Journal of Research in Science and Engineering (JRSE)
ISSN: 1656-1996 Volume-7, Issue-2, February 2025
DOI: 10.53469/jrse.2025.07(02).06

Zebra Optimization Algorithm Incorporating Multiple Improvement Strategies

Pingqi Cao[1], Tao Jiang[2], Fang Quan[3]

[1,2,3]Beijing Xinghang Mechanical-Electric Equipment Co, Ltd, Beijing 100074, China
[1]caopingqi@163.com, [2]jiangtao8603463@126.com, [3]13051386828@163.com

Abstract: ***Given the limitations of the Zebra Optimization Algorithm in terms of both the ability to jump out the local optimum solution and convergence speed, this study developed a zebra optimization algorithm incorporating multiple improvement strategies (MI-ZOA). In order to enhance the global search capability and improve the uniformity of the population distribution within the search space, the algorithm initially introduces Kent chaotic mapping to produce random sequences for population initialization. Moreover, the algorithm capitalizes on the long-tailed attribute of the Lévy flight and puts in a factor that has a non - linear variation with the iteration number, with the aim of increasing the search space coverage while being in coordination with the algorithm's local development capacity. Furthermore, the golden-sine update mechanism is introduced into the algorithm to improve search efficiency and optimization accuracy at a later stage. Subsequently, after the ZOA algorithm resists predator attacks, a Gaussian-Cauchy mutation operator is introduced to effectively avoid getting trapped in local optima and accelerate the algorithm's convergence rate. Finally, using eight benchmark functions in the CEC2017 test set, comparative tests were conducted on MI-ZOA, ZOA, DBO, GA, and HHO. The results showed that the MI-ZOA had advantages in convergence speed and global search ability compared to other algorithms.***

Keywords: Zebra Optimization Algorithm, Kent Chaos Mapping, Dynamic Lévy Flight, Golden Sine Algorithm, Gaussian-Cauchy Mutation**.**

1. Introduction

In the past few years, swarm intelligence algorithms have experienced rapid development, among which swarm intelligence optimization algorithms relying on the behavior of groups of natural organisms, with their advantages of great flexibility, independence from gradient mechanism, and excellent local development capability, have provided solutions for complex optimization problems such as system identification, image processing, and signal processing [1]. With the continuous deepening of algorithm research, numerous new swarm intelligence optimization algorithms have emerged. Recently, various swarm intelligence algorithms have been proposed, such as the Kepler Optimization Algorithm [2], Nutcracker optimizer algorithm [3], Greater Cane Rat Algorithm [4], and Black-winged Kite Algorithm [5].

Specifically, the Zebra Optimization Algorithm (ZOA) [6], a bio-inspired meta-heuristic algorithm that came out in 2022, has the traits of low parameter control requirements, simple architecture, and relatively good reconfigurability compared to other intelligent optimization algorithms. Additionally, ZOA is also recognized for its ability to adapt to complex and high-dimensional problems.

Its robustness is demonstrated by its insensitivity to control system parameters, and the ease with which it can be implemented increases its appeal as an effective tool for addressing these matters [7]. However, during the foraging phase, the behavior of population members approaching the pioneer zebra can easily lead to premature convergence of the population, thereby reducing their ability to explore the solution space. In addition, during the defense phase, the defense strategy cannot effectively help the population entirely, as it risks converging into local optima during practical optimization searches [8].

In response to the limitations, this study proposes an improved zebra optimization algorithm incorporating multiple improvement strategies (MI-ZOA). To achieve the goal of improving population diversity, the algorithm first introduces the Kent chaotic mapping to initialize the experimental sample population. To enhance the global search capability while balancing with the local exploitation capability of the algorithm, a Levy flight strategy with dynamic parameters is introduced during the foraging phase of the zebra algorithm. To further enhance the local exploitation capability, the core idea of the golden sine algorithm is integrated into the defense phase. Finally, after the defense phase, a Gaussian-Cauchy mutation operator is introduced for perturbation, achieving a balance between exploration and diversity and improving the global optimization accuracy. Through eight benchmark functions in the CEC2017 test set, comparative tests were conducted on MI-ZOA, ZOA, DBO, GA, and HHO to verify the effectiveness and progressiveness of MI-ZOA.

