]. Both comparison experiments and ablation experiments are conducted on the 10-dimensional, 30-dimensional, and 50-dimensional CEC2017 test suite. Each experiment is conducted 30 times independently. The experimental results are then subjected to hypothesis testing using the Wilcoxon rank sum test and the Friedman rank sum test, to illustrate the test performance of the proposed algorithm in this paper. Furthermore, this paper employs a backpropagation (BP) neural network comprising two hidden layers, with 16 neurons in each. The activation functions of the hidden layer and the output layer are ”tansig” and ”purelin”, respectively. The BP neural network employs gradient descent to train the weights throughout 25 iterations.

4.4. Experimental Results

The following table presents the mean rank, which represents the average of the ranks derived from the Friedman rank sum test for each algorithm, and the total rank, which denotes the overall performance ranking of each algorithm. The symbols “+”, “−”, and “≈” indicate that the performance of the KLEC-HBA is superior, inferior, or comparable to that of the comparison algorithms. The “*p*-value” indicates the statistically significant result of the Wilcoxon rank sum test based on multiple problems. “R+/R−” denotes the corresponding rank sum. When the *p*-values is less than 0.05, the KLEC-HBA outperforms the other comparison algorithms.

4.4.1. Comparative Experiment

This subsection presents the findings of a comparison experiment between the KLEC-HBA and seven other algorithms (ABC [31], PSO [11], SSA [32], WOA [33], HHO [34], the HBA [27], and the IHBA [35]) on the CEC2017 test suite. This experiment aims to evaluate the performance of the KLEC-HBA. The outcomes of the 10-dimensional examination conducted on CEC2017 are illustrated in Table 2. Although the proposed method achieves the best overall performance, it may not always outperform all competitors on highly complex functions, such as multimodal or composition problems, where the landscape is more irregular and difficult to explore. The results of the 30-dimensional CEC2017 test suite are presented in Table 3. The *p*-values of the seven comparison algorithms are significantly less than 0.001, and the average value of rank and the overall values are less favorable than those of the KLEC-HBA, indicating that the KLEC-HBA’s is relatively optimal. The results of the 50-dimensional CEC2017 test suite, as presented in Table 4, demonstrate that the KLEC-HBA exhibits superior performance compared with the six comparison algorithms across all functions. The *p*-values of the seven comparison algorithms are notably low, and the KLEC-HBA achieves the highest rank in terms of overall and average rank. Furthermore, as illustrated in Figure 4, the KLEC-HBA exhibits superior convergence speed and

optimization finding performance on the majority of the tested functions in comparison to the other evaluated algorithms. Due to space limitations, only representative functions are presented. These functions are selected to cover different categories in the CEC2017 benchmark suite, including unimodal, multimodal, and composition functions.

Table 2. Experimental results of algorithm comparison on the 10-dimensional CEC2017 test suite.

Fun.	Meas.	KLEC-HBA	HBA	IHBA	HHO	SSA	WOA	ABC	PSO
F1	Ave	2.4712×10^{3}	3.6318×10^{3}	2.4900×10^{7}	2.8619×10^{5}	1.6750×10^{8}	3.9352×10^{5}	1.3045×10^{9}	5.4889×10^{8}
	Std	1.2551×10^{3}	3.8034×10^{3}	4.7984×10^{7}	1.6591×10^{5}	5.0553×10^{8}	5.6476×10^{5}	3.5144×10^{8}	5.6508×10^{8}
F3	Ave	1.0800×10^{-13}	1.5348×10^{-13}	2.2025×10^{3}	1.1469	7.3896×10^{-13}	6.8418×10^{2}	1.6103×10^{4}	3.9452×10^{3}
	Std	6.2556×10^{-14}	1.3677×10^{-13}	2.1092×10^{3}	4.4713×10^{-1}	3.8366×10^{-14}	4.6279×10^{2}	4.0452×10^{3}	1.8193×10^{3}
F4	Ave	3.9061×10^{-2}	6.3526×10^{-1}	5.3868×10^{1}	1.9443×10^{1}	1.2885×10^{1}	2.6598×10^{1}	1.0799×10^{2}	4.7128×10^{1}
	Std	6.9991×10^{-2}	4.4577×10^{-1}	3.9847×10^{1}	3.3233×10^{1}	2.2355×10^{1}	2.8237×10^{1}	2.9363×10^{1}	4.0463×10^{1}
F5	Ave	5.3573×10^{1}	1.6715×10^{1}	4.5399×10^{1}	5.2043×10^{1}	3.1500×10^{1}	5.2194×10^{1}	5.1184×10^{1}	4.2108×10^{1}
	Std	2.4675×10^{1}	6.1833	1.2687×10^{1}	1.1255×10^{1}	8.4489	1.5689×10^{1}	4.9578	5.9094
F6	Ave	1.0422×10^{1}	2.5533×10^{-1}	1.6414×10^{1}	2.8250×10^{1}	1.8946	2.5330×10^{1}	2.7663×10^{1}	1.8215×10^{1}
	Std	1.6691×10^{1}	8.0684×10^{-1}	5.3346	1.3350×10^{1}	2.8554	1.3420×10^{1}	3.5895	2.7043
F7	Ave	7.5736×10^{1}	3.0329×10^{1}	6.1245×10^{1}	7.1798×10^{1}	3.1090×10^{1}	7.9023×10^{1}	1.1129×10^{2}	9.9507×10^{1}
	Std	2.2131×10^{1}	6.4038	2.6426×10^{1}	2.0826×10^{1}	8.0458	2.4554×10^{1}	9.9065	1.0098×10^{1}
F8	Ave	3.2834×10^{1}	2.1292×10^{1}	3.0393×10^{1}	3.3240×10^{1}	3.2949×10^{1}	4.2037×10^{1}	5.4323×10^{1}	4.8527×10^{1}
	Std	4.7137	6.3656	5.4872	6.7881	1.1652×10^{1}	1.5377×10^{1}	6.0954	5.9331
F9	Ave	3.5041×10^{2}	1.7906×10^{-2}	1.3502×10^{2}	5.2611×10^{2}	2.3098×10^{2}	4.7266×10^{2}	5.9174×10^{2}	1.9067×10^{2}
	Std	3.1137×10^{2}	3.7748×10^{-2}	1.3018×10^{2}	2.9106×10^{2}	3.2951×10^{2}	3.9347×10^{2}	1.6130×10^{2}	4.7176×10^{1}
F10	Ave	6.0463×10^{2}	6.8334×10^{2}	1.0502×10^{3}	9.6009×10^{2}	7.6026×10^{2}	1.1888×10^{3}	1.4695×10^{3}	1.0945×10^{3}
	Std	4.4440×10^{2}	3.9731×10^{2}	1.8321×10^{2}	1.7419×10^{2}	2.7753×10^{2}	1.8154×10^{2}	7.1833×10^{1}	2.0426×10^{2}
F11	Ave	4.5211	1.5935×10^{1}	8.7185×10^{1}	7.8794×10^{1}	9.4265×10^{1}	1.1480×10^{2}	3.2482×10^{2}	1.4706×10^{2}
	Std	2.2238	1.4455×10^{1}	4.3358×10^{1}	6.4999×10^{1}	9.2174×10^{1}	8.1004×10^{1}	1.1599×10^{2}	9.5088×10^{1}
F12	Ave	1.0625×10^{4}	1.3455×10^{4}	3.6627×10^{6}	1.1703×10^{6}	1.3372×10^{6}	2.2913×10^{6}	3.1404×10^{7}	9.2610×10^{6}
	Std	1.0993×10^{4}	1.7310×10^{4}	5.0256×10^{6}	1.1965×10^{6}	2.8666×10^{6}	2.5966×10^{6}	1.2916×10^{7}	4.9932×10^{6}
F13	Ave	2.9366×10^{2}	6.5475×10^{3}	1.5325×10^{4}	1.4960×10^{4}	6.2690×10^{3}	1.6158×10^{4}	5.2182×10^{4}	3.2530×10^{4}
	Std	2.5530×10^{2}	1.0245×10^{4}	1.4102×10^{4}	1.1112×10^{4}	7.0145×10^{3}	1.1418×10^{4}	2.3985×10^{4}	2.6685×10^{4}
F14	Ave	3.7218×10^{1}	9.1736×10^{1}	1.1606×10^{2}	1.1346×10^{2}	7.8639×10^{1}	2.1369×10^{2}	6.8415×10^{2}	1.4104×10^{2}
	Std	1.9916×10^{1}	3.7920×10^{1}	3.7271×10^{1}	2.4647×10^{1}	3.2492×10^{1}	1.8127×10^{2}	6.6585×10^{2}	4.0629×10^{1}
F15	Ave	1.6172×10^{1}	8.5970×10^{1}	1.0089×10^{3}	1.8205×10^{3}	9.6152×10^{2}	3.5187×10^{3}	3.8436×10^{3}	1.8956×10^{3}
	Std	2.5366×10^{1}	8.0287×10^{1}	1.1486×10^{3}	1.5262×10^{3}	2.5372×10^{3}	5.0936×10^{3}	2.2338×10^{3}	1.9026×10^{3}
F16	Ave	3.5372×10^{2}	5.6657×10^{1}	1.5230×10^{2}	2.1802×10^{2}	1.9905×10^{2}	2.9872×10^{2}	2.1327×10^{2}	9.5409×10^{1}
	Std	6.7874×10^{1}	6.0834×10^{1}	1.1432×10^{2}	1.8588×10^{2}	1.7108×10^{2}	1.5495×10^{2}	6.2462×10^{1}	7.2891×10^{1}
F17	Ave	3.7113×10^{1}	4.9375×10^{1}	6.9450×10^{1}	6.4578×10^{1}	8.0889×10^{1}	9.0055×10^{1}	1.2071×10^{2}	8.6556×10^{1}
	Std	2.1080×10^{1}	2.8775×10^{1}	3.2281×10^{1}	2.6610×10^{1}	6.6972×10^{1}	6.0715×10^{1}	1.2499×10^{1}	2.9752×10^{1}
F18	Ave	2.3260×10^{3}	6.3085×10^{3}	1.7704×10^{4}	1.8023×10^{4}	9.4007×10^{3}	1.2028×10^{4}	2.2437×10^{5}	4.3292×10^{4}
	Std	1.9213×10^{3}	5.0238×10^{3}	1.5998×10^{4}	1.3809×10^{4}	1.2705×10^{4}	9.8545×10^{3}	2.0177×10^{5}	1.9469×10^{4}
F19	Ave	2.0035×10^{1}	8.5859×10^{1}	2.2923×10^{4}	1.2586×10^{4}	9.4871×10^{1}	1.6415×10^{4}	4.5056×10^{3}	6.8834×10^{3}
	Std	2.4216×10^{1}	6.3902×10^{1}	3.0517×10^{4}	8.4171×10^{3}	5.2162×10^{1}	1.7515×10^{4}	4.8081×10^{3}	1.2793×10^{4}
F20	Ave	1.5521×10^{2}	6.8049×10^{1}	1.4794×10^{2}	1.8825×10^{2}	4.8487×10^{1}	1.6317×10^{2}	1.1153×10^{2}	7.8101×10^{1}
	Std	2.1366×10^{1}	6.7754×10^{1}	6.4163×10^{1}	7.2922×10^{1}	4.5742×10^{1}	1.0410×10^{2}	1.5485×10^{1}	1.4538×10^{1}
F21	Ave	1.4092×10^{2}	1.7443×10^{2}	1.3315×10^{2}	2.0715×10^{2}	2.1262×10^{2}	2.2841×10^{2}	1.5340×10^{2}	2.4382×10^{2}
	Std	6.4054×10^{1}	6.3283×10^{1}	4.6371×10^{1}	7.2661×10^{1}	3.9512×10^{1}	6.7592×10^{1}	1.8180×10^{1}	5.1120
F22	Ave	1.0217×10^{2}	1.0225×10^{2}	1.0774×10^{2}	1.1585×10^{2}	1.1219×10^{2}	1.1030×10^{2}	2.5557×10^{2}	2.6957×10^{2}
	Std	7.0033×10^{-1}	8.3940×10^{-1}	4.2243×10^{1}	6.1962	1.9960×10^{1}	1.4630×10^{1}	4.8221×10^{1}	3.1870×10^{2}
F23	Ave	3.2115×10^{2}	3.2306×10^{2}	3.3466×10^{2}	3.7630×10^{2}	3.2937×10^{2}	3.4599×10^{2}	3.4818×10^{2}	3.3678×10^{2}
	Std	2.7270×10^{1}	7.2282	1.1937×10^{1}	4.1011×10^{1}	8.1281	2.2066×10^{1}	5.2873	9.1109
F24	Ave	1.9765×10^{2}	3.5918×10^{2}	3.0204×10^{2}	3.8149×10^{2}	3.6730×10^{2}	3.6028×10^{2}	3.4439×10^{2}	3.6771×10^{2}
	Std	1.2609×10^{2}	1.2612×10^{1}	1.1151×10^{2}	1.0415×10^{2}	8.4777	7.5941×10^{1}	4.9510×10^{1}	6.5852

