# Tuna Swarm Optimization Algorithm with Sine Selection Factor to Solve High-dimensional Feature Selection Problems

Yu-Cai Wang, Shi Li, Jie-Sheng Wang*, Zi-Rui Xu, Si-Yu Jin

*Abstract*—**When dealing with high-dimensional datasets, traditional feature selection (FS) methods often have limitations such as being prone to falling into local optima and having low computational efficiency. Swarm intelligence optimization algorithms, with their group collaboration and global search capabilities, provide new solutions to the FS problem. This paper proposes a tuna optimization algorithm with sinusoidal selection factors to efficiently screen the optimal feature subset. Aiming at the problem that the tuna algorithm cannot continuously explore and converge to a better solution. By introducing the sinusoidal oscillation mechanism, the selection of the two foraging behaviors can be dynamically adjusted, thereby enhancing the local exploitation and global exploration effects of the algorithm. Make the algorithm exploitation and exploration behaviors have dynamic adaptability and improve the convergence accuracy. In the IEEE CEC-2022 section: Firstly, verify the effectiveness of the sinusoidal selection factor. Select the parameter settings of the better sine selection factor. Then, the optimal algorithm SSFTSO was compared with other algorithms, and it was found that the SSFTSO performed better in terms of the average fitness value and convergence speed. In the feature selection section: To verify the ability of the algorithm, 10 public high-dimensional datasets were selected for experiments. The SSFTSO was compared with PDO, WOA, HHO, AOA, OOA and AO. The results showed that the SSFTSO had great advantages in terms of average fitness, average accuracy rate and average running time, and achieved efficient search for the optimal feature subset. It ranked first in terms of the average number of selected features after the Friedman test. This research provides an effective solution for feature selection of high-dimensional data.**

*Index Terms*—**feature selection, sinusoidal selection factor, tuna swarm optimization algorithm, high-dimensional data**

Manuscript received May 20, 2025; revised July 31, 2025. This work was supported by the Basic Scientific Research Project of Institution of Higher Learning of Liaoning Province (Grant No. LJ222410146054), and Postgraduate Education Reform Project of Liaoning Province (Grant No. LNYJG2022137).

Yu-Cai Wang is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: wyc@stu.ustl.edu.cn).

Shi Li is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: ls@stu.ustl.edu.cn).

Jie-Sheng Wang is a professor of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (Corresponding author, phone: 86-0412-2538246; fax: 86-0412-2538244; e-mail: wjs@ustl.edu.cn).

Zi-Rui Xu is a postgraduate student at School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, 114051, P. R. China (e-mail: xzr@stu.ustl.edu.cn).

Si-Yu Jin is a postgraduate student at School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan 114051, China (E-mail: jsy@stu.ustl.edu.cn).

## I. Introduction

With the trend of the big data era, high-dimensional data has become increasingly common in fields such as machine learning, data mining and pattern recognition [1]. However, not all features can provide valid information. In high-dimensional data, there are often a large number of redundant and irrelevant features, which not only increase the computational cost, but also may lead to the decline of over-fitting [2]. Feature selection (FS) can be used as an effective dimensional reduction technique. By picking out more representative feature subsets from the original data set, the purpose of dimensional reduction can be achieved [3]. Up to now, significant progress has been made in the research of the FS method. FS can mainly be divided into Filter, Wrapper and Embedded types [4]. Filter method assesses the importance of features through statistical metrics, is computationally efficient but ignores model feedback. Wrapper method relies on the performance of a specific model for feature screening, which has a good effect but a high computational cost. Embedded method automatically selects features through the model training process, taking into account both efficiency and performance.

Most scholars have conducted research on the Filter method. For example, Ref. [5] takes the DE algorithm as the carrier, selects the features with higher rankings among the two filtering formulas, and proposes a new filtering formula standard. In the single-objective and multi-objective frameworks, the superiority of the proposed method has been demonstrated by comparison with MI. Ref. [6] proposed a new MRelief method based on Relief. The effectiveness of the proposed method was proved through experiments conducted on the UCI data set and the gene expression benchmark data set. Ref. [7] considered two classifiers, KNN and SVM, and explored the advantages and disadvantages of 16 filter-based FS methods. It was found that there is no single method that can ensure the performance is always optimal, but some filtering methods have better coordination with classifiers. For the Wrapper method, Ref. [8] introduced Q-learning and comprehensive learning strategies and designed an enhanced algorithm (QCLFRIME). Through the comparison on high-dimensional datasets, it is found that QCLFRIME is an effective tool. Ref. [9] adopted the adaptive sequential forward selection method in SFSBW to generate the candidate optimal feature subset. This method was used as one of its components to study the problem of pedestrian trajectory prediction. It was found that the proposed method could provide better prediction effects. For the problem of plant leaf classification, Ref. [10] designed a binary cuckoo algorithm. It was found that the accuracy rates on the Swedish and Flavia datasets reached 95.67% and 99.6% respectively. For the Embedded method, Ref. [11] uses scaling factors to impose penalty measures on the data set and employs two support vector machine techniques to propose an embedded FS method for SVM classification. It was found that the average prediction performance of the proposed method was the best. For the traffic prediction problem, Ref. [12] designed an embedded FS method for improving the Grey Wolf Optimizer based on SVM. Compared with other models, the proposed method can obtain better prediction results.

Although the existing methods have their own good performances in different scenarios, they still face many challenges. For example, for the FS problem, data in different fields have unique distributions and characteristic correlations, and targeted algorithms need to be designed. For high-dimensional data, the feature space is more complex, and the applicability of traditional methods may be limited. Furthermore, on the premise of ensuring the accuracy of the model, how to achieve the efficiency and interpret-ability of FS is also an urgent problem to be solved. Therefore, proposing a new feature selection algorithm has important theoretical and application value.

Inspired by the predatory behavior of the Marine creature Tuna, the Tuna Swarm Optimization Algorithm was proposed [13]. With the advantages of having few parameters and a balance between development and exploration, TSO has been successfully applied in a variety of real-world problems. Aiming at the problem of autonomous path planning for underwater vehicles, Yan et al. designed the QLTSO algorithm based on reinforcement learning. It was found that path planning achieved a 100% success rate in various two-dimensional and three-dimensional complex environments [14]. Wang et al. introduced the chaotic strategy and the Levy flight strategy and designed the enhanced CLTSO. The BP neural network was optimized by the proposed method, and it was found that the accuracy rate was higher [15]. Sun et al. proposed the chaotic strategy and the offset distribution estimation strategy, and designed a new MSTSO algorithm. After verification through the UCI data set, it was found that the proposed method has achieved a significant improvement in accuracy [16]. For the job-shop scheduling, Fan et al. introduced the genetic chaos strategy and the Levi operator strategy, and designed the GCLNTSO algorithm. It was found that the proposed method had better performance [17]. For the parameter estimation problem of different photovoltaic models, Hassan et al. combined EO with TSO and designed the hybrid algorithm EOTSO. Through five cases, it was found that the proposed method was due to other techniques [18].

Taking TSO as the carrier, this paper conducts research on the problem of high-dimensional feature selection and proposes a tuna swarm optimization algorithm with sine selection factor (SSFTSO). And its validity is verified through theoretical analysis and experiments. The main contributions are as follows:

(1) The sinusoidal selection factor was proposed. By introducing the sinusoidal oscillation mechanism, the selection of the two foraging behaviors was dynamically adjusted, thereby enhancing the local exploitation and global search effects of the algorithm and improving the convergence accuracy.

(2) At IEEE CEC-2022, the effectiveness of the sinusoidal selection factor strategy was proved. By comparing SSFTSO with other algorithms, it was found that the SSFTSO had better performance.

(3) On 10 public high-dimensional datasets, the performance of SSFTSO was verified by comparison with other algorithms. The SSFTSO performs strongly in terms of average fitness value, average accuracy rate and average running time.

The overall structure of this article is arranged as follows. Section II introduces the implementation of the tuna optimization algorithm with sinusoidal selection factor. Section III is the experiments and analyses at IEEE CEC-2022. Section IV adopts the tuna optimization algorithm based on sinusoidal selection factors to solve the high-dimensional FS. Section V presents the conclusion.

## II. Tuna Swarm Optimization Algorithm With Sine Selection Factor (SSFTSO)

The detailed content of this section is as follows. Sub-section *A* introduces the basic tuna swarm optimization algorithm. Sub-section *B* designs the sine selection factor strategy and proposes an enhanced tuna swarm optimization algorithm with sine selection factor (SSFTSO).

### A. Basic TSO

TSO conducts group hunting behaviors through two methods, namely spiral foraging and parabolic foraging, respectively, and then searches for the optimal solution. Furthermore, the TSO algorithm has a mutation probability value z, and a new position can be regenerated through the judgment of z. $z = 0.05$ , The detailed modeling is as follows:

(a) Spiral Foraging Behavior. When tuna groups are engaged in predatory behavior, they sometimes encounter targets that are difficult to lock onto. A small portion of tuna will move in a certain direction, and the other tuna around will adjust their direction and swim along with that small portion of tuna. Eventually, they start hunting in a spiral shape. In addition, there will be information exchange between each tuna and the optimal individual. Information sharing among adjacent individuals is achieved through the form of following. This behavior prompts the algorithm to be biased towards local exploitation. However, constantly exchanging information with the optimal individual sometimes leads to blind search and the inability to find food. Random positions were introduced in the spiral foraging behavior stage through the relationship between the current and the maximum iterations. The aim is to expand the search space, increase randomness, improve the ability of the algorithm, and avoid getting trapped in local optimal solutions. The detailed formula of spiral foraging behavior is as follows:

$$X_i^{t+1} = \begin{cases} \alpha_1(X_{rand}^t + \beta \cdot |X_{rand}^t - X_i^t|) + \alpha_2 \cdot X_i^t, i = 1 \\ \alpha_1(X_{rand}^t + \beta \cdot |X_{rand}^t - X_i^t|) + \alpha_2 \cdot X_{i-1}^t, i = 2..N \\ \alpha_1(X_{best}^t + \beta \cdot |X_{best}^t - X_i^t|) + \alpha_2 \cdot X_i^t, i = 1 \\ \alpha_1(X_{best}^t + \beta \cdot |X_{best}^t - X_i^t|) + \alpha_2 \cdot X_{i-1}^t, i = 2..N \end{cases} \tag{1}$$

$$\alpha_1 = a + (1 - a) \cdot \frac{t}{T} \tag{2}$$

$$\alpha_2 = (1 - a) - (1 - a) \cdot \frac{t}{T} \tag{3}$$

$$\beta = e^{bl} . cos(2\pi b) \tag{4}$$

$$l = e^{3cos(((T+1/t)-1)\pi)} \tag{5}$$

where, $X_{rand}^t$ represents the position of the random individual, $\alpha_1$ and $\alpha_2$ devote the weight coefficients of each corresponding part, $\alpha_1 + \alpha_2 = 1$, $\beta$ is the spiral parameter, $X_{best}^t$ devotes the position of the best individual. When the random number is less than $t/T$, it is dominated by random individuals. Conversely, it is dominated by the optimal individual.

(b) Parabolic Foraging Behavior. In addition to spiral foraging behavior, tuna can also use food as a reference point to form a parabolic foraging pattern. By alternately using spiral foraging and parabolic foraging, TSO is prompted to conduct iterative optimization. The probabilities of choosing both are equal, both being 50%. The detailed formula of parabolic foraging behavior is as follows:

$$X_i^{t+1} = \begin{cases} X_{best}^t + rand \cdot (X_{best}^t - X_i^t) + TF \cdot p^2 \cdot (X_{best}^t - X_i^t) \\ \quad if\ rand < 0.5 \\ TF \cdot p^2 \cdot X_i^t \\ \quad if\ rand \geq 0.5 \end{cases} \tag{6}$$

$$p = (1 - \frac{t}{t_{max}})^{(t/t_{max})} \tag{7}$$

where, $TF$ is a random value of 1 or -1.