2. Zebra Optimization Algorithm

The Zebra Optimization Algorithm is a bio-inspired meta-heuristic algorithm fundamentally inspired by the herd behaviors of wild zebra populations in the African Savannah. As a kind of gregarious herbivore, zebras have two behaviors in nature that are important for the study of optimization algorithms: foraging and defending against predators. The former is manifested in the process in which the population leader leads the collective to search for the necessary water sources and fresh plants for survival. The latter is manifested in the process in which the population relies on different strategies to ensure the survival of the whole population when attacked by different kinds of predators.

2.1 Foraging Phase

The Zebra Optimization Algorithm updates the positions of population members by simulating the foraging behavior of zebra populations, where the best performing individual is considered as the population leader, leading other members to approach its positions. The process of updating the positions of the population individuals mentioned above can be represented by formula (1) and formula (2).

$$x_{i,j}^{new} = x_{i,j} + r \cdot (PZ_j - I \cdot x_{i,j}) \tag{1}$$

$$x_i = \begin{cases} x_i^{new}, F_i^{new} < F_i \\ x_i, else \end{cases} \tag{2}$$

$x_{i,j}^{new}$ is the updated value of the *j*th dimension value of the *i*th population member in the foraging phase, *F* represents the fitness value, *PZ* represents the optimal member in the zebra population, and r is a stochastic number in the open interval between 0 and 1. *I* is a random value in the interval (1, 2) and reflects the degree of change in the population.

2.2 Defense Phase

During the defense phase, zebra populations adopt different defense strategies against different predators. When facing large predators such as lions, zebras tend to adopt an escape strategy. When faced with attacks from small predators such as hyenas and dogs, zebras choose to confuse and scare predators by gathering populations. The process of renewing the positions of the group individuals mentioned above can be represented by formula (3) and formula (4).

$$x_{i,j}^{new} = \begin{cases} x_{i,j} + R \cdot (2r-1) \cdot (1 - \frac{t}{T}) \cdot x_{i,j}, P_s < 0.5 \\ x_{i,j} + r \cdot (AZ_j - I \cdot x_{i,j}), else \end{cases} \tag{3}$$

$$x_i = \begin{cases} x_i^{new}, F_i^{new} < F_i \\ x_i, else \end{cases} \tag{4}$$

$x_{i,j}^{new}$ is the updated value of the *j*th dimension value of the *i*th population member in the defending phase, F_i^{new} is the fitness value of the ith member's updated position, *R* is a constant equal to 0.01, *t represents the present iteration number*, *T* is the maximum iteration number, and *AZ* represents the zebra attacked during the defense phase.