Table 2. *Cont.*

Fun.	Meas.	KLEC-HBA	HBA	IHBA	HHO	SSA	WOA	ABC	PSO
F25	Ave	4.2199×10^{2}	4.2321×10^{2}	4.7667×10^{2}	4.2324×10^{2}	4.2682×10^{2}	4.4005×10^{2}	5.0092×10^{2}	4.5947×10^{2}
	Std	2.3937×10^{1}	2.7086×10^{1}	3.0529×10^{1}	2.4651×10^{1}	2.1019×10^{1}	2.7269×10^{1}	1.5376×10^{1}	1.2752×10^{1}
F26	Ave	3.5707×10^{2}	4.0275×10^{2}	5.2681×10^{2}	7.0004×10^{2}	3.9461×10^{2}	6.2748×10^{2}	6.1100×10^{2}	5.1092×10^{2}
	Std	1.9027×10^{2}	3.0870×10^{2}	1.5229×10^{2}	5.7163×10^{2}	9.0021×10^{1}	4.6460×10^{2}	5.9939×10^{1}	1.8745×10^{2}
F27	Ave	4.0211×10^{2}	4.0231×10^{2}	4.0780×10^{2}	4.4504×10^{2}	4.2604×10^{2}	4.4924×10^{2}	4.0750×10^{2}	4.1349×10^{2}
	Std	2.6658×10^{1}	1.9559×10^{1}	1.4750×10^{1}	3.2148×10^{1}	3.7068×10^{1}	4.4026×10^{1}	2.3033	2.5304×10^{1}
F28	Ave	4.9967×10^{2}	5.3382×10^{2}	5.0559×10^{2}	6.1705×10^{2}	5.9537×10^{2}	6.4111×10^{2}	5.3811×10^{2}	5.8367×10^{2}
	Std	5.1791×10^{-1}	1.8245×10^{2}	1.1683×10^{2}	1.9289×10^{1}	1.3926×10^{2}	1.7464×10^{2}	4.5065×10^{1}	1.4251×10^{2}
F29	Ave	2.9567×10^{2}	2.9572×10^{2}	3.2899×10^{2}	4.1525×10^{2}	3.3856×10^{2}	4.2566×10^{2}	3.8363×10^{2}	3.3409×10^{2}
	Std	3.8329×10^{1}	4.7869×10^{1}	5.4177×10^{1}	8.0072×10^{1}	7.7664×10^{1}	7.4123×10^{1}	3.2728×10^{1}	7.4942×10^{1}
F30	Ave	3.5043×10^{4}	1.8444×10^{6}	5.7684×10^{5}	1.3227×10^{6}	4.6678×10^{5}	2.2185×10^{5}	2.3510×10^{6}	7.2491×10^{5}
	Std	1.0655×10^{5}	3.9483×10^{6}	2.9630×10^{5}	2.2517×10^{6}	6.5064×10^{5}	3.6688×10^{5}	1.0048×10^{6}	1.3067×10^{6}
+/−/≈		-	22/7/0	22/7/0	26/3/0	23/6/0	27/2/0	26/3/0	25/4/0
R+/R−		-	320/115	364/71	408/27	348/87	422/13	411/24	384/51
***p*-Values**		-	0.009	0.009	<0.001	0.003	<0.001	<0.001	<0.001
α = 0.05		-	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Mean Rank		2.10	2.24	4.21	5.45	3.69	6.03	6.62	5.66
Total Rank		1	2	4	5	3	7	8	6

Table 3. Experimental results of algorithm comparison on the 30-dimensional CEC2017 test suite.

Fun.	Meas.	KLEC-HBA	HBA	IHBA	HHO	SSA	WOA	ABC	PSO
F1	Ave	7.2980×10^{3}	1.3739×10^{7}	4.3720×10^{9}	1.3739×10^{7}	9.0295×10^{9}	1.3529×10^{8}	3.8169×10^{10}	2.2206×10^{10}
	Std	6.8579×10^{3}	3.1168×10^{6}	4.4068×10^{9}	3.1168×10^{6}	7.1790×10^{9}	7.4151×10^{7}	2.7755×10^{9}	3.7228×10^{9}
F3	Ave	3.2248×10^{3}	1.6295×10^{4}	4.9311×10^{4}	1.6295×10^{4}	1.0246×10^{5}	2.2354×10^{5}	1.8204×10^{5}	1.3182×10^{5}
	Std	1.9539×10^{3}	4.5914×10^{3}	1.0964×10^{4}	4.5914×10^{3}	4.7681×10^{4}	5.9700×10^{4}	2.6150×10^{4}	3.6594×10^{4}
F4	Ave	9.5367×10^{1}	1.2255×10^{2}	1.0325×10^{3}	1.2255×10^{2}	5.4010×10^{2}	2.2348×10^{2}	5.4476×10^{3}	1.2234×10^{3}
	Std	1.3767×10^{1}	3.0402×10^{1}	5.2810×10^{2}	3.0402×10^{1}	5.5785×10^{2}	8.3580×10^{1}	9.5305×10^{2}	8.5814×10^{2}
F5	Ave	1.2650×10^{2}	2.3896×10^{2}	3.1275×10^{2}	2.3896×10^{2}	2.0899×10^{2}	2.9819×10^{2}	3.7936×10^{2}	3.0908×10^{2}
	Std	2.1663×10^{1}	3.1232×10^{1}	5.2974×10^{1}	3.1232×10^{1}	3.0718×10^{1}	4.6938×10^{1}	2.4075×10^{1}	1.9200×10^{1}
F6	Ave	6.1644	6.4253×10^{1}	6.6186×10^{1}	6.4253×10^{1}	3.6544×10^{1}	7.3604×10^{1}	7.5733×10^{1}	5.5267×10^{1}
	Std	3.9683	4.6040	1.1172×10^{1}	4.6040	1.0883×10^{1}	5.0456	8.2902	4.1248
F7	Ave	1.9161×10^{2}	5.5609×10^{2}	4.6400×10^{2}	5.5609×10^{2}	3.9203×10^{2}	4.7641×10^{2}	1.1969×10^{3}	9.2103×10^{2}
	Std	4.9850×10^{1}	5.5394×10^{1}	9.6715×10^{1}	5.5394×10^{1}	1.0472×10^{2}	7.9548×10^{1}	8.2810×10^{1}	4.2212×10^{1}
F8	Ave	1.0254×10^{2}	1.7847×10^{2}	2.2047×10^{2}	1.7847×10^{2}	1.9803×10^{2}	2.0151×10^{2}	3.6436×10^{2}	3.0045×10^{2}
	Std	2.6056×10^{1}	2.3766×10^{1}	2.1573×10^{1}	2.3766×10^{1}	4.8279×10^{1}	5.1543×10^{1}	1.2830×10^{1}	1.8364×10^{1}
F9	Ave	1.0298×10^{3}	6.2758×10^{3}	5.9830×10^{3}	6.2758×10^{3}	5.5451×10^{3}	9.3322×10^{3}	1.3915×10^{4}	8.2014×10^{3}
	Std	4.8145×10^{2}	1.1433×10^{3}	8.4040×10^{2}	1.1433×10^{3}	2.4690×10^{3}	5.0427×10^{3}	9.1227×10^{2}	2.2981×10^{3}
F10	Ave	4.5841×10^{3}	4.7103×10^{3}	5.7748×10^{3}	4.7103×10^{3}	4.1181×10^{3}	5.8566×10^{3}	7.4325×10^{3}	7.1813×10^{3}
	Std	9.9296×10^{2}	6.2534×10^{2}	1.0007×10^{3}	6.2534×10^{2}	6.8185×10^{2}	8.3184×10^{2}	4.2936×10^{2}	5.0753×10^{2}
F11	Ave	1.5477×10^{2}	1.7533×10^{2}	1.1531×10^{3}	1.7533×10^{2}	5.7933×10^{2}	2.7118×10^{3}	9.4205×10^{3}	2.2833×10^{3}
	Std	1.0339×10^{2}	2.2841×10^{1}	1.1497×10^{3}	2.2841×10^{1}	2.0158×10^{2}	2.2209×10^{3}	1.9195×10^{3}	5.3545×10^{2}
F12	Ave	1.3408×10^{5}	1.2939×10^{7}	6.0674×10^{8}	1.2939×10^{7}	1.2757×10^{9}	8.9692×10^{7}	3.4422×10^{9}	1.0930×10^{9}
	Std	1.7041×10^{5}	1.0509×10^{7}	3.7089×10^{8}	1.0509×10^{7}	1.8639×10^{9}	7.1521×10^{7}	5.1071×10^{8}	3.1980×10^{8}
F13	Ave	2.2608×10^{4}	4.0220×10^{5}	6.7587×10^{7}	4.0220×10^{5}	2.8593×10^{8}	2.1333×10^{5}	1.2469×10^{9}	2.7798×10^{8}
	Std	2.1046×10^{4}	1.4107×10^{5}	7.8565×10^{7}	1.4107×10^{5}	6.2596×10^{8}	1.1148×10^{5}	4.6528×10^{8}	7.3760×10^{7}
F14	Ave	1.1935×10^{4}	1.6314×10^{5}	6.3745×10^{5}	1.6314×10^{5}	2.0802×10^{5}	1.1044×10^{6}	1.4227×10^{6}	2.5117×10^{5}
	Std	1.2950×10^{4}	1.9656×10^{5}	6.8325×10^{5}	1.9656×10^{5}	5.1729×10^{5}	1.0981×10^{6}	8.6142×10^{5}	1.3888×10^{5}
F15	Ave	5.7961×10^{3}	5.0360×10^{4}	4.5587×10^{6}	5.0360×10^{4}	5.9488×10^{4}	1.6030×10^{5}	1.8932×10^{8}	4.5655×10^{7}
	Std	7.5264×10^{3}	2.5772×10^{4}	1.2989×10^{7}	2.5772×10^{4}	6.0069×10^{4}	9.4722×10^{4}	5.1674×10^{7}	3.5553×10^{7}
F16	Ave	1.0392×10^{3}	1.8117×10^{3}	1.8447×10^{3}	1.8117×10^{3}	1.3767×10^{3}	2.2616×10^{3}	2.5277×10^{3}	1.9708×10^{3}
	Std	2.2020×10^{2}	5.3022×10^{2}	4.0396×10^{2}	5.3022×10^{2}	3.3314×10^{2}	5.8853×10^{2}	1.1647×10^{2}	3.2015×10^{2}