### B. Sine Selection Factor Strategy (SSF)

In traditional optimization algorithms, the design of the position update strategy has an important impact on the performance of the algorithm. Location update strategies usually rely on the comparison between random numbers and fixed thresholds (such as 0.5) to determine which update method to implement. Although this method is simple, it lacks consideration of the dynamic adaptability in the algorithm search process. When TSO was choosing between the two foraging behaviors, a random number was compared with the fixed value of 0.5. This approach may lead to low search efficiency of the algorithm or fall into local optima. Therefore, this paper designs the sine selection factor strategy and proposes a tuna optimization algorithm based on the sine selection factor (SSFTSO). The sinusoidal selection factor strategy, by introducing the sinusoidal oscillation mechanism, can dynamically adjust the selection of the two foraging behaviors, find the more suitable ranges for each of the two foraging behaviors, and thereby enhance the local development and global search effects of the algorithm. The advantages of the SSF strategy are as follows. (1) Unlike the conventional method, which employs a fixed threshold for judgment, SSF demonstrates dynamic adaptability and can adaptive select the position update strategy based on the iterative process. (2) Improve the convergence accuracy to enable the algorithm to approach the global optimal solution more precisely.

In order to further optimize the performance of the algorithm, a total of six different sine selection factors are designed and discussed in this paper, and each selection factor is achieved by adjusting the parameters. The calculation method of SSF strategy is shown as Eq. (9). The parameter selection comparison table of the sine selection factor is shown in Table I. Fig. 1 is the flowchart of SSFTSO algorithm after adding the sine selection factor strategy. The variation trends of various SSFS are shown in Fig. 2.

$$Oscillation = 0.05 \times sin(2\pi \cdot t/20) \tag{8}$$

$$SSF = a + (0.7 - a) \times \frac{t}{T} + Oscillation \tag{9}$$

$$a + b = 1 \tag{10}$$

where, $Oscillation$ represents the oscillation term controlled by the $sin$ function, and $SSF$ represents the sine selection factor.

## III. IEEE CEC-2022 Experiment and Analysis

The detailed content of this section is as follows. Sub-section *A* verifies the effectiveness of the sinusoidal selection factor strategy. Sub-section *B* compares SSFTSO with other algorithms. Experiments were conducted on the IEEE CEC-2022 test function, including test problems of different degrees, which can comprehensively test the effectiveness of the strategy and the ability of the proposed algorithm.

All experiments were run in the same environment to ensure fairness. Set the $dim = 10$, $pop$=$30$, and $T$=$500$. To ensure stability and reduce randomness, the average fitness value, average standard deviation and optimal value of each variant after 30 independent runs were recorded.

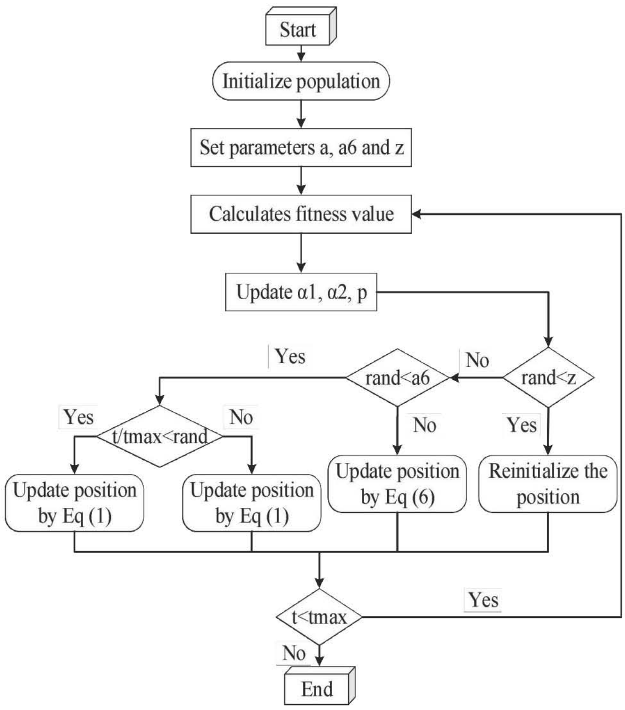


Fig. 1 Flowchart of SSFTSO algorithm.

TABLE I. PARAMETER SELECTION COMPARISON TABLE OF SINE SELECTION FACTORS

| No. | Abbreviation | Parameter settings of a | Parameter Settings of b |
|---|---|---|---|
| (1) | SSF-1 | a1 = 0.40 | b1 = 0.60 |
| (2) | SSF-2 | a2 = 0.45 | b2 = 0.55 |
| (3) | SSF-3 | a3 = 0.50 | b3 = 0.50 |
| (4) | SSF-4 | a4 = 0.55 | b4 = 0.45 |
| (5) | SSF-5 | a5 = 0.60 | b5 = 0.40 |
| (6) | SSF-6 | a6 = 0.65 | b6 = 0.35 |

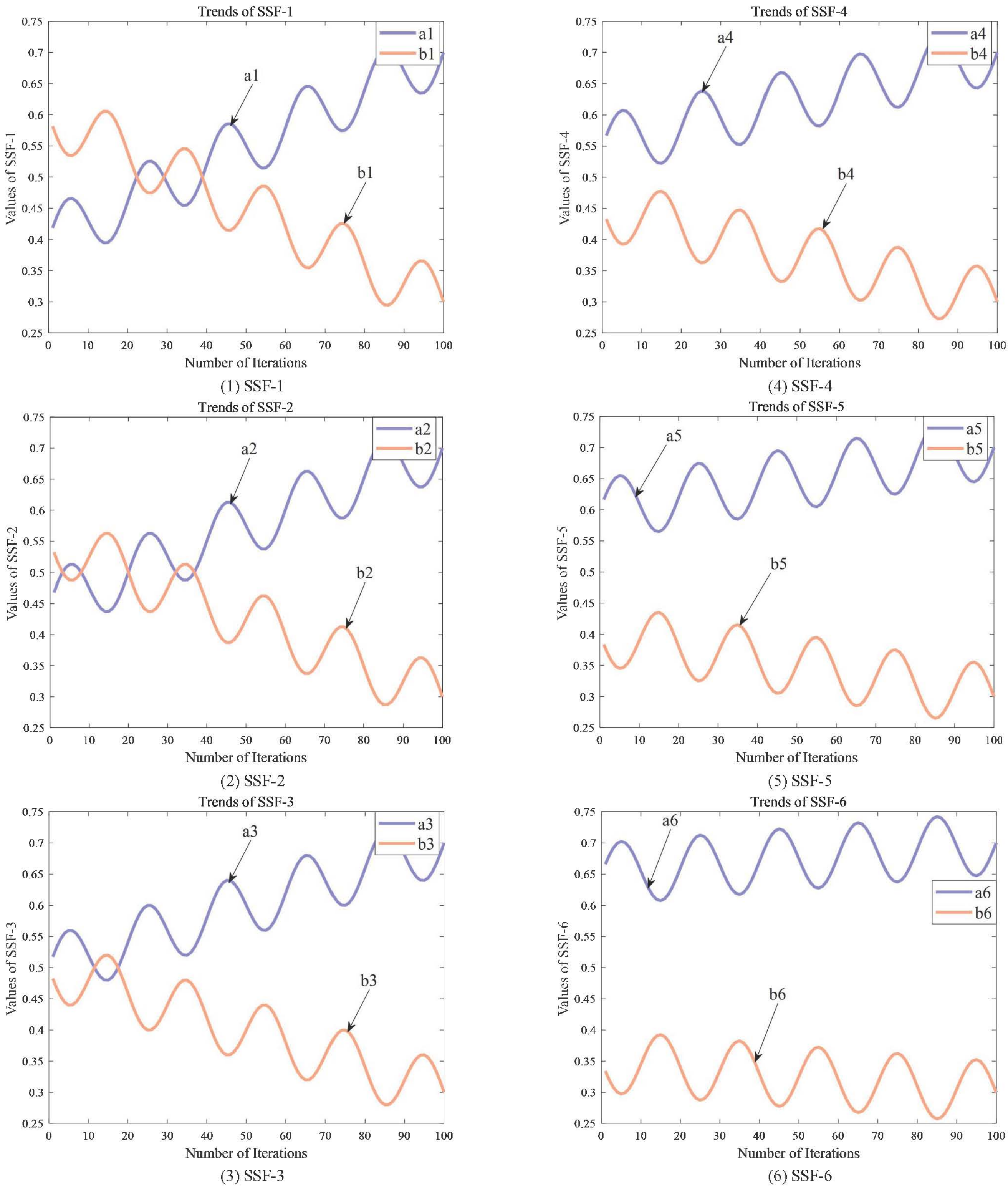



Fig. 2 The changing trend of various sine selection factors.

*A. Verify the Effectiveness of the Sine Selection Factor Strategy*

In terms of the performance comparison of each variant: Table II shows the results of TSO and its variants at IEEE CEC-2022. By comparing the experimental results on the IEEE CEC-2022, the advantages and disadvantages of different variants on different types of test functions can be discovered. On the unimodal function, compared with the original algorithm and other variants, SSF6-TSO is the best in terms of the average fitness value, the average standard deviation and the optimal value. On the basic function, SSF6-TSO obtained the minimum average fitness values at F3, F4 and F5. On F2, the average fitness value of SSF4-TSO is optimal. On the hybrid function, SSF5-TSO has the optimal average fitness at F6, SSF6-TSO achieves the minimum average fitness at F7, and SSF4-TSO obtains the optimal average fitness at F8. On the composition function, SSF5-TSO is optimal at F9 and F10, and the average fitness of SSF3-TSO and SSF6-TSO is superior to other variants at F11 and F12 respectively. It can be found from the above results that the sinusoidal selection factor strategy is effective, which improves the performance of the original TSO and can obtain a better fitness value. In addition, after conducting the Friedman test on the average fitness values of all algorithms, it was found that SSF6-TSO algorithm ranked first and had strong comprehensive strength.

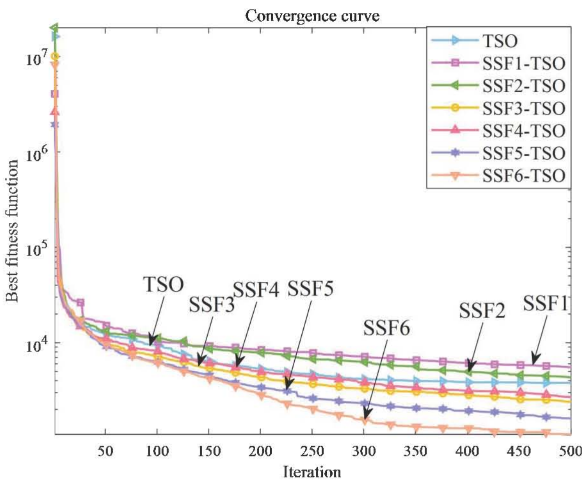
In terms of the comparison of convergence speeds of various variants, Fig. 3 shows the convergence curves of various sine selection factors of TSO and its variants. From the changing trend of the curves, the convergence characteristics of the algorithm can be intuitively observed. Except for F8, F10, F11 and F12, SSF6-TSO has greatly improved the convergence strength on other test functions and made tremendous efforts in terms of performance improvement. This indicates that the sinusoidal selection factor strategy can adaptive adjust the selection of the two foraging behaviors excellently, thereby enhancing the local exploitation and global search effects.