3. Zebra Optimization Algorithm Incorporating Multiple Improvement Strategies

3.1 Kent Chaotic Mapping

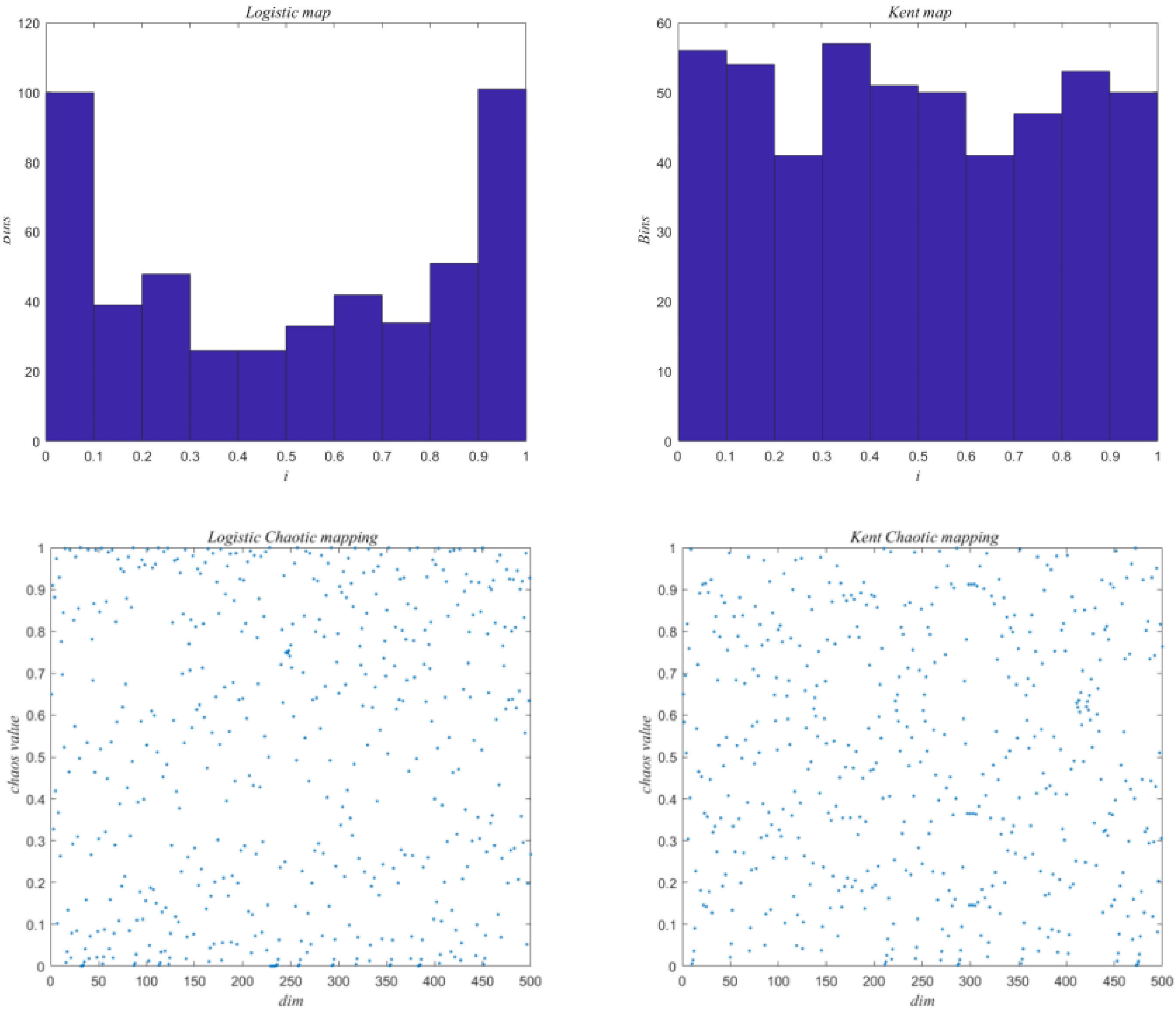

Figure 1: Scatter plot and histogram of logistic mapping and Kent mapping

Introducing chaotic mapping into algorithms is an effective method that can enhance population diversity and improve the algorithm's global search capability. Chaotic mapping is characterized by randomness and ergodicity, which can greatly improve the diversity by generating an initial population with a uniform distribution in the initialization stage of the population. In swarm intelligence optimization algorithms, chaotic mapping can be introduced through the following steps:

a) Initialization: Randomly generate an initial population with each individual's position represented as x_i.

b) Apply chaotic mapping: Apply chaotic mapping to the position of each individual x_i to generate new chaotic positions y_i.
c) Mapping to search space: Mapping the chaotic position y_i to the search space range of the optimization problem.

d) Update population: Update the population with newly generated positions x_i to obtain a more diverse initial population.

Among numerous chaotic mappings, Kent chaotic mapping [9] is a simple and discrete chaotic system with the following mathematical expression:

$$x_{n+1} = \begin{cases} \frac{x_n}{\mu}, 0 \le x_n < \mu \\ \frac{1-x_n}{1-\mu}, \mu \le x_n < 1 \end{cases} \tag{5}$$

$x_n$represents the current state value, and the chaotic orbit state value is taken as the interval value of (0,1). μ is the control parameter of Kent mapping within the range of (0,1), and to avoid the system falling into periodic behavior, μ cannot be taken as 0.5.

The Kent mapping is isomorphic to the commonly used Logistic mapping, which has a simple form and complex dynamic properties. However, according to the scatter plots and histograms of the two different chaotic mappings in Figure 1, the logistic mapping exhibits the Chebyshev distribution with more at both ends and less in the middle in terms of probability density. In contrast, Kent mapping exhibits a more uniform probability density distribution. Therefore, this study chooses to introduce Kent chaotic mapping in the early stages of the algorithm, which is in line with the goal of enhancing population diversity and improving global search capability.