Table 3. *Cont.*

Fun.	Meas.	KLEC-HBA	HBA	IHBA	HHO	SSA	WOA	ABC	PSO
F17	Ave	5.0567×10^{2}	1.0467×10^{3}	7.5646×10^{2}	1.0467×10^{3}	8.2532×10^{2}	8.5569×10^{2}	1.2629×10^{3}	9.9727×10^{2}
	Std	2.7865×10^{2}	2.7530×10^{2}	2.1298×10^{2}	2.7530×10^{2}	3.6604×10^{2}	2.9001×10^{2}	1.2019×10^{2}	2.6819×10^{2}
F18	Ave	2.3444×10^{5}	1.8728×10^{6}	5.6214×10^{6}	1.8728×10^{6}	2.1246×10^{6}	5.7881×10^{6}	2.5194×10^{7}	5.2096×10^{6}
	Std	2.3333×10^{5}	1.3356×10^{6}	5.4946×10^{6}	1.3356×10^{6}	3.8791×10^{6}	6.6532×10^{6}	6.8697×10^{6}	3.6498×10^{6}
F19	Ave	1.6821×10^{4}	3.7000×10^{5}	8.4946×10^{6}	3.7000×10^{5}	4.8701×10^{5}	6.7706×10^{6}	2.3135×10^{8}	9.2833×10^{7}
	Std	1.7429×10^{4}	2.2558×10^{5}	9.4258×10^{6}	2.2558×10^{5}	7.2158×10^{5}	6.8589×10^{6}	6.6043×10^{7}	5.0972×10^{7}
F20	Ave	6.1516×10^{2}	7.1874×10^{2}	6.2339×10^{2}	7.1874×10^{2}	6.9538×10^{2}	8.7600×10^{2}	8.9254×10^{2}	7.2860×10^{2}
	Std	1.8167×10^{2}	1.6813×10^{2}	1.6907×10^{2}	1.6813×10^{2}	2.0468×10^{2}	1.8032×10^{2}	9.1918×10^{1}	8.5689×10^{1}
F21	Ave	2.9654×10^{2}	4.5120×10^{2}	4.5334×10^{2}	4.5120×10^{2}	4.1778×10^{2}	4.7880×10^{2}	5.4523×10^{2}	4.7381×10^{2}
	Std	3.5123×10^{1}	4.4286×10^{1}	5.6779×10^{1}	4.4286×10^{1}	6.7069×10^{1}	4.6449×10^{1}	1.0540×10^{1}	1.8422×10^{1}
F22	Ave	1.5522×10^{3}	3.2906×10^{3}	1.4779×10^{3}	3.2906×10^{3}	4.2172×10^{3}	4.4278×10^{3}	6.2124×10^{3}	4.9427×10^{3}
	Std	3.1562×10^{3}	2.2354×10^{3}	1.5694×10^{3}	2.2354×10^{3}	4.5500×10^{2}	2.2874×10^{3}	5.5059×10^{2}	2.7902×10^{3}
F23	Ave	4.8800×10^{2}	8.3357×10^{2}	6.8902×10^{2}	8.3357×10^{2}	5.8856×10^{2}	7.9130×10^{2}	7.2894×10^{2}	7.4387×10^{2}
	Std	7.8847×10^{1}	1.3324×10^{2}	8.5036×10^{1}	1.3324×10^{2}	6.9208×10^{1}	8.5472×10^{1}	1.4888×10^{1}	9.8666×10^{1}
F24	Ave	6.0373×10^{2}	9.6474×10^{2}	7.3790×10^{2}	9.6474×10^{2}	6.4163×10^{2}	7.5843×10^{2}	7.6368×10^{2}	7.5712×10^{2}
	Std	1.3970×10^{2}	1.2883×10^{2}	1.1092×10^{2}	1.2883×10^{2}	4.9256×10^{1}	1.1482×10^{2}	2.2951×10^{1}	7.6578×10^{1}
F25	Ave	3.9512×10^{2}	4.1959×10^{2}	6.7386×10^{2}	4.1959×10^{2}	7.3445×10^{2}	5.0079×10^{2}	3.3497×10^{3}	1.3502×10^{3}
	Std	1.7268×10^{1}	2.1272×10^{1}	1.4372×10^{2}	2.1272×10^{1}	3.1356×10^{2}	4.6573×10^{1}	3.8495×10^{2}	2.0093×10^{2}
F26	Ave	1.7565×10^{3}	4.4311×10^{3}	4.5856×10^{3}	4.4311×10^{3}	3.7494×10^{3}	4.6455×10^{3}	5.4188×10^{3}	4.2580×10^{3}
	Std	1.1292×10^{3}	1.5723×10^{3}	1.1642×10^{3}	1.5723×10^{3}	5.9532×10^{2}	1.4817×10^{3}	2.7483×10^{2}	5.1866×10^{2}
F27	Ave	6.9276×10^{2}	7.0819×10^{2}	6.9048×10^{2}	7.0819×10^{2}	5.6134×10^{2}	6.7677×10^{2}	7.7850×10^{2}	6.8754×10^{2}
	Std	1.5056×10^{2}	1.2880×10^{2}	9.3186×10^{1}	1.2880×10^{2}	3.2687×10^{1}	1.2870×10^{2}	3.7375×10^{1}	1.0428×10^{2}
F28	Ave	4.1933×10^{2}	4.6444×10^{2}	7.3421×10^{2}	4.6444×10^{2}	1.7417×10^{3}	5.7952×10^{2}	3.2178×10^{3}	1.1312×10^{3}
	Std	2.5533×10^{1}	2.8214×10^{1}	1.3310×10^{2}	2.8214×10^{1}	9.1930×10^{2}	2.6972×10^{1}	3.0450×10^{2}	1.3870×10^{2}
F29	Ave	1.1815×10^{3}	1.5673×10^{3}	1.9009×10^{3}	1.5673×10^{3}	1.2042×10^{3}	2.0417×10^{3}	2.6328×10^{3}	1.4785×10^{3}
	Std	6.3447×10^{2}	2.9906×10^{2}	4.0212×10^{2}	2.9906×10^{2}	2.0928×10^{2}	3.8451×10^{2}	1.8158×10^{2}	2.9881×10^{2}
F30	Ave	2.2593×10^{4}	1.9793×10^{6}	5.9013×10^{7}	1.9793×10^{6}	2.7760×10^{6}	1.4958×10^{7}	1.8970×10^{8}	3.6704×10^{7}
	Std	2.2984×10^{4}	1.0139×10^{6}	5.9297×10^{7}	1.0139×10^{6}	4.7537×10^{6}	1.3560×10^{7}	2.4845×10^{7}	1.9579×10^{7}
+/−/≈		-	29/0/0	27/2/0	29/0/0	27/2/0	28/1/0	29/0/0	28/1/0
R+/R−		-	435/0	430/5	435/0	410/25	434/1	435/0	434/1
***p*-Values**		-	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
α = 0.05		-	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Mean Rank		1.21	3.78	4.76	3.78	3.59	5.45	7.76	5.69
Total Rank		1	3	5	3	2	6	8	7

Table 4. Experimental results of algorithm comparison on the 50-dimensional CEC2017 test suite.

Fun.	Meas.	KLEC-HBA	HBA	IHBA	HHO	SSA	WOA	ABC	PSO
F1	Ave	5.1539×10^{3}	1.6177×10^{4}	1.3543×10^{10}	7.7606×10^{7}	4.0124×10^{10}	1.2791×10^{9}	1.1728×10^{11}	5.9501×10^{10}
	Std	6.7184×10^{3}	1.3106×10^{4}	1.1687×10^{10}	1.0225×10^{7}	1.7217×10^{10}	5.6288×10^{8}	5.6023×10^{9}	7.7900×10^{9}
F3	Ave	2.6597×10^{4}	4.6334×10^{4}	1.1154×10^{5}	6.3561×10^{4}	2.9828×10^{5}	2.2167×10^{5}	3.5247×10^{5}	2.9147×10^{5}
	Std	8.8074×10^{3}	9.9173×10^{3}	2.0014×10^{4}	1.2315×10^{4}	7.3982×10^{4}	7.2251×10^{4}	3.8831×10^{4}	4.6033×10^{4}
F4	Ave	1.2341×10^{2}	1.2316×10^{2}	2.5414×10^{3}	2.9730×10^{2}	3.4286×10^{3}	8.9267×10^{2}	2.6002×10^{4}	5.9443×10^{3}
	Std	6.9654×10^{1}	6.5450×10^{1}	1.4938×10^{3}	9.3079×10^{1}	1.8119×10^{3}	2.4506×10^{2}	4.4351×10^{3}	1.4034×10^{3}
F5	Ave	3.3461×10^{2}	2.2792×10^{2}	4.3592×10^{2}	3.6770×10^{2}	4.7912×10^{2}	4.7598×10^{2}	7.7672×10^{2}	6.2418×10^{2}
	Std	2.0480×10^{1}	3.6141×10^{1}	3.2593×10^{1}	2.7779×10^{1}	6.3322×10^{1}	8.8742×10^{1}	2.7828×10^{1}	2.3471×10^{1}
F6	Ave	2.7579×10^{1}	2.0126×10^{1}	7.8895×10^{1}	7.2505×10^{1}	5.5409×10^{1}	8.8986×10^{1}	1.0619×10^{2}	7.2802×10^{1}
	Std	3.6316	4.6266	4.2091	6.3935	9.7683	1.3825×10^{1}	4.8434	6.1592
F7	Ave	3.2045×10^{2}	4.3773×10^{2}	1.0067×10^{3}	1.0999×10^{3}	1.0750×10^{3}	1.1300×10^{3}	3.1972×10^{3}	2.3024×10^{3}
	Std	2.0818×10^{2}	9.7036×10^{1}	1.2233×10^{2}	8.6021×10^{1}	3.1010×10^{2}	8.6956×10^{1}	1.2596×10^{2}	1.6389×10^{2}
F8	Ave	2.2097×10^{2}	2.2213×10^{2}	4.7994×10^{2}	3.8414×10^{2}	4.8558×10^{2}	4.8439×10^{2}	8.0310×10^{2}	6.4126×10^{2}
	Std	3.0750×10^{1}	4.2876×10^{1}	5.2173×10^{1}	2.6335×10^{1}	1.0058×10^{2}	8.5808×10^{1}	2.5696×10^{1}	4.0558×10^{1}