*B. Comparison of SSFTSO with Other Algorithms*

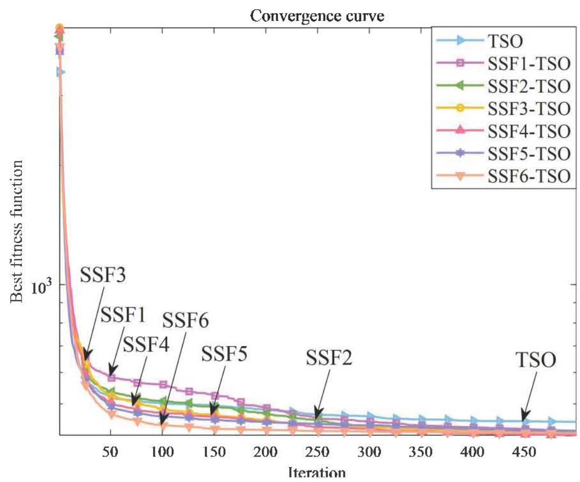
In the previous section, SSF6-TSO has the best performance and ranks first in the average fitness value. In order to reduce redundancy and facilitate comparison, SSF6-TSO is named SSFTSO. The experiment selected six population intelligence optimization algorithms to compare with the proposed algorithm to further explore the performance effect of the proposed algorithm, including PDO [19], WOA [20], HHO [21], AOA [22], OOA [23] and AO [24]. Table III is the parameter comparison table of each algorithm.

In terms of the performance comparison of various algorithms, Table IV shows the results of SSFTSO and other algorithms at IEEE CEC-2022. By comparing the experimental results on the IEEE CEC-2022, the advantages and disadvantages of each algorithm on different types of test functions can be discovered. On unimodal functions, basic functions, hybrid functions and composition functions, compared with other algorithms, SSFTSO has achieved the lowest average fitness value, showing good local search ability and being able to effectively escape the local optimum and find the global optimal solution. In terms of the optimal values, SSFTSO has obtained the optimal values on all functions. In addition, SSFTSO achieved the minimum standard deviation of average fitness on the F6, F8, F9, F10 and F12 functions. This indicates that the stability has a relatively good performance. After conducting the Friedman test on the average fitness data of all algorithms, it was found that SSFTSO algorithm ranked first, proving that the SSFTSO has advantages and can be considered for use and expansion.

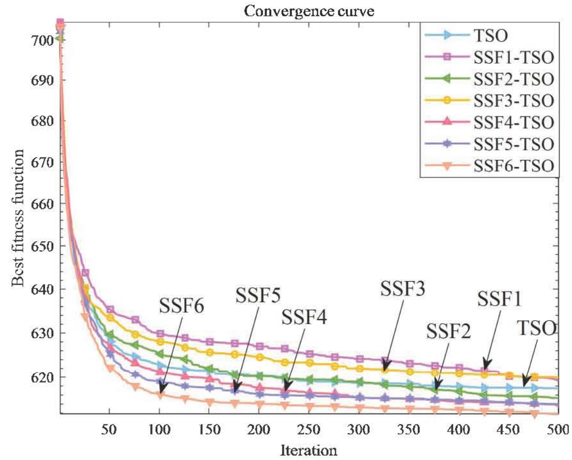
In terms of the comparison of convergence speeds of various algorithms, Fig. 4 shows the convergence curves of SSFTSO and other comparison algorithms. From the changing trends of the curves, the convergence characteristics of each algorithm can be intuitively observed. On the F1 and F4 functions, although the convergence speed of SSFTSO is not as fast as that of other algorithms, with the increase of the number of iterations, it shows a strong iterative optimization and search ability by itself, and can continuously explore and converge to a better solution. On the F8 function, the distance between SSFTSO and other algorithms is very close, but the optimal solution can still be obtained. On other functions, SSFTSO demonstrates its unique advantages and is strong both in terms of convergence speed and degree. The above results and analysis fully demonstrate the significant ability of SSFTSO after adding the sinusoidal selection factor strategy.



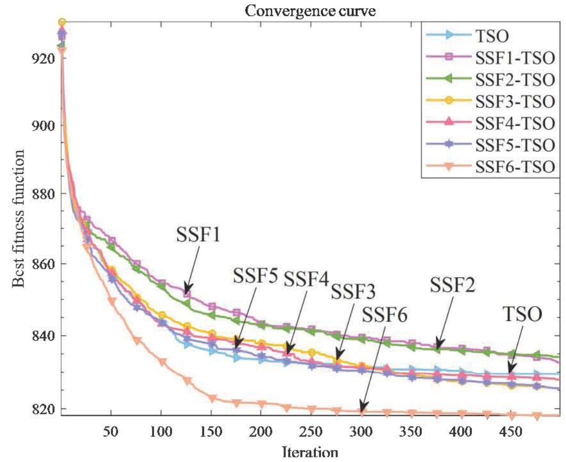

(1) F1



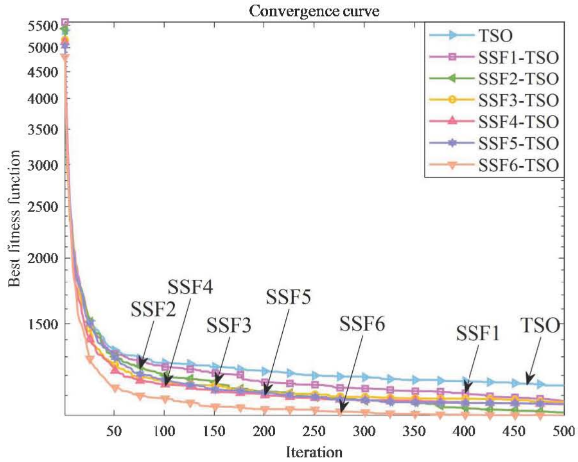

(2) F2



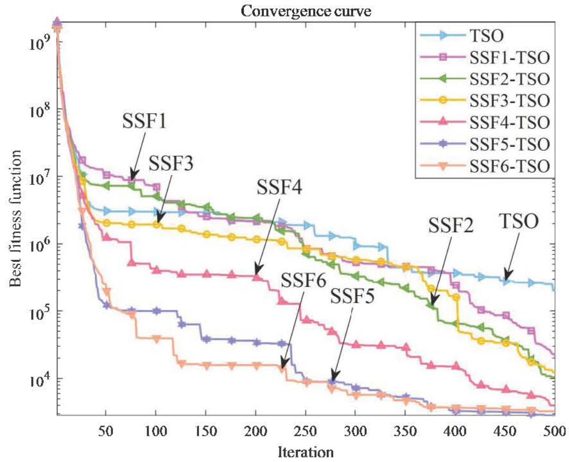
(3) F3



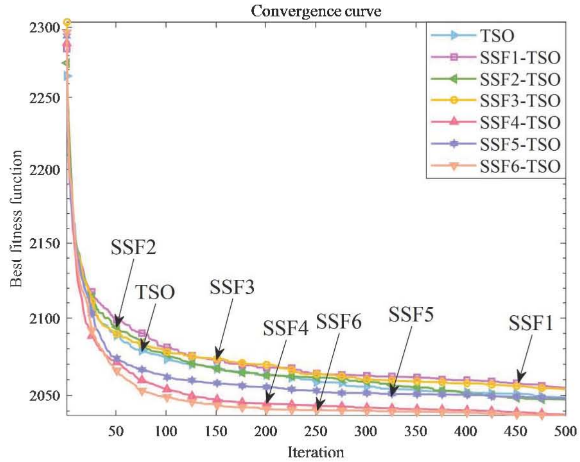
(4) F4



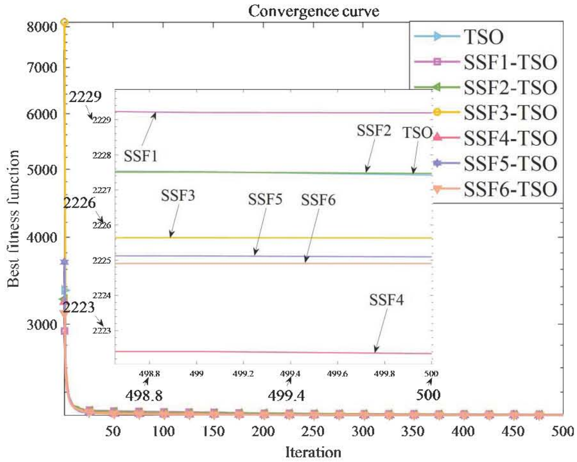
(5) F5



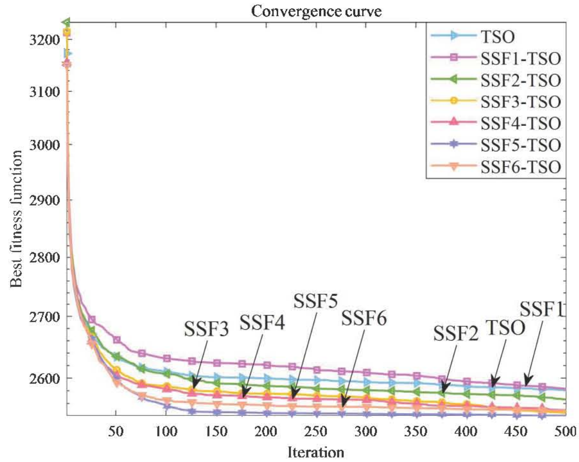
(6) F6



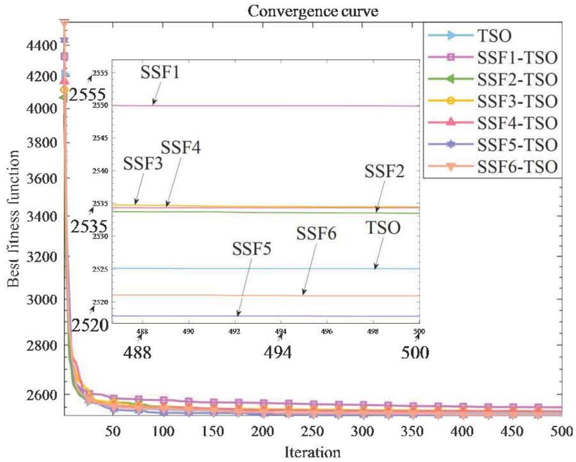
(7) F7



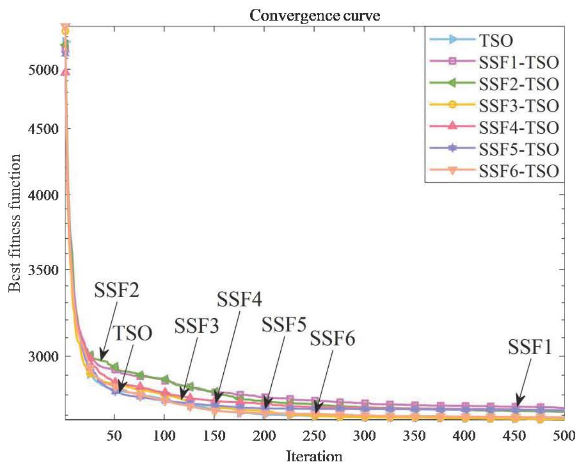
(8) F8



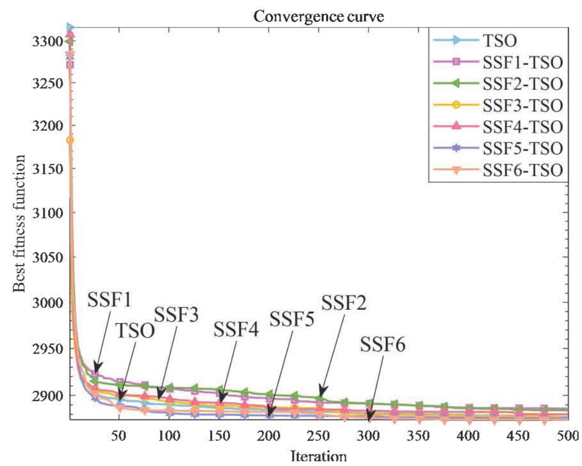
(9) F9



(10) F10




(11) F11




(12) F12

Fig. 3 Convergence curves of various sine selection factors on IEEE CEC-2022.