3.2 Dynamic Levy Flight Strategy

Levy flight is a typical random walk mechanism where the step size follows a heavy-tailed distribution [10]. The mathematical model of the probability density function for Levy's flight is described as follows:

$$L(x,\beta) = \frac{1}{\pi}\int_0^{\infty} cos(\tau \cdot x) \cdot e^{-\gamma \cdot \tau^{\beta}} d\tau \tag{6}$$

According to the probability density mathematical model, the larger the parameter β, the more uniform the random number values generated by the probability density function curve, which is beneficial for accelerating the speed of the global exploration process. Aiming to better comply with the nonlinear optimization rules in the algorithm iteration process, a nonlinear variation factor β_1 defined by the sigmoid function is added. The mathematical description of factor β_1 is shown in formula (7). In the formula (7), t is the current iteration number, T is the maximum iteration number, $\delta_1, \delta_2, \delta_3, \delta_4$ are constant coefficients.

$$\beta_1 = \delta_1 + \frac{\delta_2}{1+e^{\delta_3(\frac{t}{T}-\delta_4)}} \tag{7}$$

In the foraging phase of the original ZOA, the Levy flight strategy with a dynamic parameter is introduced to perturb the population's individual updating process. The parameter β_1 of Levy flight changes non-linearly as the number of iterations increases. Assigning a larger value to β_1 during the global exploration process in the early stages of the algorithm expands the search range and exploration discoveries, which is beneficial for enhancing population diversity and significantly reducing the risk of zebra individuals plunging into a local optimum. During the later stage of iteration, assigning a lower value to the parameter beta allows the population to explore local regions with smaller step sizes, effectively improving the quality of the algorithm's later solutions and enabling the algorithm to converge to the global optimum. After introducing the dynamic Levy flight strategy, the position renewing expression for the foraging phase is as follows:

$$x_{i,j}^{new} = x_{i,j} + r \cdot (PZ_j - I \cdot x_{i,j}) \cdot L(\tfrac{t}{T}, \beta_1) \tag{8}$$

3.3 Combining with the Golden Sine Algorithm

The Golden Sine Algorithm is a meta-heuristic algorithm proposed in 2017, which is influenced by the golden ratio and sine function [11]. The algorithm traverses every value of the sine function covering the entire unit circle, and introduces the golden ratio to narrow down the solution space. The golden ratio involved in the Golden Sine Algorithm is shown in Formula (9).

$$\begin{cases} c_1 = -\pi + \pi(1-\tau) \\ c_2 = -\pi(1-\tau) + \pi\tau \\ \tau = (\sqrt{5}-1)/2 \end{cases} \tag{9}$$

After using chaotic mapping for population initialization and introducing the Levy flight random walk to further improve the global exploration capability during the foraging phase, the problem that the algorithm plunges into a local optimum easily at a later stage needs further improvement.

Considering that the golden sine algorithm continuously reduces the current solution space and can considerably improve the algorithm's exploration performance and precision at a later stage, this study chooses to combine the idea of the Golden Sine Algorithm with the defending against predators phase. After integrating the golden sine perturbation strategy, the position update formula for the defense phase is as follows:

$$x_{i,j}^{new} = \begin{cases} S1: x_{i,j}(|sin\, r_1|) - r_2\, sin\, r \left|c_1 P_z - c_2 x_{i,j}\right|, P_s < 0.5 \\ S2: x_{i,j}(|sin\, r_1|) - r_2\, sin\, r \left|c_1 A Z_j - c_2 I x_{i,j}\right|, else \end{cases} \tag{10}$$

3.4 Gaussian-Cauchy Mutation Perturbation Strategy

As the zebra optimization algorithm goes through its later iterations and starts to converge to a current optimum that isn't the global optimum, it may result in the algorithm's premature convergence.

With the aim of overcoming or even avoiding the predicament of the algorithm's premature convergence as much as possible and further enhancing the performance of the ZOA, the MI-ZOA proposed in this paper applies a mutation perturbation strategy in the later stage of the algorithm for improvement. This study chooses the perturbation method of Gaussian-Cauchy mutation, and the mutation operator can be expressed as:

$$rr = \frac{t^2}{T^2} \cdot Gauss(0,1) + (1 - \frac{t^2}{T^2}) \cdot Cauchy(0,1) \quad (11)$$

Gauss (0,1) is the standard Gaussian mutation operator. Cauchy (0,1) is the standard Cauchy mutation operator. t is the present iteration number, and T is the upper limit of iterations.

On one hand, at lower iteration counts, the large step size of the Cauchy mutation, which is advantageous for global search capability, plays a dominant role. On the other hand, as iterations increase, the smaller step sizes of Gaussian mutation come into play, boosting the algorithm's local search capacity and enabling it to break free from local optima. By using the fusion Gaussian-Cauchy mutation operator, exploration and diversityare given due attention, thus boosting the overall exploration precision of the algorithm.