Table 4. *Cont.*

Fun.	Meas.	KLEC-HBA	HBA	IHBA	HHO	SSA	WOA	ABC	PSO
F9	Ave	1.0205×10^{4}	5.6756×10^{3}	1.8900×10^{4}	2.4148×10^{4}	2.0188×10^{4}	3.0238×10^{4}	4.8976×10^{4}	2.5037×10^{4}
	Std	2.1538×10^{2}	2.4815×10^{3}	2.5000×10^{3}	4.1827×10^{3}	5.9816×10^{3}	1.0395×10^{4}	4.5430×10^{3}	4.1739×10^{3}
F10	Ave	6.3731×10^{3}	7.4086×10^{3}	1.0279×10^{4}	8.0067×10^{3}	7.4901×10^{3}	9.9051×10^{3}	1.4176×10^{4}	1.3673×10^{4}
	Std	8.0175×10^{2}	1.8416×10^{3}	1.1046×10^{3}	7.4016×10^{2}	8.6342×10^{2}	1.4592×10^{3}	1.6223×10^{2}	4.9917×10^{2}
F11	Ave	2.5737×10^{2}	2.9063×10^{2}	3.9251×10^{3}	3.9041×10^{2}	7.7119×10^{3}	1.3865×10^{3}	3.7431×10^{4}	1.0941×10^{4}
	Std	7.0093×10^{1}	8.7993×10^{1}	2.0574×10^{3}	8.6415×10^{1}	1.3688×10^{4}	2.9900×10^{2}	5.3693×10^{3}	3.2620×10^{3}
F12	Ave	7.5861×10^{5}	2.1760×10^{6}	3.5603×10^{9}	8.9213×10^{7}	8.9221×10^{9}	7.1992×10^{8}	2.6742×10^{10}	1.3505×10^{10}
	Std	8.3321×10^{5}	9.5019×10^{5}	2.9673×10^{9}	3.6366×10^{7}	5.8526×10^{9}	4.0636×10^{8}	2.9490×10^{9}	2.6185×10^{9}
F13	Ave	1.5632×10^{4}	2.1539×10^{4}	6.0070×10^{8}	5.7262×10^{6}	2.5581×10^{9}	1.5194×10^{7}	8.2333×10^{9}	4.6522×10^{9}
	Std	9.8629×10^{3}	1.4257×10^{4}	4.3752×10^{8}	6.8766×10^{6}	2.7861×10^{9}	1.8263×10^{7}	1.8070×10^{9}	2.6543×10^{9}
F14	Ave	5.0368×10^{4}	9.4355×10^{4}	3.1077×10^{6}	8.8073×10^{5}	2.3533×10^{6}	1.9527×10^{6}	1.5310×10^{7}	2.9956×10^{6}
	Std	4.6939×10^{4}	6.5019×10^{4}	1.8637×10^{6}	6.2876×10^{5}	3.8753×10^{6}	1.0557×10^{6}	4.7558×10^{6}	1.5928×10^{6}
F15	Ave	1.5357×10^{4}	3.5535×10^{4}	4.3504×10^{7}	5.0346×10^{5}	9.0902×10^{7}	3.8278×10^{6}	2.1820×10^{9}	5.8034×10^{8}
	Std	8.7724×10^{3}	3.9542×10^{4}	7.5218×10^{7}	2.6379×10^{5}	1.7402×10^{8}	5.4880×10^{6}	6.0278×10^{8}	2.4626×10^{8}
F16	Ave	1.8047×10^{3}	1.9970×10^{3}	3.3062×10^{3}	2.5792×10^{3}	2.6980×10^{3}	3.5938×10^{3}	5.1114×10^{3}	4.1046×10^{3}
	Std	3.3163×10^{2}	4.2447×10^{2}	5.3332×10^{2}	6.2295×10^{2}	7.8164×10^{2}	4.9180×10^{2}	2.5448×10^{2}	3.9625×10^{2}
F17	Ave	1.2341×10^{3}	1.3316×10^{3}	2.3403×10^{3}	1.9890×10^{3}	2.2745×10^{3}	2.3832×10^{3}	5.1760×10^{3}	3.8869×10^{3}
	Std	4.7166×10^{2}	5.1187×10^{2}	4.5552×10^{2}	6.1292×10^{2}	3.2451×10^{2}	3.6772×10^{2}	5.0865×10^{2}	5.1542×10^{2}
F18	Ave	4.6262×10^{5}	5.3384×10^{5}	1.5850×10^{7}	4.9735×10^{6}	7.8286×10^{6}	2.0334×10^{7}	9.1956×10^{7}	2.2967×10^{7}
	Std	4.7805×10^{5}	5.0778×10^{5}	1.5755×10^{7}	4.3020×10^{6}	1.0987×10^{7}	2.2646×10^{7}	3.9355×10^{7}	1.1785×10^{7}
F19	Ave	1.5060×10^{4}	1.7821×10^{4}	8.4431×10^{7}	1.2393×10^{6}	1.2799×10^{8}	5.7535×10^{6}	9.3660×10^{8}	3.3294×10^{8}
	Std	1.1295×10^{4}	1.3398×10^{4}	1.7561×10^{8}	8.7107×10^{5}	3.9402×10^{8}	6.1485×10^{6}	2.9793×10^{8}	1.6604×10^{8}
F20	Ave	1.2043×10^{3}	1.2342×10^{3}	1.5544×10^{3}	1.5267×10^{3}	1.5177×10^{3}	1.6778×10^{3}	2.2248×10^{3}	2.2375×10^{3}
	Std	3.0128×10^{2}	4.3781×10^{2}	2.7163×10^{2}	2.9343×10^{2}	2.0557×10^{2}	4.8109×10^{2}	1.6505×10^{2}	2.0352×10^{2}
F21	Ave	4.0067×10^{2}	4.1993×10^{2}	7.2212×10^{2}	7.9565×10^{2}	6.5654×10^{2}	8.7656×10^{2}	9.5230×10^{2}	8.0939×10^{2}
	Std	6.0917×10^{1}	4.5998×10^{1}	7.5298×10^{1}	1.0029×10^{2}	6.9249×10^{1}	9.5192×10^{1}	3.7069×10^{1}	3.0585×10^{1}
F22	Ave	8.2424×10^{3}	8.4832×10^{3}	9.5816×10^{3}	8.9498×10^{3}	9.3460×10^{3}	1.0932×10^{4}	1.4407×10^{4}	1.4279×10^{4}
	Std	6.9477×10^{2}	1.4739×10^{3}	2.1497×10^{3}	9.5511×10^{2}	1.2511×10^{3}	1.5999×10^{3}	5.3636×10^{2}	4.5282×10^{2}
F23	Ave	7.3278×10^{2}	7.6595×10^{2}	1.2666×10^{3}	1.4665×10^{3}	9.9961×10^{2}	1.3865×10^{3}	1.2540×10^{3}	1.3055×10^{3}
	Std	8.1820×10^{1}	6.7437×10^{1}	1.6609×10^{2}	1.8328×10^{2}	1.8515×10^{2}	1.3817×10^{2}	8.4956×10^{1}	1.6329×10^{2}
F24	Ave	8.7134×10^{2}	8.7639×10^{2}	1.1647×10^{3}	1.8439×10^{3}	1.0291×10^{3}	1.3967×10^{3}	1.2857×10^{3}	1.3124×10^{3}
	Std	2.7359×10^{2}	2.5454×10^{2}	5.8138×10^{1}	1.9814×10^{2}	9.5492×10^{1}	2.0414×10^{2}	1.6224×10^{1}	1.6390×10^{2}
F25	Ave	5.7235×10^{2}	5.8505×10^{2}	2.3005×10^{3}	6.6497×10^{2}	1.9605×10^{3}	1.0030×10^{3}	1.8398×10^{4}	5.0673×10^{3}
	Std	5.7866×10^{1}	3.5794×10^{1}	6.8961×10^{2}	3.8412×10^{1}	1.1819×10^{3}	9.3515×10^{1}	2.9520×10^{3}	7.6133×10^{2}
F26	Ave	2.3095×10^{3}	3.1879×10^{3}	9.5933×10^{3}	6.7757×10^{3}	7.0849×10^{3}	1.2457×10^{4}	1.0813×10^{4}	8.3570×10^{3}
	Std	3.2331×10^{3}	2.2422×10^{3}	8.1490×10^{2}	2.4541×10^{3}	1.1135×10^{3}	1.1111×10^{3}	4.8498×10^{2}	7.6879×10^{2}
F27	Ave	5.0001×10^{2}	1.3397×10^{3}	1.7620×10^{3}	1.6120×10^{3}	9.8154×10^{2}	2.1149×10^{3}	2.0177×10^{3}	1.3138×10^{3}
	Std	5.3864×10^{-5}	5.6414×10^{2}	3.7863×10^{2}	3.7125×10^{2}	1.2628×10^{2}	5.2624×10^{2}	1.1766×10^{2}	1.8783×10^{2}
F28	Ave	4.9735×10^{2}	1.3298×10^{3}	2.9716×10^{3}	6.7312×10^{2}	5.0233×10^{3}	1.5480×10^{3}	9.2954×10^{3}	4.2780×10^{3}
	Std	8.4096	2.5700×10^{3}	9.3126×10^{2}	3.4942×10^{1}	2.1493×10^{3}	2.8732×10^{2}	9.5336×10^{2}	8.2609×10^{2}
F29	Ave	1.2722×10^{3}	1.9161×10^{3}	3.9291×10^{3}	2.8588×10^{3}	2.8996×10^{3}	4.9678×10^{3}	7.4549×10^{3}	4.1295×10^{3}
	Std	2.6568×10^{2}	9.4159×10^{2}	5.3168×10^{2}	5.3530×10^{2}	1.0686×10^{3}	1.0108×10^{3}	1.5262×10^{3}	4.2957×10^{2}
F30	Ave	1.7115×10^{6}	2.3148×10^{6}	2.6155×10^{8}	3.4087×10^{7}	2.0281×10^{8}	2.3085×10^{8}	2.0035×10^{9}	6.6872×10^{8}
	Std	9.0876×10^{5}	8.7511×10^{5}	1.1868×10^{8}	1.1113×10^{7}	2.4869×10^{8}	8.8802×10^{7}	4.1545×10^{8}	2.5811×10^{8}
+/−/≈		-	25/4/0	29/0/0	29/0/0	29/0/0	29/0/0	29/0/0	29/0/0
R+/R−		-	398/37	435/0	435/0	435/0	435/0	435/0	435/0
***p*-Values**		-	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
α = 0.05		-	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Mean Rank		1.14	1.97	4.97	3.69	4.62	5.48	7.66	6.48
Total Rank		1	2	5	3	4	6	8	7

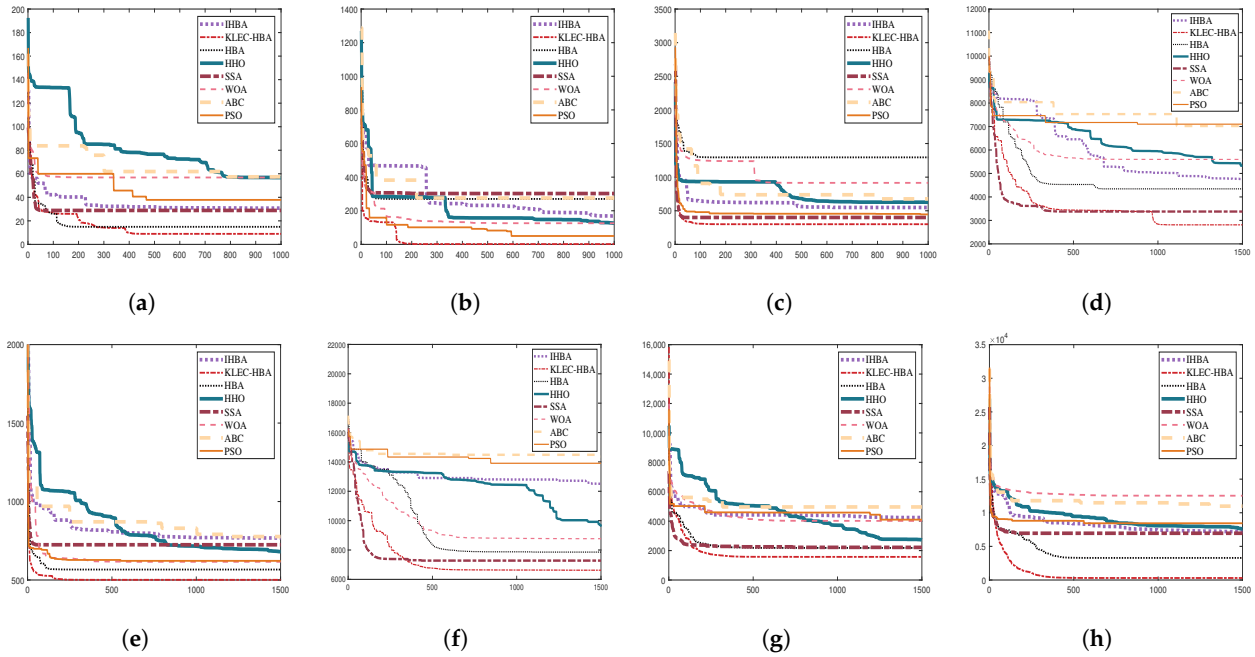


Figure 4. Convergence curves of the algorithm on comparative experiments. (**a**) Fun5-10dim. (**b**) Fun16-10dim. (**c**) Fun26-10dim. (**d**) Fun10-30dim. (**e**) Fun27-30dim. (**f**) Fun10-50dim. (**g**) Fun16-50dim. (**h**) Fun26-50dim.

