Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.051336

ARTICLE

Wild Gibbon Optimization Algorithm

Jia Guo[1,2,4,6], Jin Wang[2], Ke Yan[3], Qiankun Zuo[1,2,4,*], Ruiheng Li[1,2,4], Zhou He[1,2,4], Dong Wang[5] and Yuji Sato[6]

[1]Hubei Key Laboratory of Digital Finance Innovation, Hubei University of Economics, Wuhan, 430205, China

[2]School of Information Engineering, Hubei University of Economics, Wuhan, 430205, China

[3]China Construction Third Engineering Bureau Installation Engineering Co., Ltd., Wuhan, 430074, China

[4]Hubei Internet Finance Information Engineering Technology Research Center, Hubei University of Economics, Wuhan, 430205, China

[5]College of Informatics, Huazhong Agricultural University, Wuhan, 430070, China

[6]Faculty of Computer and Information Sciences, Hosei University, Tokyo, 184-8584, Japan

*Corresponding Author: Qiankun Zuo. Email: qk.zuo@hbue.edu.cn

Received: 02 March 2024 Accepted: 07 June 2024 Published: 18 July 2024

ABSTRACT

Complex optimization problems hold broad significance across numerous fields and applications. However, as the dimensionality of such problems increases, issues like the curse of dimensionality and local optima trapping also arise. To address these challenges, this paper proposes a novel Wild Gibbon Optimization Algorithm (WGOA) based on an analysis of wild gibbon population behavior. WGOA comprises two strategies: community search and community competition. The community search strategy facilitates information exchange between two gibbon families, generating multiple candidate solutions to enhance algorithm diversity. Meanwhile, the community competition strategy reselects leaders for the population after each iteration, thus enhancing algorithm precision. To assess the algorithm's performance, CEC2017 and CEC2022 are chosen as test functions. In the CEC2017 test suite, WGOA secures first place in 10 functions. In the CEC2022 benchmark functions, WGOA obtained the first rank in 5 functions. The ultimate experimental findings demonstrate that the Wild Gibbon Optimization Algorithm outperforms others in tested functions. This underscores the strong robustness and stability of the gibbon algorithm in tackling complex single-objective optimization problems.

KEYWORDS

Complex optimization; wild gibbon optimization algorithm; community search; community competition

1 Introduction

In recent years, the field of optimization has emerged a surge of complex problems, including those with non-linear dynamics [1], intricate constraints [2], and notably, high dimensions. This evolving landscape has brought optimization algorithms into the limelight of scholarly research. The increasing dimensions in various domains such as resource allocation, scheduling, and network design, have catalyzed the emergence of high-dimensional optimization challenges. These challenges are marked by an array of variables, which significantly complicates both analysis and solution process. Compared with other problems, a distinctive aspect of high-dimensional optimization problems is the exponential increase in the search space proportional to the number of decision variables involved.

Metaheuristic algorithms have their roots in the 1960s and 1970s, a period marked by the exploration of heuristic methods derived from observing natural phenomena and human decision-making to tackle complex optimization challenges. These innovative approaches, inspired by the mechanisms of biological evolution, collective behavior of species, and physical processes, were developed to simulate the exploration and learning behaviors found in nature. The aim was to navigate the vast solution spaces of optimization problems efficiently, seeking out optimal or sufficiently good solutions. These algorithms have found widespread application across various fields, notably in engineering design, scheduling problems, network design, and machine learning, where they facilitate resource arrangement, network configuration optimization, and learning model enhancement. The application of metaheuristic algorithms spans a wide array of fields like segmentation [3,4] feature processing [5], vision transformer [6], and multiple-disease detection [7].

Despite their versatility and demonstrated efficacy, metaheuristic algorithms are not without limitations. Challenges include ensuring convergence to global optima in high-dimensional and complex problem spaces, the complexity and sensitivity of algorithm parameters, which can significantly affect performance, and the substantial computational resources required for large-scale problems. Additionally, the lack of a rigorous theoretical foundation for some metaheuristic approaches makes their behavior and efficiency difficult to predict accurately. Nevertheless, the adaptability and success of metaheuristic algorithms in practical applications continue to support their widespread use and ongoing development in the field of optimization, highlighting the balance between their potential and limitations.