In this paper, we use the defined Gaussian-Cauchy mutation operator to perturb the optimal zebra individual in each iteration, and the perturbation process is shown in formula (12)

$$X^t_{perturbed\_best} = X^t_{best} \cdot (1 + rr) \quad (12)$$

3.5 The Execution Process of MI-ZOA

The process of the zebra optimization algorithm incorporating multiple improvement strategies is as follows:

Step 1: Initialize the population parameters using Kent chaotic mapping.

Step 2: During the foraging phase, evaluate the fitness of each zebra in the population, select pioneer zebra and followers, and update their positions using the formula based on the dynamic Levy flight strategy.

Step 3: In the defense phase, using the Golden Sine Algorithm, update the position by emulating the zebras' defense methods against attacks from various predators.

Step 4: After the defense phase, use the Gaussian-Cauchy mutation operator to perturb the optimal solution of the current iteration.

Step 5: Compare the fitness value of the current iteration's optimal solution, the fitness value of the new solution perturbed by the Gaussian-Cauchy mutation operator, and the fitness values of previous iterations, and then renew historical optimal individual.

Step 6: Determine if the maximum number of iterations has been reached. If not, return to step 2.

Step 7: Output the optimal result and end the algorithm.

The algorithm flowchart of MI-ZOA is shown in Figure 2:

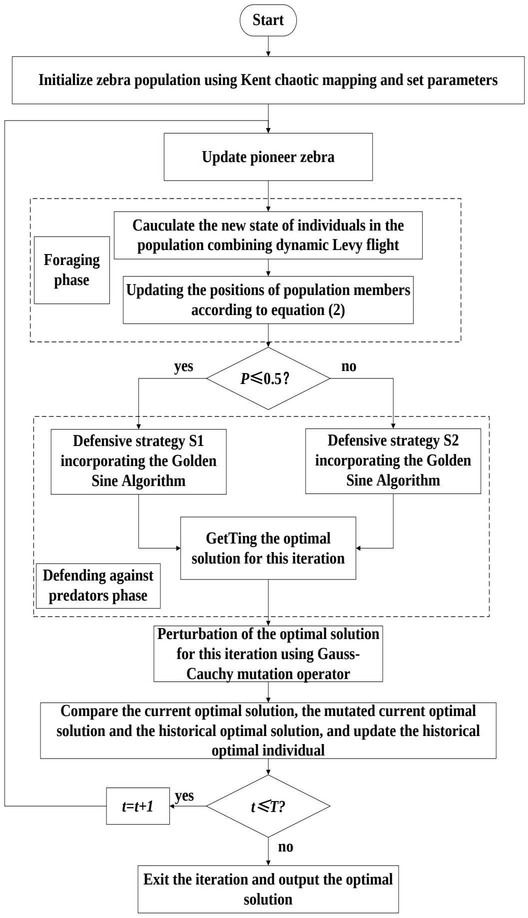

Figure 2: Algorithmic flow of MI-ZOA

4. Experiments, Results, and Analysis

4.1 Test Environment and Parameter Settings

The experimental procedures in this paper were all conducted in an AMD Ryzen 7 5800H CPU, 3.20GHz, Windows 10 64-bit testing environment, and simulations were implemented using Pycharm software.

Aiming to measure the comprehensive performance of MI-ZOA, in the present study, MI-ZOA, ZOA, and some other algorithms in recent years are tested and compared using the eight benchmark test functions in the CEC2017 test set [14]. As listed in Table 1, the mean fitness value and standard deviation of the fitness value using different test functions are recorded in order to test and analyze the algorithm's performance in several aspects.

Table 1: Test Functions

Test function	Function name	dim
Function 1	Shifted and Rotated Bent Cigar Function	30
Function 3	Shifted and Rotated Zakharov Function	30
Function 6	Shifted and Rotated Expanded Scaffer's F6 Function	30
Function 7	Shifted and Rotated Lunacek Bi_Rastrigin Function	30
Function 12	Hybrid Function2(N=3)	30
Function 13	Hybrid Function3(N=3)	30
Function 22	Composition Function2(N=3)	30
Function 23	Composition Function3(N=4)	30

aIn order to ensure the fairness of the algorithm testing and the accuracy of the experimental results, the population size of all algorithms is set to 30, the number of iterations is 800, the testing dimension is 30, and each function is tested for 50 times, and the mean and standard deviation of the experiments will be recorded.

4.2 Test Results and Analysis

On the test function, this study compared the mean and standard deviation values of the optimal fitness of MI-ZOA with ZOA, DBO [13], GA [14], and HHO [15]. The specific results are shown in Table 2.