4.4.2. Ablation Experiment

To ascertain the efficacy of each improvement to the KLEC-HBA, ablation experiments are conducted in this section on 10-, 30-, and 50-dimensional CEC2017. Each experiment is run independently 30 times. In this context, "KLEC-HBA-LR" denotes that it is identical to the KLEC-HBA, except for the "learning rate LR = 0.2" parameter. Similarly, "KLEC-HBA-LOS" denotes that the loser learning strategy has been removed from the KLEC-HBA. Furthermore, "KLEC-HBA-M" denotes that the adaptive mutation strategy has been removed from the KLEC-HBA. Furthermore, the experimental results were subjected to a Friedman rank sum test to elucidate the relative merits and shortcomings of the performance of the proposed algorithms.

Table 5 illustrates the outcomes of ablation experiments conducted on the 10-dimensional CEC2017 test suite. Similarly, Tables 6 and 7 present the results of ablation experiments on the 30-dimensional and 50-dimensional CEC2017 test suites, respectively. As can be observed in Tables 5–7, the mean and overall values of the KLEC-HBA's rank on the 10-dimensional, 30-dimensional, and 50-dimensional CEC2017 test suites are as follows: the test suites are optimal, and the majority of the mean values of fitness on the 29 functions are relative to "KLEC-HBA-LR", "KLEC-HBA-LOS" and "KLEC-HBA-M", indicating that each improvement to the algorithm proposed in this chapter is effective. Furthermore, the efficacy of each enhancement made to the KLEC-HBA can be gauged by referencing Figure 5.

Table 5. Results of ablation experiments on the 10-dimensional CEC2017 test suite.

Fun.	Meas.	KLEC-HBA-LR	KLEC-HBA	KLEC-HBA-LOS	KLEC-HBA-M
F1	Ave	1.8203×10^{3}	3.1559×10^{3}	1.8335×10^{3}	3.5638×10^{3}
	Std	1.5792×10^{3}	2.7500×10^{3}	1.5514×10^{3}	3.9791×10^{3}
F3	Ave	7.1623×10^{-13}	3.0695×10^{-13}	3.0695×10^{-13}	2.3874×10^{-13}
	Std	1.2209×10^{-12}	1.8681×10^{-13}	2.7438×10^{-13}	1.2323×10^{-13}

Table 5. *Cont.*

Fun.	Meas.	KLEC-HBA-LR	KLEC-HBA	KLEC-HBA-LOS	KLEC-HBA-M
F4	Ave	9.1716×10^{-2}	3.7333×10^{-2}	8.1877×10^{-3}	7.0311×10^{-2}
	Std	1.1613×10^{-1}	7.5935×10^{-2}	7.6050×10^{-3}	1.1063×10^{-1}
F5	Ave	7.7009×10^{1}	5.4918×10^{1}	5.0543×10^{1}	6.6960×10^{1}
	Std	5.4616×10^{1}	3.4870×10^{1}	2.8189×10^{1}	3.9989×10^{1}
F6	Ave	2.6634×10^{1}	1.3020×10^{1}	2.2977×10^{1}	3.8618×10^{1}
	Std	1.4189×10^{1}	1.5499×10^{1}	1.7364×10^{1}	1.1611×10^{1}
F7	Ave	8.5555×10^{1}	7.0324×10^{1}	4.4271×10^{1}	6.2290×10^{1}
	Std	1.7012×10^{1}	3.7529×10^{1}	2.7661×10^{1}	3.1140×10^{1}
F8	Ave	3.2038×10^{1}	3.2237×10^{1}	3.5420×10^{1}	2.5670×10^{1}
	Std	6.2294	5.1025	1.3349	6.4250
F9	Ave	4.5898×10^{2}	3.7687×10^{2}	4.3554×10^{2}	5.6557×10^{2}
	Std	3.9628×10^{2}	2.6303×10^{2}	3.4306×10^{2}	4.3974×10^{2}
F10	Ave	7.3922×10^{2}	5.3335×10^{2}	8.7803×10^{2}	3.9646×10^{2}
	Std	6.8557×10^{1}	4.9396×10^{2}	2.5442×10^{2}	2.7179×10^{2}
F11	Ave	2.8447×10^{1}	4.9625	3.7889	1.4272×10^{1}
	Std	5.0199×10^{1}	9.7376	3.0130	2.2502×10^{1}
F12	Ave	8.9925×10^{3}	6.4094×10^{3}	6.7265×10^{3}	5.8127×10^{3}
	Std	6.5167×10^{3}	4.3040×10^{3}	5.1111×10^{3}	3.7434×10^{3}
F13	Ave	3.8071×10^{2}	4.6452×10^{2}	7.6082×10^{2}	8.2035×10^{2}
	Std	2.6578×10^{2}	1.2938×10^{2}	6.2509×10^{2}	5.1097×10^{2}
F14	Ave	5.2383×10^{1}	4.4235×10^{1}	5.0276×10^{1}	5.3599×10^{1}
	Std	2.2339×10^{1}	2.7903×10^{1}	1.9946×10^{1}	2.8696×10^{1}
F15	Ave	4.1887×10^{1}	3.8629×10^{1}	5.2714×10^{1}	4.6228×10^{1}
	Std	4.0849×10^{1}	3.0408×10^{1}	6.1523×10^{1}	5.2871×10^{1}
F16	Ave	3.8015×10^{2}	3.1692×10^{2}	3.6671×10^{2}	3.7325×10^{2}
	Std	1.0152×10^{2}	1.1046×10^{2}	8.8550×10^{1}	9.2425×10^{1}
F17	Ave	4.6603×10^{1}	4.4151×10^{1}	4.9229×10^{1}	5.8474×10^{1}
	Std	2.2107×10^{1}	2.5820×10^{1}	3.6392×10^{1}	2.8283×10^{1}
F18	Ave	7.3714×10^{3}	3.9155×10^{3}	6.5070×10^{3}	8.9575×10^{2}
	Std	1.0853×10^{4}	2.3170×10^{3}	7.0051×10^{3}	8.8734×10^{2}
F19	Ave	1.1505×10^{1}	1.5589×10^{1}	2.5397×10^{1}	5.3048×10^{1}
	Std	4.4746	1.1767×10^{1}	2.4674×10^{1}	4.2735×10^{1}
F20	Ave	1.8189×10^{2}	1.5361×10^{2}	1.8328×10^{2}	1.6455×10^{2}
	Std	4.1157×10^{1}	7.0264×10^{1}	3.1383×10^{1}	2.0447×10^{1}
F21	Ave	1.7319×10^{2}	1.7633×10^{2}	1.8231×10^{2}	1.4752×10^{2}
	Std	6.5966×10^{1}	6.9902×10^{1}	7.3615×10^{1}	6.5105×10^{1}
F22	Ave	1.0156×10^{2}	1.0200×10^{2}	1.0170×10^{2}	1.0116×10^{2}
	Std	6.2500×10^{-1}	8.5867×10^{-1}	5.2302×10^{-1}	5.4186×10^{-1}
F23	Ave	3.1766×10^{2}	3.1474×10^{2}	3.4218×10^{2}	3.2147×10^{2}
	Std	5.3944	1.5341×10^{1}	2.9891×10^{1}	9.3365
F24	Ave	2.0174×10^{2}	2.4826×10^{2}	3.0287×10^{2}	2.7800×10^{2}
	Std	1.3932×10^{2}	1.3535×10^{2}	1.1365×10^{2}	1.6869×10^{2}
F25	Ave	4.2573×10^{2}	4.2599×10^{2}	4.2572×10^{2}	4.3483×10^{2}
	Std	2.4706×10^{1}	2.4108×10^{1}	2.5550×10^{1}	2.0756×10^{1}
F26	Ave	2.5084×10^{2}	3.4409×10^{2}	5.9088×10^{2}	4.9500×10^{2}
	Std	7.2213×10^{1}	8.2287×10^{2}	6.6855×10^{2}	2.9882×10^{2}

Table 5. *Cont.*

Fun.	Meas.	KLEC-HBA-LR	KLEC-HBA	KLEC-HBA-LOS	KLEC-HBA-M
F27	Ave	3.9228×10^{2}	3.9879×10^{2}	3.9527×10^{2}	3.9270×10^{2}
	Std	2.4562	9.2604	7.2670	3.2180
F28	Ave	4.9943×10^{2}	4.9912×10^{2}	5.3260×10^{2}	4.9982×10^{2}
	Std	7.4425×10^{-1}	7.8534×10^{-1}	1.3060×10^{2}	2.1066×10^{-1}
F29	Ave	2.9509×10^{2}	2.8929×10^{2}	3.3130×10^{2}	3.0757×10^{2}
	Std	3.8615×10^{1}	3.8219×10^{1}	6.5885×10^{1}	4.7770×10^{1}
F30	Ave	1.6231×10^{3}	1.1690×10^{3}	7.9007×10^{2}	1.8316×10^{3}
	Std	2.1496×10^{3}	4.4567×10^{2}	3.7130×10^{2}	1.9990×10^{3}
+/−/≈		19/10/0	-	19/9/1	20/9/0
Mean Rank		2.55	1.98	2.74	2.72
Total Rank		2	1	4	3

Table 6. Results of ablation experiments on the 30-dimensional CEC2017 test suite.