TABLE II. EXPERIMENTAL RESULTS OF TSO AND ITS VARIANTS AT IEEE CEC-2022

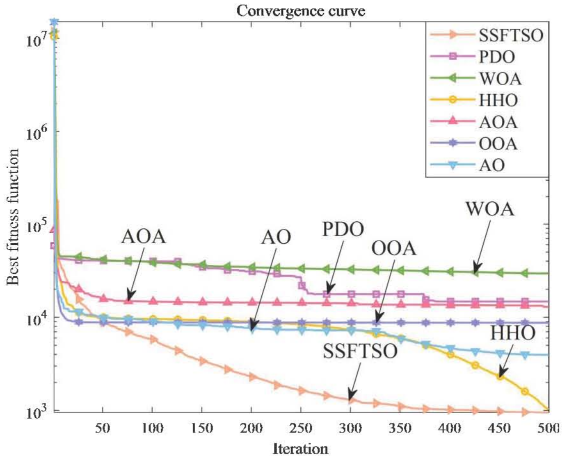
| $F$ | | TSO | SSF1-TSO | SSF2-TSO | SSF3-TSO | SSF4-TSO | SSF5-TSO | SSF6-TSO |
|---|---|---|---|---|---|---|---|---|
| | Ave | 3.7813E+03 | 5.4971E+03 | 4.3309E+03 | 2.3980E+03 | 2.6894E+03 | 1.6076E+03 | **1.0967E+03** |
| $F_1$ | Std | 4.0817E+03 | 3.2356E+03 | 3.4705E+03 | 3.1148E+03 | 2.7673E+03 | 2.3520E+03 | **1.8157E+03** |
| | Best | 3.0001E+02 | 3.0006E+02 | **3.0000E+02** | **3.0000E+02** | **3.0000E+02** | **3.0000E+02** | **3.0000E+02** |
| | Ave | 4.5035E+02 | 4.2752E+02 | 4.2284E+02 | 4.2091E+02 | **4.1693E+02** | 4.2751E+02 | 4.2172E+02 |
| $F_2$ | Std | 6.2112E+01 | 2.8154E+01 | 2.4613E+01 | 2.8402E+01 | **2.3661E+01** | 6.4976E+01 | 4.0243E+01 |
| | Best | **4.0000E+02** | **4.0000E+02** | 4.0002E+02 | **4.0000E+02** | 4.0003E+02 | **4.0000E+02** | 4.0001E+02 |
| | Ave | 6.1741E+02 | 6.1922E+02 | 6.1534E+02 | 6.1994E+02 | 6.1356E+02 | 6.1400E+02 | **6.1189E+02** |
| $F_3$ | Std | 1.2465E+01 | 1.0480E+01 | 9.4377E+00 | 1.3814E+01 | 1.0799E+01 | 7.5077E+00 | **6.6963E+00** |
| | Best | 6.0356E+02 | 6.0406E+02 | 6.0108E+02 | 6.0298E+02 | 6.0159E+02 | 6.0336E+02 | **6.0089E+02** |
| | Ave | 8.2943E+02 | 8.3264E+02 | 8.3414E+02 | 8.2546E+02 | 8.2779E+02 | 8.2544E+02 | **8.1821E+02** |
| $F_4$ | Std | 2.0217E+01 | 1.7644E+01 | 1.8965E+01 | 1.2516E+01 | 1.3738E+01 | 9.8956E+00 | **7.9548E+00** |
| | Best | 8.0497E+02 | **8.0398E+02** | 8.0696E+02 | 8.0497E+02 | 8.1194E+02 | 8.0995E+02 | 8.0597E+02 |
| | Ave | 1.1481E+03 | 1.0731E+03 | 1.0214E+03 | 1.0688E+03 | 1.0576E+03 | 1.0595E+03 | **1.0080E+03** |
| $F_5$ | Std | 2.8733E+02 | 2.0363E+02 | 1.0951E+02 | 2.1854E+02 | 1.1518E+02 | 1.2190E+02 | **9.1123E+01** |
| | Best | **9.0252E+02** | 9.0443E+02 | 9.2070E+02 | 9.0319E+02 | 9.0261E+02 | 9.1506E+02 | 9.0454E+02 |
| | Ave | 1.9527E+05 | 2.1264E+04 | 1.0347E+04 | 1.2260E+04 | 3.9858E+03 | **2.8360E+03** | 3.2608E+03 |
| $F_6$ | Std | 5.7078E+05 | 3.0884E+04 | 1.7725E+04 | 3.0655E+04 | 2.7617E+03 | 1.4307E+03 | **1.3783E+03** |
| | Best | 1.8632E+03 | 1.9873E+03 | 1.8576E+03 | 1.8581E+03 | 1.8573E+03 | **1.8382E+03** | 1.8398E+03 |
| | Ave | 2.0487E+03 | 2.0548E+03 | 2.0473E+03 | 2.0539E+03 | 2.0376E+03 | 2.0482E+03 | **2.0372E+03** |
| $F_7$ | Std | 2.7824E+01 | 3.5902E+01 | 2.4368E+01 | 2.3055E+01 | **1.5180E+01** | 2.0975E+01 | 1.8096E+01 |
| | Best | 2.0086E+03 | 2.0180E+03 | **2.0037E+03** | 2.0179E+03 | 2.0111E+03 | 2.0224E+03 | 2.0207E+03 |
| | Ave | 2.2274E+03 | 2.2292E+03 | 2.2275E+03 | 2.2256E+03 | **2.2223E+03** | 2.2251E+03 | 2.2249E+03 |
| $F_8$ | Std | 9.5966E+00 | 1.0303E+01 | 6.4289E+00 | 5.8248E+00 | 8.4678E+00 | 6.2802E+00 | **4.6602E+00** |
| | Best | 2.2081E+03 | 2.2099E+03 | 2.2133E+03 | 2.2102E+03 | **2.2009E+03** | 2.2034E+03 | 2.2104E+03 |
| | Ave | 2.5812E+03 | 2.5833E+03 | 2.5665E+03 | 2.5452E+03 | 2.5497E+03 | **2.5411E+03** | 2.5487E+03 |
| $F_9$ | Std | 5.9350E+01 | 6.4186E+01 | 5.6303E+01 | **2.8038E+01** | 4.0925E+01 | 3.7207E+01 | 4.6787E+01 |
| | Best | **2.5293E+03** | **2.5293E+03** | **2.5293E+03** | **2.5293E+03** | **2.5293E+03** | **2.5293E+03** | **2.5293E+03** |
| | Ave | 2.5250E+03 | 2.5499E+03 | 2.5334E+03 | 2.5344E+03 | 2.5343E+03 | **2.5178E+03** | 2.5209E+03 |
| $F_{10}$ | Std | 5.3696E+01 | 7.0680E+01 | 6.0028E+01 | 5.6677E+01 | 5.6714E+01 | **4.3849E+01** | 4.5522E+01 |
| | Best | 2.5005E+03 | **2.5004E+03** | 2.5005E+03 | **2.5004E+03** | 2.5005E+03 | 2.5006E+03 | **2.5004E+03** |
| | Ave | 2.6851E+03 | 2.7351E+03 | 2.7171E+03 | **2.6792E+03** | 2.7233E+03 | 2.7244E+03 | 2.6904E+03 |
| $F_{11}$ | Std | **1.0225E+02** | 1.3819E+02 | 1.1599E+02 | 1.1315E+02 | 1.3568E+02 | 1.2364E+02 | 1.4399E+02 |
| | Best | **2.6000E+03** | **2.6000E+03** | **2.6000E+03** | **2.6000E+03** | **2.6000E+03** | **2.6000E+03** | **2.6000E+03** |
| | Ave | 2.8773E+03 | 2.8860E+03 | 2.8843E+03 | 2.8756E+03 | 2.8801E+03 | 2.8759E+03 | **2.8744E+03** |
| $F_{12}$ | Std | 2.8677E+01 | 2.5759E+01 | 3.8489E+01 | 1.8321E+01 | 2.7773E+01 | 2.8702E+01 | **1.5612E+01** |
| | Best | 2.8639E+03 | 2.8639E+03 | 2.8639E+03 | **2.8626E+03** | **2.8626E+03** | **2.8626E+03** | 2.8640E+03 |
| Friedman | | 5.0833 | 6.5833 | 4.5833 | 3.8333 | 3.2500 | 2.9167 | **1.7500** |
| Rank | | 6 | 7 | 5 | 4 | 3 | 2 | **1** |

TABLE III. PARAMETER COMPARISON TABLE OF EACH ALGORITHM

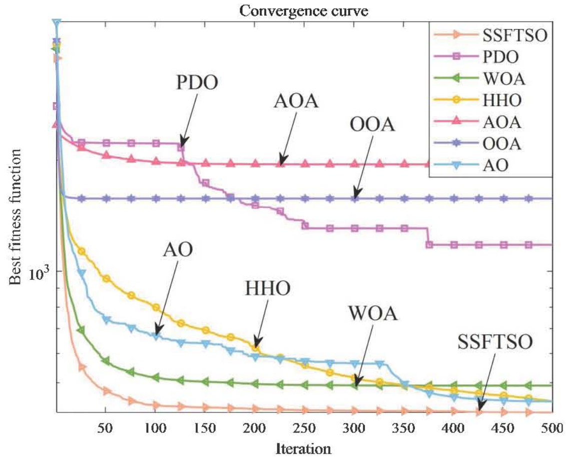
| Full title | Abbreviation | Parameters setting |
|---|---|---|
| Tuna Swarm Optimization Algorithm with Sine Selection Factor | SSFTSO | $a = 0.7, z = 0.05, a6 = 0.65$ |
| Prairie Dog Optimization Algorithm | PDO | $\rho = 0.1, \varepsilon = 0.005$ |
| Whale Optimization Algorithm | WOA | $b = 1$ |
| Harris Hawks Optimization | HHO | $\beta = 1.5$ |
| Arithmetic Optimization Algorithm | AOA | $c_1 = 2, c_2 = 6,, u = 0.9, l = 0.1, c_3 = 1, c_4 = 2$ |
| Osprey Optimization Algorithm | OOA | $I = 1 or 2, cf = 0.63$ |
| Aquila Optimizer | AO | $\alpha = 0.1, \delta = 0.1, \mu = 0.0265$ |