In the functions used for testing, Function 1 and Function 3 are simple unimodal functions, Function 6 and Function 7 are simple multimodal functions, Function 12 and Function 13 are hybrid functions, and Function 22 and Function 23 are composition functions. The test functions span a wide variety of function types and have all been tested with 30 dimensions.

According to Table 2, MI-ZOA performs better than ZOA in unimodal, multimodal, hybrid, and composition functions, with lower mean and standard deviation fitness values, demonstrating the effectiveness of the improved method used by MI-ZOA. Compared with other algorithms such as DBO, the mean value and standard deviation of fitness value obtained by using MI-ZOA also have advantages, indicating that the MI-ZOA is still progressive.

In addition, by analyzing the convergence curves of MI-ZOA and other algorithms such as HHO presented in this article on different functions, the performance of the algorithms can be intuitively reflected. The test results are shown in Figure 3~10.

Table 2: Test results of algorithms on various test functions

Test function		MI-ZOA	ZOA	DBO	GA	HHO
Function 1	mean	6.3213e+08	4.0299e+09	7.2778e+10	3.3391e+10	4.5319e+11
	std	9.3047e+07	6.2000e+08	3.8665e+10	8.3040e+09	8.8611e+10
Function 3	mean	8.0380e+04	8.0446e+04	8.6271e+04	9.3835e+04	8.8171e+04
	std	6.3245e+03	5.9018e+03	1.4449e+04	2.0022e+04	2.8760e+03
Function 6	mean	6.7261e+02	6.8561e+02	6.8507e+02	7.9845e+02	7.9292e+02
	std	7.6727e+00	9.8462e+00	1.5504e+01	1.3437e+01	9.7799e+00
Function 7	mean	1.1088e+03	1.2783e+03	1.2164e+03	1.2624e+03	1.3100e+03
	std	2.8456e+01	5.8751e+01	8.4937e+01	4.4870e+01	5.2476e+01
Function 12	mean	7.6395e+06	5.5661e+08	2.6369e+09	1.6550e+09	6.4149e+10
	std	2.9696e+06	2.0159e+08	7.3844e+09	7.5857e+08	2.7022e+10
Function 13	mean	2.7629e+05	6.8586e+08	3.2889e+09	2.0068e+09	2.5324e+10
	std	2.7526e+05	3.4848e+08	1.0216e+10	1.5307e+09	3.0685e+10
Function 22	mean	3.5606e+03	7.7291e+03	5.1939e+03	5.3199e+03	9.1619e+03
	std	4.3235e+02	9.8776e+02	2.4124e+03	2.5609e+03	6.2649e+02
Function 23	mean	2.9357e+03	3.5288e+03	3.1018e+03	3.8845e+03	3.4873e+03
	std	2.2293e+01	1.5368e+02	9.8031e+01	2.5734e+01	1.5229e+02