Fun.	Meas.	KLEC-HBA-LR	KLEC-HBA	KLEC-HBA-LOS	KLEC-HBA-M
F1	Ave	6.4220×10^{3}	4.9761×10^{3}	4.2706×10^{3}	5.4101×10^{3}
	Std	5.8702×10^{3}	3.3086×10^{3}	5.4120×10^{3}	5.4818×10^{3}
F3	Ave	4.9015×10^{2}	1.6539×10^{1}	8.7121	2.7835×10^{2}
	Std	7.0915×10^{2}	1.8233×10^{1}	8.0106	5.6325×10^{2}
F4	Ave	9.3162×10^{1}	8.8849×10^{1}	7.5528×10^{1}	7.7177×10^{1}
	Std	5.2871×10^{1}	1.0880×10^{1}	3.8975×10^{1}	3.6796×10^{1}
F5	Ave	2.2282×10^{2}	1.9517×10^{2}	2.3043×10^{2}	2.2290×10^{2}
	Std	6.2478×10^{1}	5.4120×10^{1}	5.5004×10^{1}	3.6002×10^{1}
F6	Ave	4.9791×10^{1}	4.0714×10^{1}	3.9870×10^{1}	4.6278×10^{1}
	Std	5.0617	5.0428	1.3016×10^{1}	8.8454
F7	Ave	2.4644×10^{2}	3.0004×10^{2}	3.6099×10^{2}	2.7747×10^{2}
	Std	4.1360×10^{1}	1.3162×10^{2}	1.0611×10^{2}	1.5402×10^{2}
F8	Ave	1.4785×10^{2}	1.5680×10^{2}	1.4646×10^{2}	1.6180×10^{2}
	Std	2.0092×10^{1}	2.7108×10^{1}	2.1379×10^{1}	2.6804×10^{1}
F9	Ave	4.2850×10^{3}	4.3984×10^{3}	4.2398×10^{3}	4.2135×10^{3}
	Std	1.6749×10^{2}	1.9432×10^{2}	1.5667×10^{2}	2.2874×10^{2}
F10	Ave	4.0630×10^{3}	4.0389×10^{3}	4.1905×10^{3}	4.0846×10^{3}
	Std	2.6007×10^{2}	1.5240×10^{2}	1.2720×10^{2}	1.6496×10^{2}
F11	Ave	1.4422×10^{2}	1.2245×10^{2}	1.2247×10^{2}	8.0269×10^{1}
	Std	7.2644×10^{1}	2.6398×10^{1}	7.3484×10^{1}	3.9725×10^{1}
F12	Ave	3.3164×10^{4}	4.2497×10^{4}	4.8350×10^{4}	4.0982×10^{4}
	Std	2.1060×10^{4}	2.5596×10^{4}	1.8589×10^{4}	1.1251×10^{4}
F13	Ave	4.2413×10^{4}	1.4190×10^{4}	2.6131×10^{4}	2.3182×10^{4}
	Std	2.5687×10^{4}	1.5540×10^{4}	1.6960×10^{4}	2.1305×10^{4}
F14	Ave	2.8949×10^{3}	1.5099×10^{3}	2.0838×10^{3}	3.6882×10^{3}
	Std	3.1707×10^{3}	1.3350×10^{3}	2.5385×10^{3}	6.0957×10^{3}
F15	Ave	3.3073×10^{3}	3.3053×10^{3}	6.9196×10^{3}	4.2014×10^{3}
	Std	3.5987×10^{3}	3.0784×10^{3}	6.3629×10^{3}	5.0520×10^{3}
F16	Ave	1.1889×10^{3}	6.6950×10^{2}	9.3125×10^{2}	9.6064×10^{2}
	Std	2.1696×10^{2}	7.9748×10^{1}	3.6378×10^{2}	3.6596×10^{2}
F17	Ave	4.0733×10^{2}	4.1903×10^{2}	4.4774×10^{2}	4.2388×10^{2}
	Std	2.9986×10^{2}	1.4817×10^{2}	1.5898×10^{2}	1.2953×10^{2}

Table 6. *Cont.*

Fun.	Meas.	KLEC-HBA-LR	KLEC-HBA	KLEC-HBA-LOS	KLEC-HBA-M
F18	Ave	6.1096×10^{4}	6.1531×10^{4}	5.1560×10^{4}	7.2356×10^{4}
	Std	3.1037×10^{4}	3.3684×10^{4}	2.4250×10^{4}	4.0215×10^{4}
F19	Ave	4.3462×10^{3}	2.2561×10^{3}	2.3613×10^{3}	2.2617×10^{3}
	Std	3.3475×10^{3}	1.2253×10^{3}	1.2218×10^{3}	8.6012×10^{2}
F20	Ave	5.9647×10^{2}	4.8808×10^{2}	6.3206×10^{2}	5.1524×10^{2}
	Std	2.4350×10^{2}	1.1106×10^{2}	1.9619×10^{2}	1.7279×10^{2}
F21	Ave	3.1946×10^{2}	2.8222×10^{2}	4.2262×10^{2}	2.9354×10^{2}
	Std	1.4136×10^{2}	3.2298×10^{1}	6.6894×10^{1}	7.7105×10^{1}
F22	Ave	1.0555×10^{3}	1.0077×10^{3}	3.8190×10^{3}	1.1892×10^{3}
	Std	2.1333×10^{3}	2.0269×10^{3}	2.1342×10^{3}	2.2970×10^{3}
F23	Ave	5.0208×10^{2}	4.8424×10^{2}	4.9789×10^{2}	4.4043×10^{2}
	Std	4.3479×10^{1}	3.9596×10^{1}	3.8153×10^{1}	3.6706×10^{1}
F24	Ave	4.5815×10^{2}	5.9495×10^{2}	6.3960×10^{2}	5.1768×10^{2}
	Std	2.0568×10^{2}	6.6514×10^{1}	1.3556×10^{2}	2.5455×10^{1}
F25	Ave	3.9502×10^{2}	3.8775×10^{2}	3.8671×10^{2}	3.8925×10^{2}
	Std	2.1260×10^{1}	4.6719	2.7585	7.7958
F26	Ave	9.6258×10^{2}	2.8000×10^{2}	3.0000×10^{2}	4.9742×10^{2}
	Std	1.5381×10^{3}	4.4721×10^{1}	3.7106×10^{-4}	6.9581×10^{2}
F27	Ave	5.0001×10^{2}	5.0001×10^{2}	5.3561×10^{2}	5.0001×10^{2}
	Std	8.3430×10^{-5}	8.9130×10^{-5}	2.6340×10^{1}	7.0758×10^{-5}
F28	Ave	3.6217×10^{2}	3.8462×10^{2}	3.5962×10^{2}	4.3811×10^{2}
	Std	9.0684×10^{1}	5.1033×10^{1}	7.8795×10^{1}	6.5353×10^{1}
F29	Ave	7.9131×10^{2}	9.1186×10^{2}	8.8326×10^{2}	8.3565×10^{2}
	Std	2.3716×10^{2}	2.1150×10^{2}	2.8548×10^{2}	2.0446×10^{2}
F30	Ave	1.3061×10^{4}	1.0955×10^{4}	6.5056×10^{3}	1.2051×10^{4}
	Std	1.6554×10^{3}	7.8790×10^{3}	1.4813×10^{4}	9.8243×10^{3}
+/−/≈		19/9/1	-	18/11/1	20/8/1
Mean Rank		3.33	2.47	3.21	3.09
Total Rank		4	1	3	2

Table 7. Results of ablation experiments on the 50-dimensional CEC2017 test suite.

Fun.	Meas.	KLEC-HBA-LR	KLEC-HBA	KLEC-HBA-LOS	KLEC-HBA-M
F1	Ave	6.7451×10^{3}	3.1394×10^{3}	7.8807×10^{3}	4.5539×10^{3}
	Std	3.8730×10^{3}	1.0304×10^{4}	8.1994×10^{3}	2.5422×10^{3}
F3	Ave	3.1278×10^{4}	2.7833×10^{4}	3.0661×10^{4}	5.0589×10^{4}
	Std	7.9956×10^{3}	8.9704×10^{3}	7.3270×10^{3}	1.4143×10^{4}
F4	Ave	1.8467×10^{2}	8.3658×10^{1}	1.4174×10^{2}	1.6345×10^{2}
	Std	4.6560×10^{1}	5.8719×10^{1}	4.9326×10^{1}	5.1945×10^{1}
F5	Ave	3.5331×10^{2}	3.5769×10^{2}	3.4764×10^{2}	3.6236×10^{2}
	Std	3.4804×10^{1}	1.7198×10^{1}	2.6433×10^{1}	1.7020×10^{1}
F6	Ave	5.6315×10^{1}	5.2391×10^{1}	5.6049×10^{1}	5.4630×10^{1}
	Std	6.1957	2.4335	6.3027	5.3508
F7	Ave	8.4450×10^{2}	8.2205×10^{2}	8.0860×10^{2}	7.5503×10^{2}
	Std	1.1859×10^{2}	1.2276×10^{2}	1.9621×10^{2}	2.3436×10^{2}
F8	Ave	3.7238×10^{2}	3.7868×10^{2}	3.8743×10^{2}	3.8079×10^{2}
	Std	1.4240×10^{1}	2.8581×10^{1}	1.6860×10^{1}	4.5359×10^{1}

Table 7. *Cont.*

Fun.	Meas.	KLEC-HBA-LR	KLEC-HBA	KLEC-HBA-LOS	KLEC-HBA-M
F9	Ave	1.2197×10^{4}	1.2167×10^{4}	1.2297×10^{4}	1.2181×10^{4}
	Std	2.0648×10^{2}	1.8481×10^{2}	2.3388×10^{2}	2.2325×10^{2}
F10	Ave	6.3143×10^{3}	6.0722×10^{3}	6.3915×10^{3}	6.6655×10^{3}
	Std	7.3498×10^{2}	7.8776×10^{2}	8.9703×10^{2}	5.8111×10^{2}
F11	Ave	2.1975×10^{2}	2.7219×10^{2}	2.7511×10^{2}	2.2034×10^{2}
	Std	6.5423×10^{1}	7.6254×10^{1}	7.2566×10^{1}	6.8842×10^{1}
F12	Ave	1.3183×10^{6}	4.4202×10^{5}	6.5461×10^{5}	1.3696×10^{6}
	Std	1.3450×10^{6}	1.9140×10^{5}	5.1779×10^{5}	1.0082×10^{6}
F13	Ave	1.1952×10^{4}	1.4367×10^{4}	1.5342×10^{4}	1.5809×10^{4}
	Std	8.3953×10^{3}	1.2768×10^{4}	5.3362×10^{3}	1.1661×10^{4}
F14	Ave	9.5108×10^{5}	1.3536×10^{4}	4.5932×10^{4}	3.5058×10^{4}
	Std	2.9250×10^{6}	4.1747×10^{4}	3.9052×10^{4}	2.6873×10^{4}
F15	Ave	7.6744×10^{7}	1.8585×10^{4}	1.4823×10^{4}	1.7460×10^{4}
	Std	2.4264×10^{8}	1.9094×10^{8}	6.8056×10^{3}	4.3429×10^{3}
F16	Ave	1.8765×10^{3}	1.4308×10^{3}	1.5485×10^{3}	1.4742×10^{3}
	Std	6.7752×10^{2}	3.6547×10^{2}	4.3889×10^{2}	4.3405×10^{2}
F17	Ave	1.1699×10^{3}	1.2678×10^{3}	1.4765×10^{3}	1.2146×10^{3}
	Std	2.4208×10^{2}	3.4140×10^{2}	2.4021×10^{2}	3.2957×10^{2}
F18	Ave	3.2004×10^{5}	2.0647×10^{5}	3.6529×10^{5}	3.0214×10^{5}
	Std	1.1165×10^{5}	6.3321×10^{4}	1.6173×10^{5}	2.1860×10^{5}
F19	Ave	2.2718×10^{4}	1.9823×10^{4}	2.8837×10^{4}	3.3508×10^{4}
	Std	1.6016×10^{4}	1.2269×10^{4}	1.3445×10^{4}	1.5896×10^{4}
F20	Ave	1.5432×10^{3}	1.5321×10^{3}	1.6541×10^{3}	1.6224×10^{3}
	Std	2.0297×10^{2}	2.4922×10^{2}	1.4526×10^{2}	1.3707×10^{2}
F21	Ave	5.1570×10^{2}	5.3064×10^{2}	5.3393×10^{2}	4.7145×10^{2}
	Std	7.5917×10^{1}	5.5542×10^{1}	6.6308×10^{1}	7.0605×10^{1}
F22	Ave	9.6272×10^{3}	8.5476×10^{3}	9.0288×10^{3}	9.1518×10^{3}
	Std	1.0655×10^{3}	1.3920×10^{3}	7.6082×10^{2}	5.9770×10^{2}
F23	Ave	7.3148×10^{2}	7.5163×10^{2}	7.2357×10^{2}	6.8910×10^{2}
	Std	5.4037×10^{1}	7.9353×10^{1}	6.4455×10^{1}	5.9416×10^{1}
F24	Ave	8.7368×10^{2}	1.0356×10^{3}	8.3711×10^{2}	7.6560×10^{2}
	Std	3.2088×10^{2}	4.9646×10^{2}	1.1680×10^{2}	2.3482×10^{2}
F25	Ave	6.0653×10^{2}	5.9346×10^{2}	5.6456×10^{2}	6.2270×10^{2}
	Std	5.2638×10^{1}	4.6313×10^{1}	3.9344×10^{1}	2.3102×10^{1}
F26	Ave	1.2026×10^{3}	1.0517×10^{3}	3.9052×10^{3}	2.1641×10^{3}
	Std	1.9864×10^{3}	2.3762×10^{3}	3.8525×10^{3}	3.0304×10^{3}
F27	Ave	5.0001×10^{2}	5.0001×10^{2}	8.3083×10^{2}	5.0001×10^{2}
	Std	8.4022×10^{-5}	4.8001×10^{1}	1.1851×10^{2}	8.9994×10^{-5}
F28	Ave	5.0346×10^{2}	5.0001×10^{2}	5.1282×10^{2}	5.0001×10^{2}
	Std	1.0895×10^{1}	9.6943×10^{-5}	1.4948×10^{1}	7.4658×10^{-5}
F29	Ave	1.4587×10^{3}	1.4156×10^{3}	1.3953×10^{3}	1.2019×10^{3}
	Std	4.1734×10^{2}	3.5416×10^{2}	4.1788×10^{2}	3.8352×10^{2}
F30	Ave	1.2983×10^{6}	1.5119×10^{6}	1.3779×10^{6}	1.6880×10^{6}
	Std	1.6910×10^{5}	7.1042×10^{5}	3.5065×10^{5}	7.6298×10^{5}
+/−/≈		19/9/1	-	21/8/0	19/8/2
Mean Rank		2.69	1.91	2.86	2.53
Total Rank		3	1	4	2