TABLE IV. EXPERIMENTAL RESULTS OF SSFTSO AND OTHER ALGORITHMS AT IEEE CEC-2022

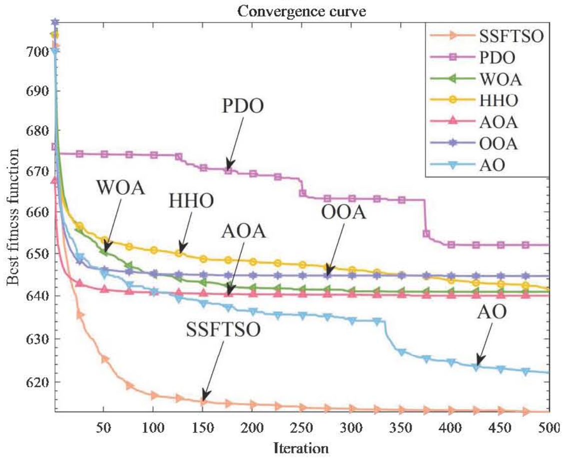
| $F$ | | SSFTSO | PDO | WOA | HHO | AOA | OOA | AO |
|---|---|---|---|---|---|---|---|---|
| $F_1$ | Ave | **9.5228E+02** | 1.4756E+04 | 2.9542E+04 | 1.0309E+03 | 1.2934E+04 | 8.7372E+03 | 3.9559E+03 |
| | Std | 1.6498E+03 | 6.2544E+03 | 1.2530E+04 | **4.8013E+02** | 5.1367E+03 | 1.8869E+03 | 1.2922E+03 |
| | Best | **3.0000E+02** | 7.0426E+03 | 9.5144E+03 | 3.9202E+02 | 5.5718E+03 | 5.3715E+03 | 1.7192E+03 |
| $F_2$ | Ave | **4.1558E+02** | 1.1781E+03 | 4.9134E+02 | 4.4591E+02 | 1.9510E+03 | 1.5773E+03 | 4.4535E+02 |
| | Std | 2.4091E+01 | 3.4504E+02 | **1.1254E+02** | 3.9849E+01 | 1.2234E+03 | 7.0856E+02 | 2.7153E+01 |
| | Best | **4.0001E+02** | 5.4636E+02 | 4.0636E+02 | 4.0022E+02 | 7.2033E+02 | 6.1031E+02 | 4.1085E+02 |
| $F_3$ | Ave | **6.1326E+02** | 6.5201E+02 | 6.4096E+02 | 6.4130E+02 | 6.4005E+02 | 6.4467E+02 | 6.2218E+02 |
| | Std | 7.1628E+00 | 8.4102E+00 | 1.5885E+01 | 1.0741E+01 | 8.5411E+00 | 1.0979E+01 | **6.7597E+00** |
| | Best | **6.0373E+02** | 6.4042E+02 | 6.1684E+02 | 6.2200E+02 | 6.2320E+02 | 6.2268E+02 | 6.0608E+02 |
| $F_4$ | Ave | **8.2128E+02** | 8.5399E+02 | 8.4350E+02 | 8.2659E+02 | 8.3913E+02 | 8.5148E+02 | 8.2302E+02 |
| | Std | 9.6714E+00 | 1.4467E+01 | 1.6318E+01 | 8.9022E+00 | 1.0231E+01 | 9.6907E+00 | **6.4846E+00** |
| | Best | **8.0796E+02** | 8.2485E+02 | 8.1517E+02 | 8.1132E+02 | 8.1954E+02 | 8.2926E+02 | 8.1124E+02 |
| $F_5$ | Ave | **1.0254E+03** | 1.5241E+03 | 1.5720E+03 | 1.4079E+03 | 1.3977E+03 | 1.4007E+03 | 1.0494E+03 |
| | Std | 1.2124E+02 | 2.2247E+02 | 3.6765E+02 | 1.7053E+02 | 1.7991E+02 | 2.1227E+02 | **1.1360E+02** |
| | Best | **9.0770E+02** | 1.0370E+03 | 1.0442E+03 | 1.0713E+03 | 1.1443E+03 | 1.0745E+03 | 9.2519E+02 |
| $F_6$ | Ave | **2.9564E+03** | 2.3944E+08 | 4.8633E+03 | 7.9099E+03 | 5.3155E+07 | 3.1965E+06 | 2.8803E+04 |
| | Std | **1.3019E+03** | 3.4093E+08 | 2.6766E+03 | 4.9481E+03 | 1.4280E+08 | 5.8942E+06 | 2.3628E+04 |
| | Best | **1.8521E+03** | 8.3039E+05 | 1.9424E+03 | 2.1136E+03 | 1.9921E+03 | 1.9014E+03 | 2.7877E+03 |
| $F_7$ | Ave | **2.0346E+03** | 2.1170E+03 | 2.0800E+03 | 2.0889E+03 | 2.1048E+03 | 2.0878E+03 | 2.0482E+03 |
| | Std | 1.6379E+01 | 3.2334E+01 | 3.6894E+01 | 4.3886E+01 | 4.0011E+01 | 2.1308E+01 | **1.1988E+01** |
| | Best | **2.0090E+03** | 2.0642E+03 | 2.0278E+03 | 2.0309E+03 | 2.0314E+03 | 2.0588E+03 | 2.0274E+03 |
| $F_8$ | Ave | **2.2226E+03** | 2.2685E+03 | 2.2361E+03 | 2.2354E+03 | 2.2797E+03 | 2.2325E+03 | 2.2299E+03 |
| | Std | **4.0221E+00** | 3.5547E+01 | 9.3957E+00 | 1.4331E+01 | 7.2099E+01 | 8.0689E+00 | 4.0831E+00 |
| | Best | **2.2077E+03** | 2.2338E+03 | 2.2238E+03 | 2.2241E+03 | 2.2264E+03 | 2.2180E+03 | 2.2199E+03 |
| $F_9$ | Ave | **2.5375E+03** | 2.7535E+03 | 2.6064E+03 | 2.6213E+03 | 2.7678E+03 | 2.7436E+03 | 2.6117E+03 |
| | Std | **2.3561E+01** | 5.9089E+01 | 5.5471E+01 | 4.0421E+01 | 7.0509E+01 | 4.3326E+01 | 3.7851E+01 |
| | Best | **2.5293E+03** | 2.6327E+03 | 2.5327E+03 | 2.5567E+03 | 2.6731E+03 | 2.6652E+03 | 2.5543E+03 |
| $F_{10}$ | Ave | **2.5252E+03** | 2.6722E+03 | 2.6830E+03 | 2.6063E+03 | 2.7451E+03 | 2.7426E+03 | 2.5695E+03 |
| | Std | **4.9796E+01** | 3.3614E+02 | 2.6810E+02 | 1.4267E+02 | 2.4518E+02 | 2.6177E+02 | 6.0659E+01 |
| | Best | **2.5004E+03** | 2.5138E+03 | 2.5008E+03 | 2.5008E+03 | 2.5116E+03 | 2.5222E+03 | 2.5009E+03 |
| $F_{11}$ | Ave | **2.6988E+03** | 3.1028E+03 | 2.9003E+03 | 2.8349E+03 | 3.5391E+03 | 3.6527E+03 | 2.7322E+03 |
| | Std | 1.6827E+02 | 3.4365E+02 | 1.7644E+02 | 1.8346E+02 | 4.5861E+02 | 4.1796E+02 | **7.3364E+01** |
| | Best | **2.6000E+03** | 2.7364E+03 | 2.7338E+03 | 2.6086E+03 | 2.8130E+03 | 2.8641E+03 | 2.6091E+03 |
| $F_{12}$ | Ave | **2.8700E+03** | 2.8902E+03 | 2.8964E+03 | 2.9188E+03 | 3.0215E+03 | 3.0681E+03 | 2.8714E+03 |
| | Std | **8.7562E+00** | 9.9527E+00 | 2.9785E+01 | 5.6873E+01 | 6.1949E+01 | 8.8954E+01 | 9.5839E+00 |
| | Best | **2.8626E+03** | 2.8714E+03 | 2.8628E+03 | 2.8655E+03 | 2.9282E+03 | 2.9202E+03 | 2.8633E+03 |
| Friedman | | **1.0000** | 5.7500 | 4.3333 | 3.7500 | 5.5833 | 5.2500 | 2.3333 |
| Rank | | **1** | 7 | 4 | 3 | 6 | 5 | 2 |