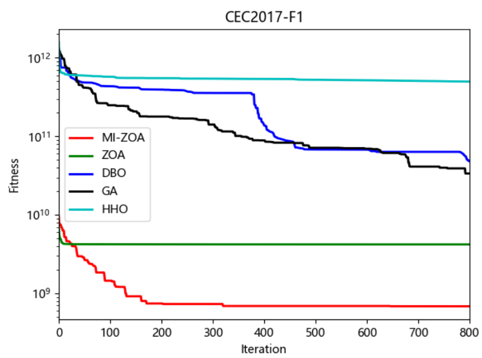

Figure 3: Convergence curve of test function Function 1

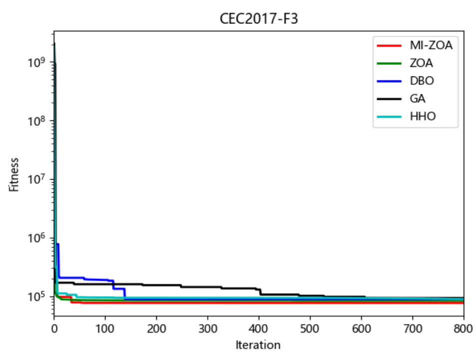

Figure 4: Convergence curve of test function 3

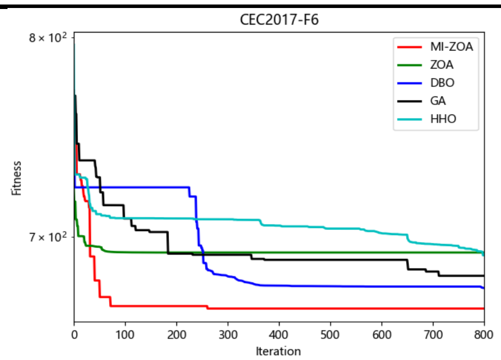

Figure 5: Convergence curve of test function 6

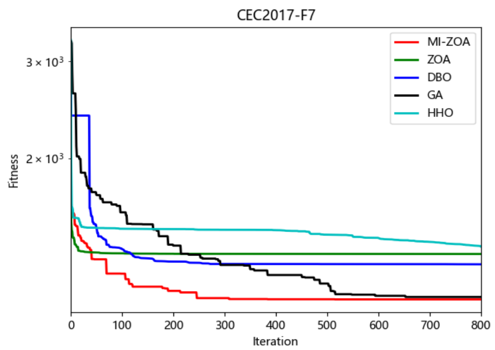

Figure 6: Convergence curve of test function 7

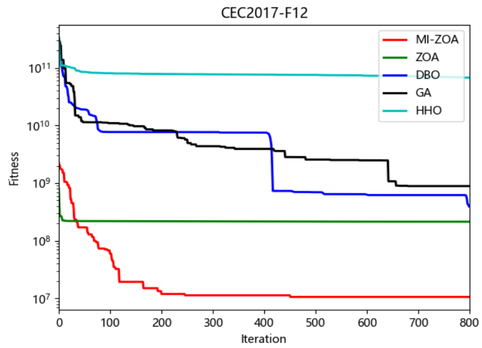

Figure 7: Convergence curve of test function 12

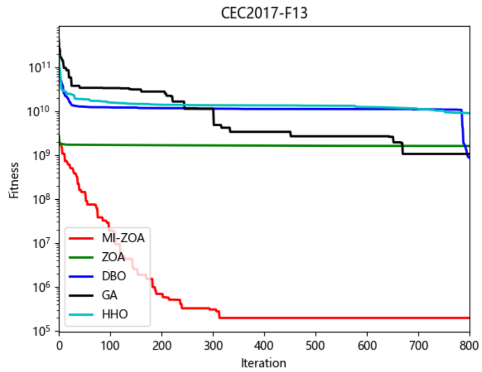

Figure 8: Convergence curve of test function 13

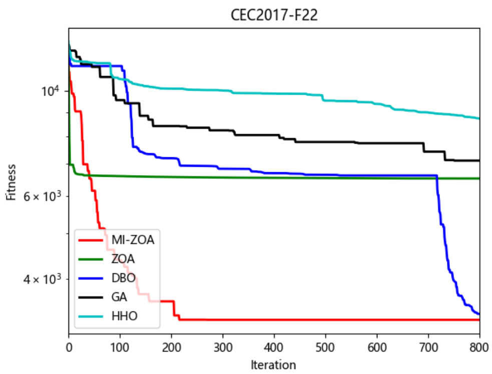

Figure 9: Convergence curve of test function 22

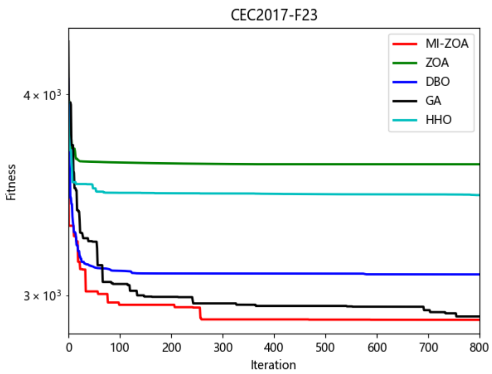

Figure 10: Convergence curve of test function 23

From Figure 3~10, it is evident that the MI-ZOA suggested here, when contrasted against the ZOA, is not characterized by a quick convergence speed. However, due to the introduction of strategies such as Levy flight and golden sine perturbation, it ultimately converges to a better globally optimal solution. Compared to other algorithms, the convergence curve of MI-ZOA rapidly decreases in the early stages of iteration, fluctuates less during the iteration process, and becomes more stable in the later stages of iteration.

In conclusion, MI-ZOA shows obvious advantages in stability of convergence process, global search ability, ability to get rid of local optimal traps and final convergence accuracy, which proves the progressiveness and superiority of this algorithm.