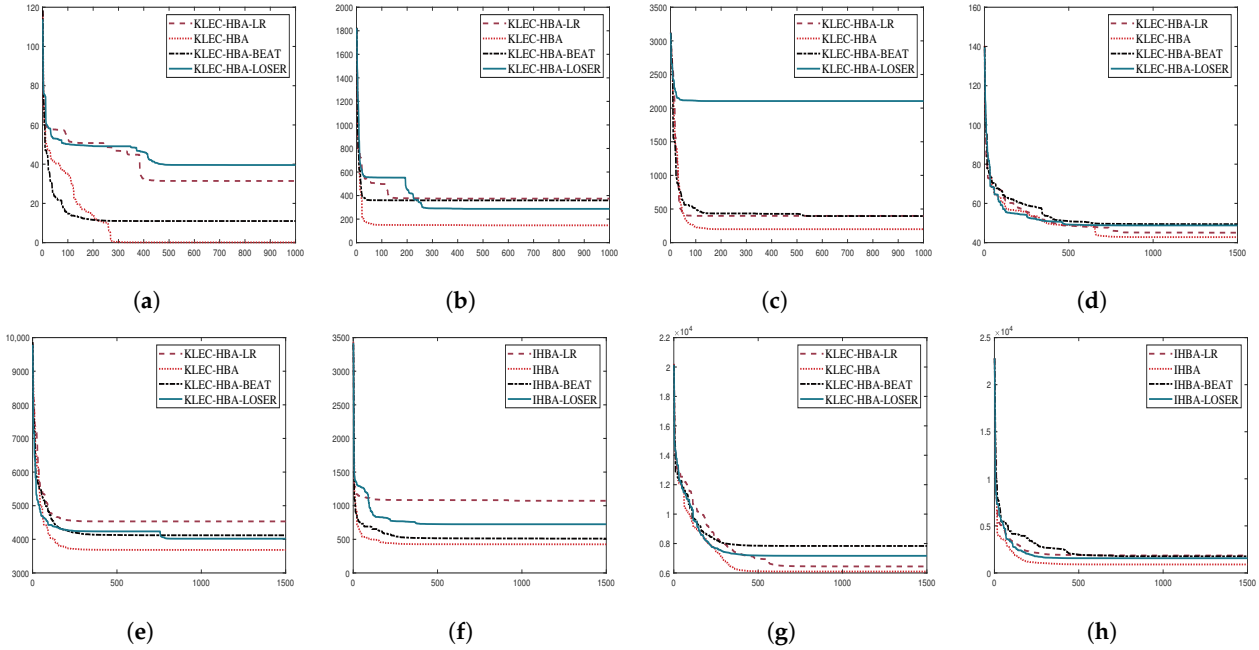


Figure 5. Convergence curves of the algorithm on ablation experiments. (**a**) Fun6-10dim. (**b**) Fun16-10dim. (**c**) Fun26-10dim. (**d**) Fun6-30dim. (**e**) Fun10-30dim. (**f**) Fun20-30dim. (**g**) Fun10-50dim. (**h**) Fun16-50dim.

5. Engineering Application

The objective of welded beam design [36] is to minimize its cost $f(x)$ while satisfying several constraints. These constraints include seven inequality constraints, e.g., shear stresses (τ), bending stresses of the beam (σ), buckling loads of the bars (P_c), and deflection at the end of the beams (δ), and four design variables, namely, $h(x_1)$, $l(x_2)$, $t(x_3)$ and $b(x_4)$, as illustrated in Figure 6.

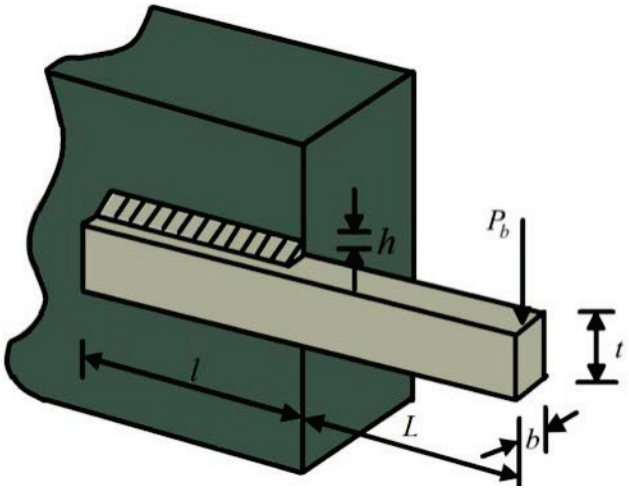


Figure 6. Schematic diagram of welded beam structure.

The specific mathematical model is shown below:
Objective function:

$$minf(x) = 1.1047x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2) \quad (9)$$

Constraints:

$$\begin{aligned} g_1(x) &= \tau(x) - \tau_{\max} \le 0, \\ g_2(x) &= \sigma(x) - \sigma_{\max} \le 0, \\ g_3(x) &= x_1 - x_4 \le 0, \\ g_4(x) &= 0.1047x_1^2 + 0.04811x_3x_4(14 + x_2) - 5 \le 0, \\ g_5(x) &= 0.125 - x_1 \le 0, \\ g_6(x) &= \delta(x) - \delta_{\max} \le 0, \\ g_7(x) &= P - P_c(x) \le 0. \end{aligned} \tag{10}$$

Relevant parameters and boundary constraints:

$$\tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2},$$

$$\tau' = \frac{P}{\sqrt{2x_1x_2}}, \quad \tau'' = \frac{MR}{J},$$

$$M = P\left(L + \frac{x_2}{2}\right), \quad R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2},$$

$$J = 2[\sqrt{2}x_1x_2\{\frac{x_2^2}{12} + (\frac{x_1 + x_3}{2})^2\}],$$

$$\sigma(x) = \frac{6PL}{x_3^2x_4}, \quad \delta(x) = \frac{4PL}{Ex_3^3x_4},$$

$$P_c(x) = \frac{4.013E}{L^2}\sqrt{\frac{x_3^2x_4^6}{36}}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right),$$

$$P = 6000 \text{ lb}, \quad L = 14 \text{ in}, \quad E = 30 \times 10^6 \text{ psi},$$

$$G = 12 \times 10^6 \text{ psi}, \quad \tau_{\max} = 13600 \text{ psi},$$

$$\sigma_{\max} = 30000 \text{ psi}, \quad \delta_{\max} = 0.25 \text{ in},$$

$$0.1 \le x_1 \le 2, \quad 0.1 \le x_2 \le 10,$$

$$0.1 \le x_3 \le 10, \quad 0.1 \le x_4 \le 2.$$

For the welded beam design problem, the maximum number of iterations and population size were set to 100 and 30 respectively, and the experiment was run independently 10 times. The optimization results for the welded beam design problem are presented in Table 8.

Table 8. Experimental results of algorithms for the welded beam design problem.

Algorithm	h	l	t	b	Mean f(x)
KLEC-HBA	2.1357×10^{-1}	1.0000×10^{1}	9.0368	2.0573×10^{-1}	4.7417×10^{3}
HHO	2.3539×10^{-1}	9.9587	9.3015	2.0444×10^{-1}	7.9402×10^{3}
GWO	2.1473×10^{-1}	1.0000×10^{1}	9.0469	2.0581×10^{-1}	4.7883×10^{3}
BKA	2.1387×10^{-1}	1.0000×10^{1}	9.0315	2.0618×10^{-1}	4.9193×10^{3}
HBA	2.1354×10^{-1}	1.0000×10^{1}	9.0321	2.0594×10^{-1}	4.9465×10^{3}
DBO	2.1354×10^{-1}	1.0000×10^{1}	9.0313	2.0597×10^{-1}	5.0693×10^{3}

For this problem, the optimal function value of the KLEC-HBA is the best compared to the optimal function values of the other algorithms. Figure 7 shows the convergence curves of the KLEC-HBA and the five other comparative algorithms, including HBA [27], HHO [34], GWO [37], DBO [38], and DKA [39], in finding the optimum. Figure 8 provides

the boxplot comparison of the experimental results obtained by the six algorithms over multiple independent runs. The boxplot is used to illustrate the distribution, median, and dispersion of the obtained objective function values, thereby reflecting the robustness and stability of each algorithm. Based on the graphs of the experimental results shown in Figures 7 and 8, it can be seen that the KLEC-HBA is better than the other comparison algorithms at finding the optimal results for this problem.

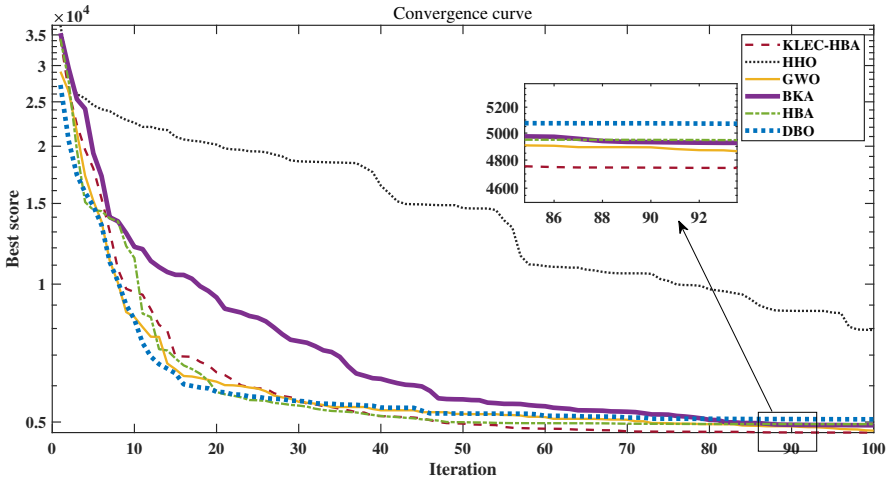


Figure 7. Schematic convergence curve of the algorithm.