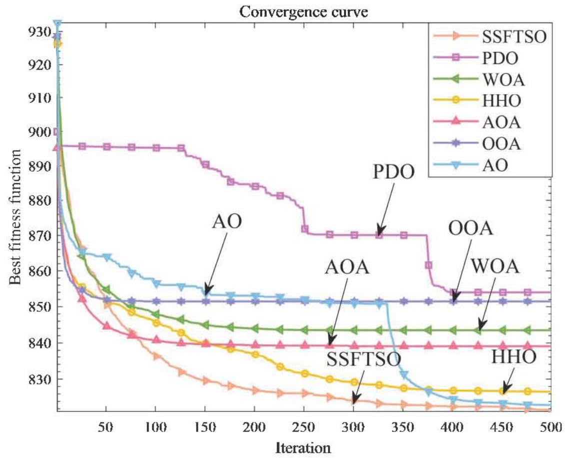
(1) F1



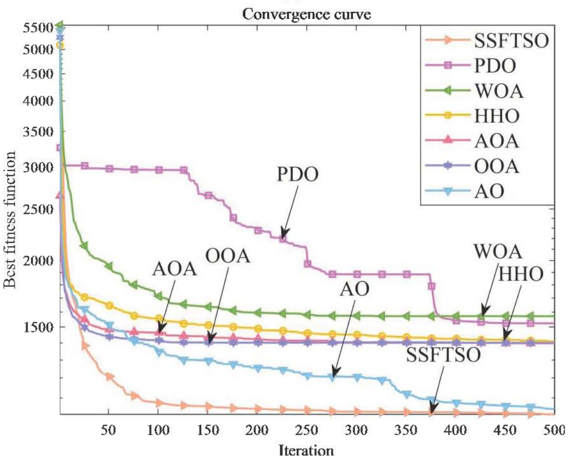
(2) F2



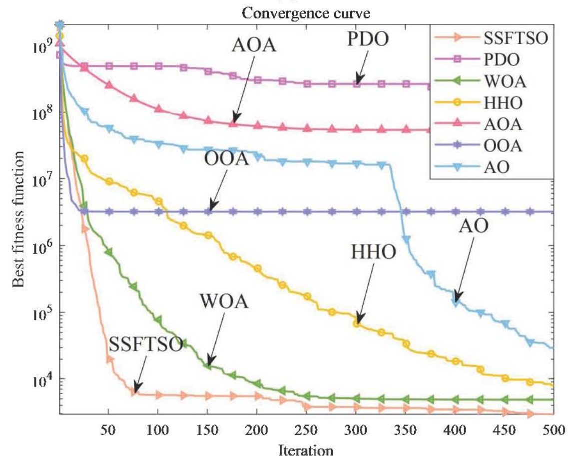
(3) F3



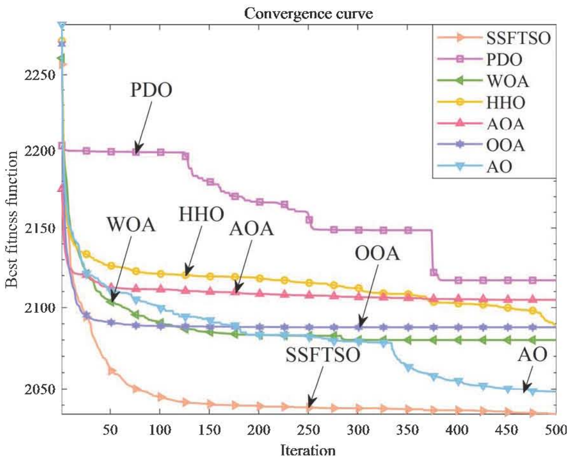
(4) F4



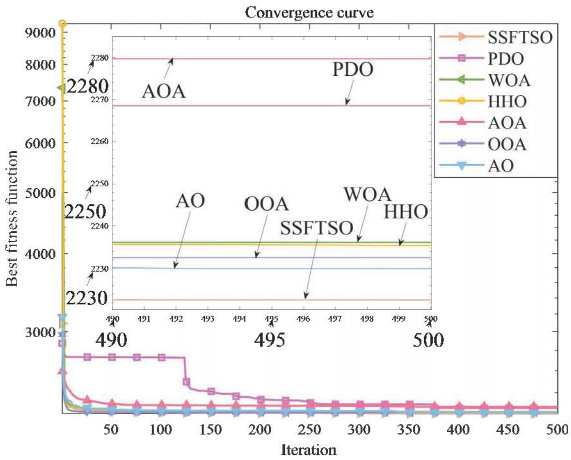
(5) F5



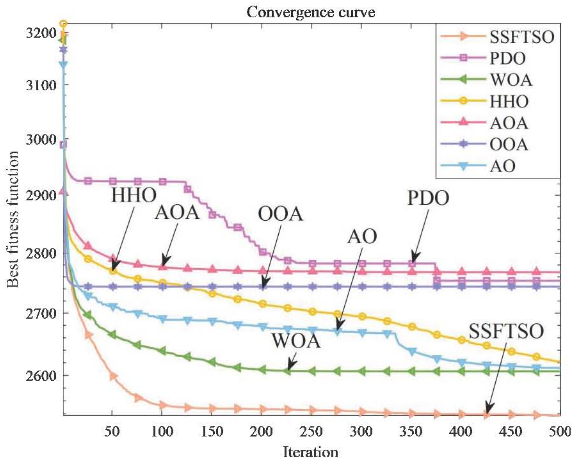
(6) F6



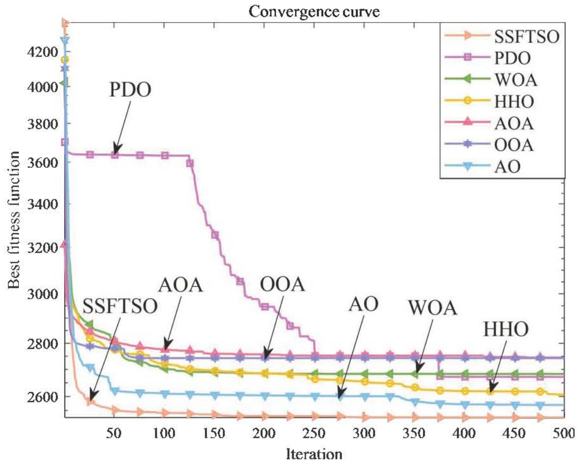
(7) F7



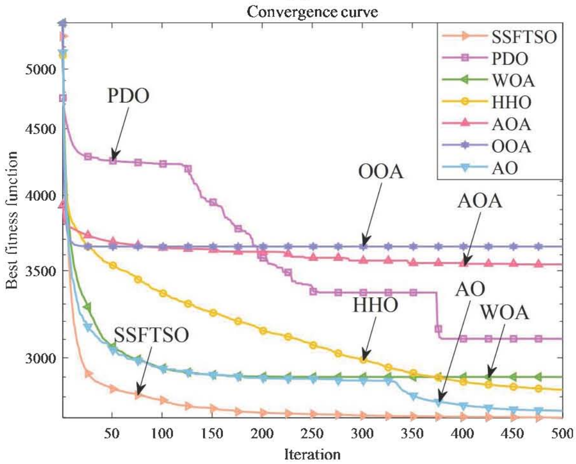
(8) F8



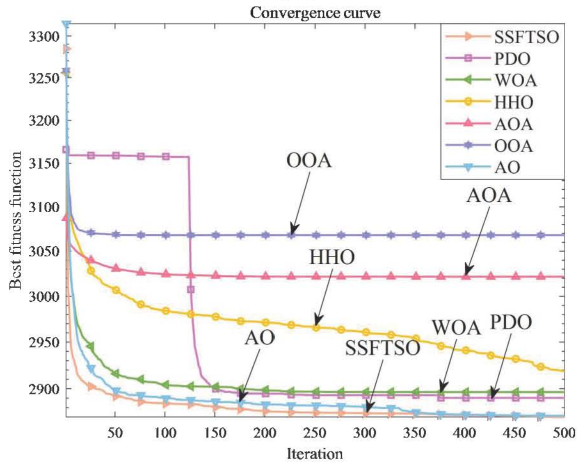
(9) F9



(10) F10



(11) F11



(12) F12

Fig. 4 Convergence curves of SSFTSO and other algorithms on IEEE CEC-2022.

## IV. Tuna Swarm Optimization Algorithm with Sine Selection Factor to Solve High-dimensional Feature Selection

To continue exploring the performance and effect of SSFTSO, this section uses SSFTSO to solve high-dimensional FS problems. Section *A* introduces the selected high-dimensional data set. Section *B* is about the specific settings of parameters in the experiment. Section *C* presents the experimental results and analysis.

### *A. Datasets Description*

Ten public high-dimensional datasets were selected to test the performance of the algorithm, which covered many fields. The number of featuress, specific instance and the categories of each data set are shown in Table V.

### *B. Parameter Settings*

To reduce the influence of randomness on the experiment, fivefold cross-validation was used to test the performance of the model. The method of taking the average of 30 cycles was adopted to statistically analyze the results such as the average fitness value, the number of selected features, and the accuracy rate. In addition, in FS, the selection of classifiers and the setting of fitness functions are involved. This paper selects the parameters in the references [25-26]. In this paper, the KNN classifier is used, and the weight values in the fitness function are 0.99 and 0.01 respectively. The parameters of the comparison algorithm have not changed and remain consistent with the previous text. The detailed Settings and explanations of other parameters are shown in Table VI.

Table V. Parameter table of 10 public high-dimensional datasets

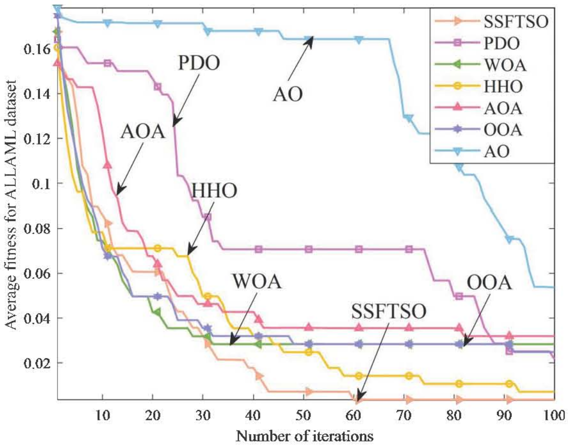
| No. | Datasets | Features | Instances | Classes |
|---|---|---|---|---|
| (1) | ALLAML | 7129 | 72 | 2 |
| (2) | warpPIE10P | 2420 | 210 | 10 |
| (3) | Brain Tumor1 | 5920 | 90 | 5 |
| (4) | Brain Tumor2 | 10367 | 50 | 4 |
| (5) | Yale | 1024 | 165 | 15 |
| (6) | Cancer | 12600 | 203 | 4 |
| (7) | DLBCL | 5469 | 77 | 2 |
| (8) | DrivFace | 6400 | 606 | 3 |
| (9) | GLI85 | 22283 | 85 | 2 |
| (10) | Har | 561 | 270 | 6 |

TABLE VI. PARAMETER SETTINGS AND DESCRIPTION TABLE

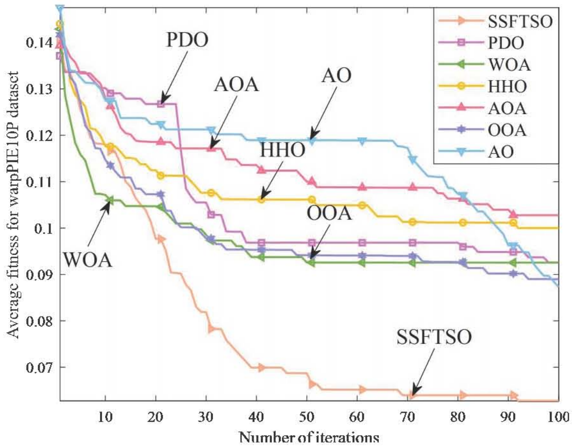
| Symbol description | Abbreviation | Parameter settings |
|---|---|---|
| Fitness function | $fit$ | $\alpha = 0.99,\ \beta = 0.01$ |
| K-nearest neighbor classifier | $KNN$ | $KNN = 5$ |
| Population size | $Pop$ | $Pop = 10$ |
| Maximum number of iterations | $Max\_T$ | $Max\_T = 100$ |
| Dimension | $Dim$ | The dimension sizes corresponding to each data set |

### C. *Results and Analysis*

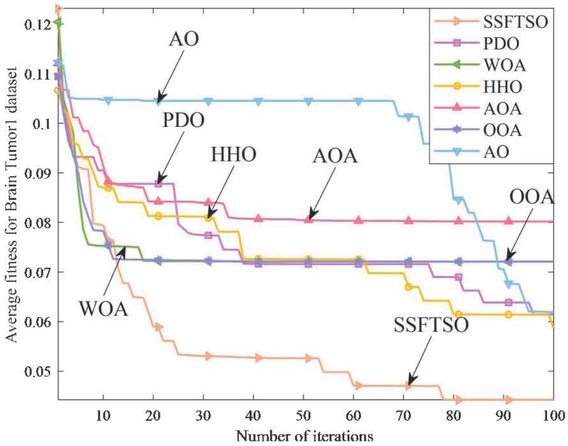
In terms of the performance comparison between SSFTSO and other algorithms: Table VII presents the average fitness values and standard deviation experimental results of SSFTSO and other algorithms. It can be found from the table that SSFTSO has achieved the best average fitness on all datasets. Except for the four datasets (2,3,5,6), SSFTSO has the lowest standard deviation of the average fitness. The above results indicate that SSFTSO has superior optimization and stability when dealing with FS problems. Subsequently, PDO achieved better standard deviations of the average fitness on two datasets (3 and 5). Finally, the standard deviations of the average fitness values of HHO and AO on datasets (2) and (6) are the best respectively. In addition, the Friedman test for the average fitness of all algorithms was conducted. After testing, SSFTSO ranked first, once again proving the advantages of the SSFTSO. Table VIII presents the experimental results of the average accuracy rates of SSFTSO and other algorithms. It can be found from this that SSFTSO has achieved the highest average accuracy rate on all datasets. Table IX shows the experimental results of the average number of selected features of SSFTSO and other algorithms. It can be found from the table that SSFTSO has achieved the minimum average number of selected features on all three datasets (3,6,9). For other algorithms, AOA obtained the optimal values on four datasets (2,5,7,8). HHO, OOA and AO respectively selected the minimum average number of features on data set (1), data set (4) and data set (10). Overall, after the Friedman test of the average number of selected features, SSFTSO ranked first, indicating that the average strength of the proposed algorithm is stronger. Table X shows the results of the average running time of SSFTSO and other algorithms. It can be found from this that SSFTSO spent the least time on the seven datasets (1,2,3,4,5,7,10). WOA has the least average running time on the remaining three datasets (6,8,9). After the Friedman test of the average running time, SSFTSO ranked first, indicating that SSFTSO has an advantage in the required time. In terms of the comparison of convergence curves between SSFTSO and other algorithms: Fig. 5 shows the average fitness value convergence curves of SSFTSO and other algorithms on different datasets.



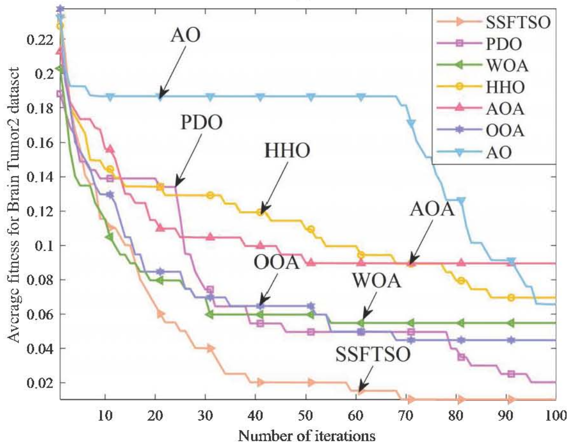

(1)




(2)




(3)




(4)