5. Conclusion

Given the problem of being trapped by the local optimum in the Zebra Optimization Algorithm, this article proposes an improved zebra optimization algorithm incorporating multiple improvement strategies. During population initialization, MI-ZOA introduces Kent chaotic mapping phase to enhance population diversity. During the foraging stage, the Levy flight random walk with dynamic parameters is adopted to enhance the global search capability while coordinating with algorithm's local exploration ability. In the defense phase, the core idea of the Golden Sine Algorithm is introduced by MI-ZOA to further enhance local search capabilities. After the defense phase, the Gaussian- Cauchy mutation operator is introduced to perturb the optimal solution of the current iteration, further enhancing the ability to escape from local optima.

Next, a series of tests were carried out on the improved algorithm. Through simulation experiments on eight benchmark test functions and comparison with other three intelligent optimization algorithms, the average value, standard deviation data and convergence curve were analyzed to ascertain the effectiveness and progressiveness of MI-ZOA. In the future, the plan focuses on further improving the performance of MI-ZOA and applying it to solve practical engineering problems.

References

[1] J. Tang, G. Liu, Q. Pan, A review on representative swarm intelligence algorithms for solving optimization problems: applications and trends, IEEE/CAA J. Autom. Sin. 8 (10) (2021) 1627–1643.

[2] Abdel-Basset, M. et al. Kepler optimization algorithm: A new metaheuristic algorithm inspired by Kepler's laws of planetary motion. Knowl. Based Syst. 268, 110454 (2023).

[3] "Abdel-Basset M, Mohamed R, Jameel M, et al. Nutcracker optimizer: A novel nature-inspired metaheuristic algorithm for global optimization and engineering design problems [J]. Knowledge-Based Systems, 2023, 262: 110248."

[4] Agushaka J O, Ezugwu A E, Saha A K, et al. Greater Cane Rat Algorithm (GCRA): A Nature-Inspired Metaheuristic for Optimization Problems [J]. Heliyon, 2024.

[5] Wang J, Wang W, Hu X, et al. Black-winged kite algorithm: a nature-inspired meta-heuristic for solving benchmark functions and engineering problems [J]. Artificial Intelligence Review, 2024, 57(4): 1-53.

[6] Trojovská E, Dehghani M, Trojovský P. Zebra optimization algorithm: A new bio-inspired optimization algorithm for solving optimization algorithm [J]. IEEE Access, 2022, 10: 49445-49473.

[7] I.A. Khan, et al., Load frequency control in power systems with high renewable energy penetration: a strategy employing PIλ (1 + PDF) controller, hybrid

energy storage, and IPFC-FACTS, Alex. Eng. J. 106 (2024) 337–366.

[8] S. Shang, J. Zhu, Q. Liu, Y.S. Shi, T.Z. Qiao, Low-altitude small target detection in sea clutter background based on improved CEEMDAN-IZOA-ELM, Heliyon 10 (4) (2024) e26500.

[9] Tavazoei M S, Haeri M. Comparison of different one-dimensional maps as chaotic search pattern in chaos optimization algorithms [J]. Applied Mathematics and Computation, 2007, 187(2): 1076-1085.

[10] Cui Y, Shi R, Dong J. CLTSA: A Novel Tunicate Swarm Algorithm Based on Chaotic-Lévy Flight Strategy for Solving Optimization Problems. Mathematics. 2022; 10(18):3405. https://doi.org/10.3390/math10183405

[11] ZHANG J, WANG S J. Improved whale optimization algorithm based on nonlinear adaptive weight and golden sine operator [J]. IEEE access, 2020(8): 877013-877048.

[12] Awad, N. H., Ali, M. Z., Liang, J. J., Qu, B. Y., & Suganthan, P. N. (2016). “Problem definitions and evaluation criteria for the CEC2017 special session and competition on single objective real-parameter numerical optimization,” Technical Report. Nanyang Technological University, Singapore.

[13] Zhu F, Li G, Tang H, et al. Dung beetle optimization algorithm based on quantum computing and multi-strategy fusion for solving engineering problems [J]. Expert Systems with Applications, 2023: 121219.

[14] Oszczypala, M., Ziolkowski, J., & Malachowski, J. (2024). Redundancy allocation problem in repairable k out-of-n systems with cold, warm, and hot standby: A genetic algorithm for availability optimization. Applied Soft Computing, 165, 112041.

[15] Wang, M.; Zhao, G.Y.; Wang, S.F. Hybrid random forest models optimized by Sparrow search algorithm (SSA) and Harris hawk optimization algorithm (HHO) for slope stability prediction. Transp. Geotech. 2024, 48, 13.