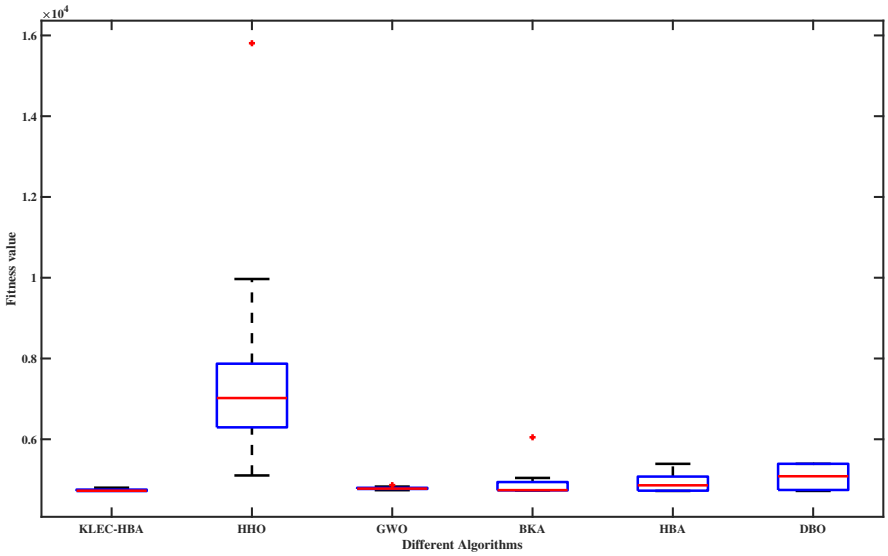


Figure 8. Algorithms box line comparison chart.

6. Conclusions and Future Directions

This paper proposes a honey badger optimization algorithm based on an evolutionary knowledge-learning strategy. The algorithm introduces adaptive learning parameters to enhance its capacity to learn historical experiences based on the original evolutionary knowledge-learning strategy. Additionally, this paper proposes an adversarial learning strategy. The strategy is based on failure experience, to increase the accumulation of the algorithm's successful historical experience and increase the diversity of the population. This is done to prevent the algorithm from falling into a local optimum. An adaptive mutation mechanism is used to mutate the globally optimal individuals. The efficacy of the proposed algorithm is validated through a comparative analysis and ablation experiments conducted on the CEC2017 test suite. Subsequently, to assess the performance of the KLEC-HBA in real-world scenarios, this study applies it to the welded beam engineering

problem, and the experimental outcomes demonstrate the effectiveness of the KLEC-HBA in addressing this challenge.

In the future, alternative strategies may be proposed or introduced to enhance the search performance of the intelligent optimization algorithm. Furthermore, the potential for extending the application of the algorithm to other fields, including feature selection, optimization of model structure and parameters, path planning, image segmentation, and equipment fault diagnosis, should be considered. Such an expansion will facilitate further exploration of the algorithm's scope of application, promote its effective utilization in a range of practical problems, and provide a broader foundation for the advancement of intelligent optimization algorithms.

Author Contributions: Methodology, R.Q. and Q.F.; Software, R.Q.; Writing—original draft, R.Q.; Writing—review & editing, H.S. and Q.F.; Supervision, H.S. and Q.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research study was supported by the National Natural Science Foundation of China (No. 12471482).

Data Availability Statement: Data will be made available upon reasonable request.

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

References

1. Amari, S. Backpropagation and stochastic gradient descent method. *Neurocomputing* **1993**, *5*, 185–196. [CrossRef]
2. Li, S.; Zhao, X. Image-based concrete crack detection using convolutional neural network and exhaustive search technique. *Adv. Civ. Eng.* **2019**, *2019*, 6520620. [CrossRef]
3. Yao, X. A new simulated annealing algorithm. *Int. J. Comput. Math.* **1995**, *56*, 161–168. [CrossRef]
4. Chakraborty, A.; Kar, A.K. Swarm intelligence: A review of algorithms. In *Nature-Inspired Computing and Optimization: Theory and Applications*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 475–494.
5. Eiben, A.E.; Smith, J.E. What is an evolutionary algorithm? In *Introduction to Evolutionary Computing*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 25–48.
6. Yang, F.; Wang, P.; Zhang, Y.; Zheng, L.; Lu, J. Survey of swarm intelligence optimization algorithms. In *Proceedings of the 2017 IEEE International Conference on Unmanned Systems (ICUS), Beijing, China, 27–29 October 2017*; IEEE: New York, NY, USA, 2017; pp. 544–549.
7. 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]
8. Cavallaro, C.; Cutello, V.; Pavone, M.; Zito, F. Machine Learning and Genetic Algorithms: A case study on image reconstruction. *Knowl.-Based Syst.* **2024**, *284*, 111194. [CrossRef]
9. Mallipeddi, R.; Suganthan, P.N.; Pan, Q.K.; Tasgetiren, M.F. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Appl. Soft Comput.* **2011**, *11*, 1679–1696. [CrossRef]
10. Liang, W.; Lou, M.; Chen, Z.; Qin, H.; Zhang, C.; Cui, C.; Wang, Y. An enhanced ant colony optimization algorithm for global path planning of deep-sea mining vehicles. *Ocean. Eng.* **2024**, *301*, 117415. [CrossRef]
11. Wang, D.; Tan, D.; Liu, L. Particle swarm optimization algorithm: An overview. *Soft Comput.* **2018**, *22*, 387–408. [CrossRef]
12. Zhou, X.; Ma, H.; Gu, J.; Chen, H.; Deng, W. Parameter adaptation-based ant colony optimization with dynamic hybrid mechanism. *Eng. Appl. Artif. Intell.* **2022**, *114*, 105139. [CrossRef]
13. Wu, L.; Huang, X.; Cui, J.; Liu, C.; Xiao, W. Modified adaptive ant colony optimization algorithm and its application for solving path planning of mobile robot. *Expert Syst. Appl.* **2023**, *215*, 119410. [CrossRef]
14. Hu, G.; Du, B.; Wang, X.; Wei, G. An enhanced black widow optimization algorithm for feature selection. *Knowl.-Based Syst.* **2022**, *235*, 107638. [CrossRef]
15. Zhu, F.; Li, G.; Tang, H.; Li, Y.; Lv, X.; Wang, X. Dung beetle optimization algorithm based on quantum computing and multi-strategy fusion for solving engineering problems. *Expert Syst. Appl.* **2024**, *236*, 121219. [CrossRef]
16. Ekinci, S.; Izci, D.; Abualigah, L.; Zitar, R.A. A modified oppositional chaotic local search strategy based aquila optimizer to design an effective controller for vehicle cruise control system. *J. Bionic Eng.* **2023**, *20*, 1828–1851. [CrossRef]
17. Liu, M.; Zhang, Y.; Guo, J.; Chen, J.; Liu, Z. An adaptive lion swarm optimization algorithm incorporating tent chaotic search and information entropy. *Int. J. Comput. Intell. Syst.* **2023**, *16*, 39. [CrossRef]

18. Sarhani, M.; Voß, S.; Jovanovic, R. Initialization of metaheuristics: Comprehensive review, critical analysis, and research directions. *Int. Trans. Oper. Res.* **2023**, *30*, 3361–3397. [CrossRef]
19. Li, Q.; Bai, Y.; Gao, W. Improved initialization method for metaheuristic algorithms: A novel search space view. *IEEE Access* **2021**, *9*, 121366–121384. [CrossRef]
20. Agushaka, J.O.; Ezugwu, A.E.; Abualigah, L.; Alharbi, S.K.; Khalifa, H.A.E.W. Efficient initialization methods for population-based metaheuristic algorithms: A comparative study. *Arch. Comput. Methods Eng.* **2023**, *30*, 1727–1787. [CrossRef]
21. Hu, G.; Zhong, J.; Du, B.; Wei, G. An enhanced hybrid arithmetic optimization algorithm for engineering applications. *Comput. Methods Appl. Mech. Eng.* **2022**, *394*, 114901. [CrossRef]
22. Shams, M.Y.; El-Kenawy, E.S.M.; Ibrahim, A.; Elshewey, A.M. A hybrid dipper throated optimization algorithm and particle swarm optimization (DTPSO) model for hepatocellular carcinoma (HCC) prediction. *Biomed. Signal Process. Control* **2023**, *85*, 104908. [CrossRef]
23. Mahajan, S.; Abualigah, L.; Pandit, A.K.; Altalhi, M. Hybrid Aquila optimizer with arithmetic optimization algorithm for global optimization tasks. *Soft Comput.* **2022**, *26*, 4863–4881. [CrossRef]
24. Al Thobiani, F.; Khatir, S.; Benaissa, B.; Ghandourah, E.; Mirjalili, S.; Wahab, M.A. A hybrid PSO and Grey Wolf Optimization algorithm for static and dynamic crack identification. *Theor. Appl. Fract. Mech.* **2022**, *118*, 103213. [CrossRef]
25. Seyyedabbasi, A. WOASCALF: A new hybrid whale optimization algorithm based on sine cosine algorithm and levy flight to solve global optimization problems. *Adv. Eng. Softw.* **2022**, *173*, 103272. [CrossRef]
26. Jiang, Y.; Zhan, Z.H.; Tan, K.C.; Zhang, J. Knowledge learning for evolutionary computation. *IEEE Trans. Evol. Comput.* **2023**, *29*, 16–30. [CrossRef]
27. Hashim, F.A.; Houssein, E.H.; Hussain, K.; Mabrouk, M.S.; Al-Atabany, W. Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems. *Math. Comput. Simul.* **2022**, *192*, 84–110. [CrossRef]
28. Fogel, D.B.; Atmar, J.W. Comparing genetic operators with Gaussian mutations in simulated evolutionary processes using linear systems. *Biol. Cybern.* **1990**, *63*, 111–114. [CrossRef]
29. Ali, M.; Pant, M. Improving the performance of differential evolution algorithm using Cauchy mutation. *Soft Comput.* **2011**, *15*, 991–1007. [CrossRef]
30. 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.
31. Karaboga, D.; Gorkemli, B.; Ozturk, C.; Karaboga, N. A comprehensive survey: Artificial bee colony (ABC) algorithm and applications. *Artif. Intell. Rev.* **2014**, *42*, 21–57. [CrossRef]
32. Xue, J.; Shen, B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Syst. Sci. Control Eng.* **2020**, *8*, 22–34. [CrossRef]
33. Gharehchopogh, F.S.; Gholizadeh, H. A comprehensive survey: Whale Optimization Algorithm and its applications. *Swarm Evol. Comput.* **2019**, *48*, 1–24. [CrossRef]
34. Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Future Gener. Comput. Syst.* **2019**, *97*, 849–872. [CrossRef]
35. Ye, Z.; Zhao, T.; Liu, C.; Zhang, D.; Bai, W. An improved honey badger algorithm through fusing multi-strategies. *Comput. Mater. Contin.* **2023**, *76*, 1479. [CrossRef]
36. Deb, K. Optimal design of a welded beam via genetic algorithms. *AIAA J.* **1991**, *29*, 2013–2015. [CrossRef]
37. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
38. Xue, J.; Shen, B. Dung beetle optimizer: A new meta-heuristic algorithm for global optimization. *J. Supercomput.* **2022**, *79*, 7305–7336. [CrossRef]
39. 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]

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.