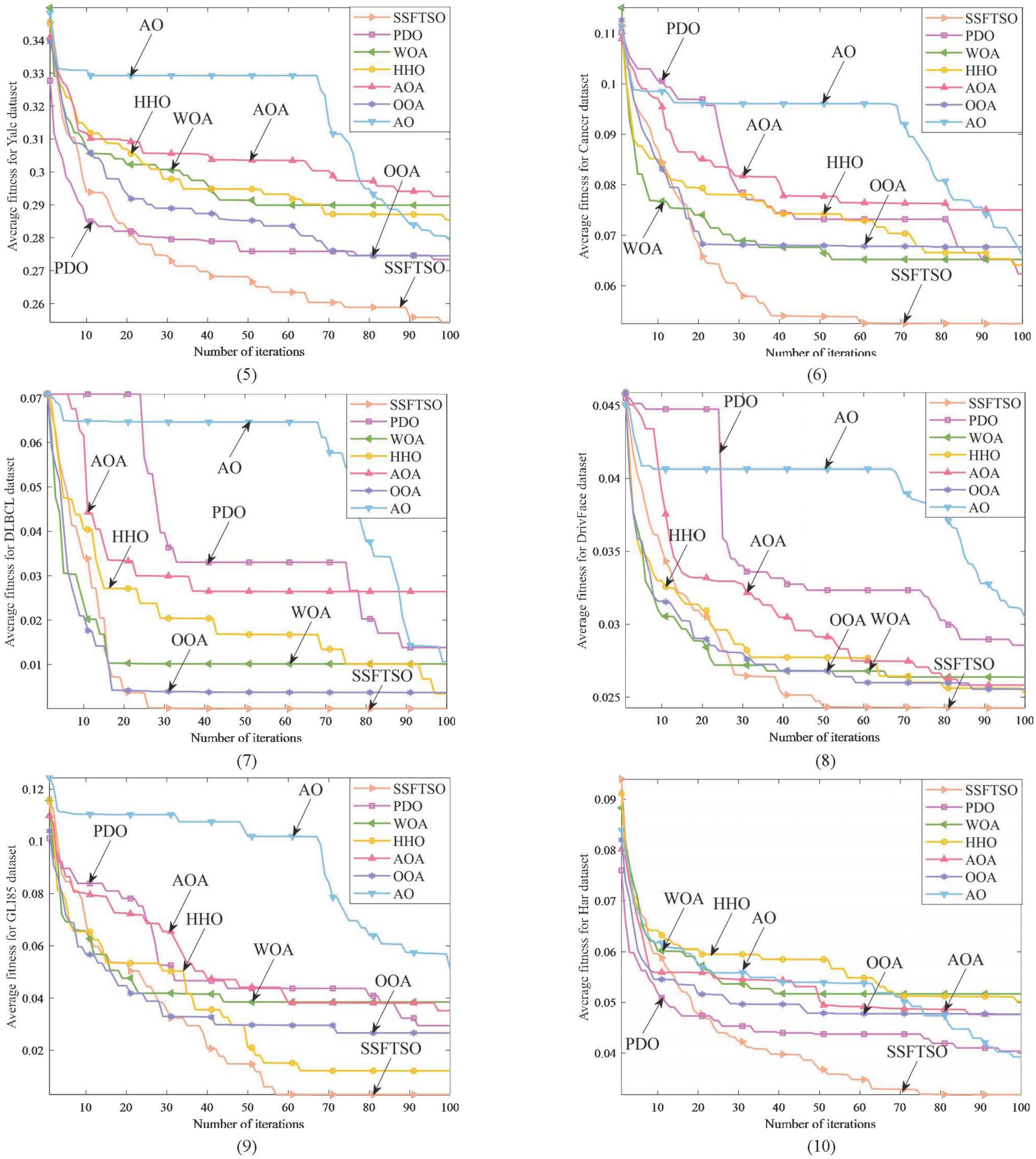
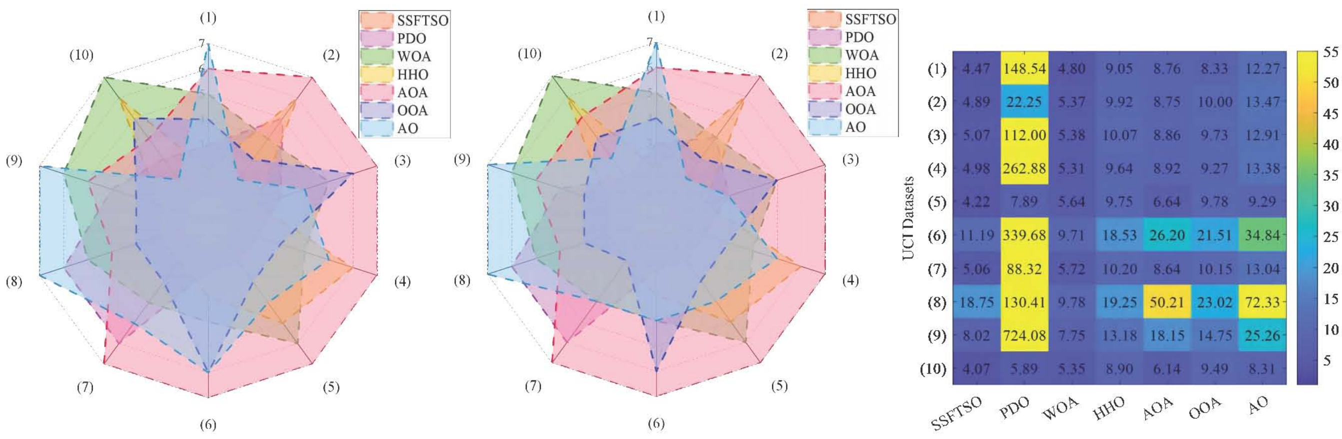



Fig. 5 Convergence curves of SSFTSO and other algorithms.




(a) Average fitness (b) Average accuracy (c) Average running time

Fig. 6 Ranking radar charts and heat chart of SSFTSO and other algorithms.

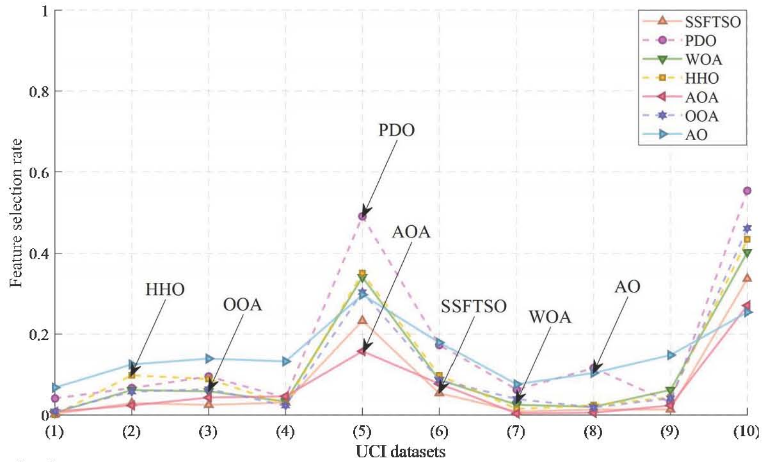



Fig. 7 Line trend chart of feature selection rate.

TABLE VII. AVERAGE FITNESS AND STANDARD DEVIATION OF SSFTSO AND OTHER ALGORITHMS

| Datasets | Measure | SSFTSO | PDO | WOA | HHO | AOA | OOA | AO |
|---|---|---|---|---|---|---|---|---|
| (1) | Ave | **3.5632E-03** | 2.1624E-02 | 2.8344E-02 | 7.0797E-03 | 3.1900E-02 | 2.4836E-02 | 5.3712E-02 |
| | Std | **1.5806E-02** | 3.3488E-02 | 3.5542E-02 | 2.1766E-02 | 4.2723E-02 | 3.4600E-02 | 3.8860E-02 |
| (2) | Ave | **6.2756E-02** | 9.2595E-02 | 9.2544E-02 | 9.9981E-02 | 1.0277E-01 | 8.8967E-02 | 8.7285E-02 |
| | Std | 1.9163E-02 | 2.4225E-02 | 2.1782E-02 | **1.6779E-02** | 1.9175E-02 | 1.8539E-02 | 1.7809E-02 |
| (3) | Ave | **4.4249E-02** | 6.1452E-02 | 7.2085E-02 | 5.8639E-02 | 8.0180E-02 | 7.2138E-02 | 6.1893E-02 |
| | Std | 2.2464E-02 | **1.6744E-02** | 2.5735E-02 | 2.8088E-02 | 2.7720E-02 | 2.5870E-02 | 2.4434E-02 |
| (4) | Ave | **1.0206E-02** | 2.0230E-02 | 5.4782E-02 | 6.9565E-02 | 8.9561E-02 | 4.4781E-02 | 6.5674E-02 |
| | Std | **3.0851E-02** | 4.0565E-02 | 6.0014E-02 | 6.4993E-02 | 5.4769E-02 | 5.0485E-02 | 5.8480E-02 |
| (5) | Ave | **2.5433E-01** | 2.7341E-01 | 2.8991E-01 | 2.8551E-01 | 2.9258E-01 | 2.7454E-01 | 2.7897E-01 |
| | Std | 3.4759E-02 | **2.4092E-02** | 2.8121E-02 | 4.0117E-02 | 2.4221E-02 | 2.5042E-02 | 2.9580E-02 |
| (6) | Ave | **5.2512E-02** | 6.2369E-02 | 6.5216E-02 | 6.4092E-02 | 7.5020E-02 | 6.7684E-02 | 6.6140E-02 |
| | Std | 1.3618E-02 | 1.5030E-02 | 1.5026E-02 | 1.5339E-02 | 1.8282E-02 | 1.6150E-02 | **1.2596E-02** |
| (7) | Ave | **8.2099E-05** | 1.3825E-02 | 1.0151E-02 | 3.4460E-03 | 2.6437E-02 | 3.6942E-03 | 1.0659E-02 |
| | Std | **9.8426E-05** | 2.6788E-02 | 2.4105E-02 | 1.4725E-02 | 3.3158E-02 | 1.4752E-02 | 2.4171E-02 |
| (8) | Ave | **2.4269E-02** | 2.8569E-02 | 2.6382E-02 | 2.5198E-02 | 2.5827E-02 | 2.5548E-02 | 3.0493E-02 |
| | Std | **1.8638E-03** | 4.6855E-03 | 4.3282E-03 | 5.8062E-03 | 6.1004E-03 | 2.4830E-03 | 4.4191E-03 |
| (9) | Ave | **3.0481E-03** | 2.9426E-02 | 3.8471E-02 | 1.2082E-02 | 3.5175E-02 | 2.6617E-02 | 5.0981E-02 |
| | Std | **1.2994E-02** | 2.9820E-02 | 2.8693E-02 | 2.4517E-02 | 3.4968E-02 | 2.9942E-02 | 2.1542E-02 |
| (10) | Ave | **3.1787E-02** | 4.0377E-02 | 5.1692E-02 | 5.0176E-02 | 4.7623E-02 | 4.7697E-02 | 3.9207E-02 |
| | Std | **1.2600E-02** | 1.4333E-02 | 1.6346E-02 | 1.7776E-02 | 1.6318E-02 | 1.6721E-02 | 1.3936E-02 |
| Friedman | | **1.00** | 3.60 | 5.00 | 3.60 | 6.10 | 3.90 | 4.80 |
| Rank | | **1** | 2 | 6 | 2 | 7 | 4 | 5 |

TABLE VIII. AVERAGE ACCURACY OF SSFTSO AND OTHER ALGORITHMS

| Datasets | SSFTSO | PDO | WOA | HHO | AOA | OOA | AO |
|---|---|---|---|---|---|---|---|
| (1) | **0.9964** | 0.9786 | 0.9714 | 0.9929 | 0.9679 | 0.9750 | 0.9464 |
| (2) | **0.9369** | 0.9071 | 0.9071 | 0.9000 | 0.8964 | 0.9107 | 0.9131 |
| (3) | **0.9556** | 0.9389 | 0.9278 | 0.9417 | 0.9194 | 0.9278 | 0.9389 |
| (4) | **0.9900** | 0.9800 | 0.9450 | 0.9300 | 0.9100 | 0.9550 | 0.9350 |
| (5) | **0.7455** | 0.7288 | 0.7106 | 0.7152 | 0.7061 | 0.7258 | 0.7212 |
| (6) | **0.9475** | 0.9388 | 0.9350 | 0.9363 | 0.9250 | 0.9325 | 0.9350 |
| (7) | **1.0000** | 0.9867 | 0.9900 | 0.9967 | 0.9733 | 0.9967 | 0.9900 |
| (8) | **0.9756** | 0.9723 | 0.9736 | 0.9748 | 0.9740 | 0.9744 | 0.9702 |
| (9) | **0.9971** | 0.9706 | 0.9618 | 0.9882 | 0.9647 | 0.9735 | 0.9500 |
| (10) | **0.9713** | 0.9648 | 0.9519 | 0.9537 | 0.9546 | 0.9565 | 0.9630 |
| Friedman | **7.00** | 4.50 | 2.80 | 4.35 | 1.80 | 4.30 | 3.25 |
| Rank | **1** | 2 | 6 | 4 | 7 | 3 | 5 |

TABLE IX. AVERAGE NUMBER OF SELECTED FEATURES OF SSFTSO AND OTHER ALGORITHMS

| Datasets | SSFTSO | PDO | WOA | HHO | AOA | OOA | AO |
|---|---|---|---|---|---|---|---|
| (1) | 19.60 | 291.85 | 41.50 | **5.90** | 56.10 | 61.35 | 482.35 |
| (2) | 70.65 | 161.20 | 148.90 | 237.50 | **57.05** | 139.00 | 302.40 |
| (3) | **147.30** | 563.85 | 346.35 | 526.20 | 254.60 | 377.75 | 824.80 |
| (4) | 317.55 | 445.35 | 344.45 | 274.65 | 478.05 | **239.60** | 1372.90 |
| (5) | 238.70 | 503.05 | 349.10 | 359.30 | **161.40** | 311.05 | 304.60 |
| (6) | **677.05** | 2181.60 | 1090.65 | 1234.30 | 970.50 | 1082.30 | 2255.55 |
| (7) | 44.90 | 341.65 | 137.35 | 79.85 | **20.50** | 215.60 | 414.95 |
| (8) | 84.80 | 742.40 | 127.80 | 155.90 | **35.05** | 118.25 | 664.30 |
| (9) | **303.70** | 687.45 | 1376.65 | 969.30 | 521.55 | 917.00 | 3299.80 |
| (10) | 189.05 | 311.00 | 225.80 | 243.65 | 151.85 | 258.80 | **142.50** |
| Friedman | **1.90** | 5.80 | 4.10 | 4.30 | 2.20 | 3.80 | 5.90 |
| Rank | **1** | 6 | 4 | 5 | 2 | 3 | 7 |

TABLE X. AVERAGE COMPUTATION TIME OF SSFTSO AND OTHER ALGORITHMS

| Datasets | SSFTSO | PDO | WOA | HHO | AOA | OOA | AO |
|---|---|---|---|---|---|---|---|
| (1) | **4.4701** | 148.5359 | 4.7957 | 9.0483 | 8.7623 | 8.3335 | 12.2655 |
| (2) | **4.8893** | 22.2469 | 5.3728 | 9.9205 | 8.7521 | 10.0011 | 13.4727 |
| (3) | **5.0668** | 112.0001 | 5.3806 | 10.0717 | 8.8574 | 9.7266 | 12.9080 |
| (4) | **4.9795** | 262.8809 | 5.3107 | 9.6390 | 8.9167 | 9.2710 | 13.3769 |
| (5) | **4.2250** | 7.8866 | 5.6449 | 9.7530 | 6.6386 | 9.7849 | 9.2901 |
| (6) | 11.1871 | 339.6779 | **9.7088** | 18.5324 | 26.2032 | 21.5097 | 34.8416 |
| (7) | **5.0622** | 88.3226 | 5.7162 | 10.2049 | 8.6398 | 10.1526 | 13.0427 |
| (8) | 18.7485 | 130.4110 | **9.7820** | 19.2521 | 50.2088 | 23.0154 | 72.3330 |
| (9) | 8.0227 | 724.0825 | **7.7485** | 13.1827 | 18.1501 | 14.7506 | 25.2620 |
| (10) | **4.0735** | 5.8861 | 5.3467 | 8.8967 | 6.1414 | 9.4881 | 8.3102 |
| Friedman | **1.30** | 6.30 | 1.70 | 4.50 | 3.80 | 4.60 | 5.80 |
| Rank | **1** | 7 | 2 | 4 | 3 | 5 | 6 |

From the changing trend of the curves, the convergence characteristics of the algorithms can be intuitively observed. It can be found from the convergence curve graph that, compared with other algorithms, SSFTSO can converge to a better value on all datasets. Especially on the datasets (2,3,6,8), there is a significant difference from other algorithms. The second one is HHO, which converges better on five datasets (1,3,7,8,9). The above results demonstrate the strength of SSFTSO in convergence and can achieve a better average fitness. Fig. 6 shows the ranking radar chart and heat chart of SSFTSO and other algorithms. Among them, Fig. 6(a) is the average fitness ranking radar chart, Fig. 6(b) is the average accuracy ranking radar chart, and Fig. 6(c) is the average running time heat chart. It can be seen from Fig. 6(a) and Fig. 6(b) that the area contained in SSFTSO is the smallest, which indicates that the proposed algorithm can achieve a lower average fitness and a higher average accuracy rate. As can be seen from Fig. 6(c), compared with other algorithms, SSFTSO has a great advantage in terms of time. Fig. 7 is a line chart of the selected feature ratios of SSFTSO and other algorithms. It can be seen that SSFTSO selects a smaller number of features on most datasets, which can effectively reduce the feature dimension.

## V. CONCLUSION

Since the TSO algorithm cannot continuously explore and converge to a better solution, this paper proposes a tuna optimization algorithm with sinusoidal selection factors to solve the high-dimensional FS. Compared with traditional methods, the sinusoidal selection factor strategy can adaptive select the position update strategy according to the iterative process and has dynamic adaptability. Meanwhile, the sinusoidal selection factor can also improve the convergence accuracy, enabling the algorithm to be further iterative optimized. The effectiveness of SSFTSO has been proven in IEEE CEC-2022. In addition, in order to further explore the performance of the proposed algorithm, SSFTSO is applied to the high-dimensional FS problem. In the context of high-dimensional datasets, after comparison with the other six algorithms, it can be found that SSFTSO has a great advantage in obtaining the average fitness value. It can not only ensure the classification accuracy rate but also effectively reduce the feature dimension. The experimental results prove that the proposed method can be used to attempt to solve the high-dimensional FS problem. This paper designs the enhanced version algorithm SSFTSO of TSO. For future research directions, the following forms can be considered.

1) In terms of application fields, it can be considered to be applied to specific datasets in other backgrounds to explore the performance of the proposed algorithm. For example: text data, DDOS network attacks and medical datasets, etc.

2) In terms of algorithm improvement, one can attempt to perturb the area near the optimal solution of the algorithm to further search for the optimal value.

3) In terms of the structure of FS, a new multi-objective algorithm can be attempted to be proposed to solve the FS problem.

## REFERENCES


[1] S. Thudumu, P. Branch, and J. Jin. "A comprehensive survey of anomaly detection techniques for high dimensional big data", *Journal of big data*, vol. 7, pp. 1-30, 2020.

[2] P. Ray, S. S. Reddy, and T. Banerjee. "Various dimension reduction techniques for high dimensional data analysis: a review", *Artificial Intelligence Review*, vol. 54, no. 5, pp. 3473-3515, 2021.

[3] M. Li, H. Wang, and L. Yang. "Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction", *Expert Systems with Applications*, vol. 150, pp. 113277, 2020.

[4] S. Bashir, I. U. Khattak, and A. Khan. "A novel feature selection method for classification of medical data using filters, wrappers, and embedded approaches", *Complexity*, vol. 1, pp. 8190814, 2022.

[5] E. Hancer, B. Xue, and M. Zhang. "Differential evolution for filter feature selection based on information theory and feature ranking", *Knowledge-Based Systems*, vol. 140, pp. 103-119, 2018.

[6] X. Cui, Y. Li, and J. Fan. "A novel filter feature selection algorithm based on relief", *Applied Intelligence*, vol. 52, no. 5, pp. 5063-5081, 2022.

[7] C. Freeman, D. Kulić, and O. Basir. "An evaluation of classifier-specific filter measure performance for feature selection", *Pattern Recognition*, vol. 48, no. 5, pp. 1812-1826, 2015.

[8] S. Yu, H. Chen, and A. A. Heidari. "Multi-strategy ensemble binary RIME optimization for feature selection", *Neurocomputing*, vol. 639, pp. 130284, 2025.

[9] B. Wang, J. Zhang, and H. Zhang. "AHFS-PTP: an adaptive hybrid feature selection method for pedestrian trajectory prediction", *Transportmetrica B: Transport Dynamics*, vol. 13, no. 1, pp. 2490549, 2025.

[10] M. S. Al-Batah, M. R. Al-Eiade, and Y. Alnsour. "A novel approach for enhancing plant leaf classification with the binary cuckoo search algorithm", *Applied Computational Intelligence and Soft Computing*, vol. 1, pp. 7696962, 2025.

[11] S. Maldonado, and J. López. "Dealing with high-dimensional class-imbalanced datasets: embedded feature selection for SVM classification", *Applied Soft Computing*, vol. 67, pp. 94-105, 2018.

[12] Q. Wang, C. Yue, and X. Li. "Enhancing robustness of monthly streamflow forecasting model using embedded-feature selection algorithm based on improved gray wolf optimizer", *Journal of Hydrology*, vol. 617, pp. 128995, 2023.

[13] L. Xie, T. Han, and H. Zhou. "Tuna swarm optimization: a novel swarm-based metaheuristic algorithm for global optimization", *Computational intelligence and Neuroscience*, vol. 1, pp. 9210050, 2021.

[14] Z. Yan, J. Yan, and Y. Wu. "A novel reinforcement learning based tuna swarm optimization algorithm for autonomous underwater vehicle path planning", *Mathematics and Computers in Simulation*, vol. 209, pp. 55-86, 2023.

[15] W. Wang, and J. Tian. "An improved nonlinear tuna swarm optimization algorithm based on circle chaos map and levy flight operator", *Electronics*, vol. 11, no. 22, pp. 3678, 2022.

[16] C. Sun, Q. Shao, and Z. Zhou. "An enhanced FCM clustering method based on multi-strategy tuna swarm optimization", *Mathematics*, vol. 12, no. 3, pp. 453, 2024.

[17] C. Fan, W. Wang, and J. Tian. "Flexible job shop scheduling with stochastic machine breakdowns by an improved tuna swarm optimization algorithm", *Journal of manufacturing systems*, vol. 74, pp. 180-197, 2024.

[18] M. H. Hassan, and S. Kamel. "A hybrid approach based on tuna swarm and equilibrium optimizers for parameter estimation of different PV models", *Cluster Computing*, vol. 28, no. 4, pp. 243, 2025.

[19] A. E. Ezugwu, J. O. Agushaka, and L. Abualigah. "Prairie dog optimization algorithm", *Neural Computing and Applications*, vol. 34, no. 22, pp. 20017-20065, 2022.

[20] S. Mirjalili, and A. Lewis. "The whale optimization algorithm", *Advances in engineering software*, vol. 95, pp. 51-67, 2016.

[21] A. A. Heidari, S. Mirjalili, and H. Faris. "Harris hawks optimization: algorithm and applications", *Future Generation Computer Systems*, vol. 97, pp. 849-872, 2019.

[22] L. Abualigah, A. Diabat, and S. Mirjalili. "The arithmetic optimization algorithm", *Computer Methods in Applied Mechanics and Engineering*, vol. 376, pp. 113609, 2021.

[23] M. Dehghani, and P. Trojovský. "Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems", *Frontiers in Mechanical Engineering*, vol. 8, pp. 1126450, 2023.

[24] L. Abualigah, D. Yousri, and M. Abd Elaziz. "Aquila optimizer: a novel meta-heuristic optimization algorithm", *Computers & Industrial Engineering*, vol. 157, pp. 107250, 2021.

[25] Y. C. Wang, H. M. Song, and J. S. Wang. "GOG-MBSHO: multi-strategy fusion binary sea-horse optimizer with Gaussian transfer function for feature selection of cancer gene expression data", *Artificial Intelligence Review*, vol. 57, no. 12, pp. 347, 2024.

[26] M. Mafarja, I. Aljarah, and A. A. Heidari. "Binary dragonfly optimization for feature selection using time-varying transfer functions", *Knowledge-Based Systems*, vol. 161, pp. 185-204, 2018.