To address these challenges, this paper proposes a novel Wild Gibbon Optimization Algorithm (WGOA) based on an analysis of the behavior of wild gibbon populations. The WGOA simplifies the family structure of gibbons to consist of a female, a male, and an offspring, utilizing particles without mass and volume to simulate the behavioral patterns of gibbon populations during foraging and territory acquisition. This search pattern enhances the algorithm's ability to provide high-precision solutions for high-dimensional optimization problems. The main contributions of the algorithm include the following main points:

- A community search strategy is proposed in this work. This strategy involves leading the particle swarm with particles that have male gibbon identities to perform global searches. This approach helps in determining the personal optima. This strategy is proposed and developed by the authors to enhance the global search capability of the algorithm.
- A community competition strategy is proposed in this work. This strategy entails the global optimal group competing with the positions explored by the male gibbons to ensure the superiority of the group's position. This method is introduced by the authors to improve the algorithm's convergence towards the global optimum.
- The combination of the community search strategy and community competition strategy enables the algorithm to conduct broad-scale searches in the initial phases and precise searches in later stages. This integration significantly enhances both the diversity and accuracy of the solution set, ensuring robust and reliable results.

The remainder of this thesis will explain this algorithm in detail. Section 2 introduces related work of this study. Section 3 describes the details of the algorithm. Section 4 shows the contents of the simulations and the results of the experiments. Section 5 summarizes the content of this study and future work.

2 Related work

Metaheuristic algorithms [8–10] are high-level heuristic methods used to find approximate solutions to complex optimization problems. These methods are often inspired by natural phenomena, biological evolution, and behaviors in different species [11–13]. Metaheuristic algorithms have shown great performance in various applications, such as image segmentation [14–16], path planning for mobile robots [17,18], and manufacturing energy optimization problems [19–21]. In recent years, many researchers have developed new metaheuristic algorithms to address these optimization challenges. Su et al. [22] introduced the RIME optimization algorithm (RIME) inspired by the physical phenomenon of rime-ice. The RIME algorithm implements the exploration and exploitation behaviors in the optimization methods by simulating the soft-rime and hard-rime growth process of rime-ice. Song et al. [23] proposed the Phasmatodea population evolution (PPE) algorithm based on the evolution characteristics of the Phasmatodea population. Połap et al. [24] proposed a red fox optimizer (ROF), searching for food, hunting, and developing population while escaping from hunters. These meta-heuristic algorithms have shown good optimization performance in many real-world optimization problems. However, with the further study of these algorithms in the application fields, some researchers have raised some problems of these original meta-heuristic algorithms such as premature convergence [25] and slow convergence speed [26].

To overcome these problems, many scholars have optimized these original algorithms from different aspects. Some scholars have applied the method of adopting new search strategies. Li et al. [27] developed an adaptive particle swarm optimization with decoupled exploration and exploitation (APSO-DEE) to effectively balance exploration and exploitation. The APSO-DEE adopted two novel learning strategies, a local sparseness degree measurement in fitness landscape and an adaptive multi-swarm strategy. Zhang et al. [28] developed a charging safety early warning model for electric vehicles based on the improved grey wolf optimization (IGWO) algorithm. The IGWO improves the classic grey wolf optimization (GWO) algorithm by utilizing a search method based on clustering search results, which can boost wolf diversity and avoid slipping into the local optimum owing to insufficient information exchange among wolves. In the context of optimizing deep learning and evolutionary algorithms, meta-heuristic methods have been integrated with neural networks [29–31] and applied to various optimization problems [32–34]. Tang et al. [35] also proposed a whale optimization algorithm combined with an artificial bee colony (ACWOA) to address the problems of slow convergence and the tendency to fall into local optima in the classic whale optimization algorithm (WOA). Thawkar et al. [36] presented a butterfly optimization algorithm and ant lion optimizer (BOAALO) for breast cancer prediction. The experimental results show that BOAALO outperforms the original butterfly optimization algorithm (BOA) and ant lion optimizer (ALO) in terms of each given statistical measure. Deb et al. [37] proposed a novel meta-heuristic considering the hybridization of chicken swarm optimization (CSO) with ant lion optimization (ALO) for effectively and efficiently coping with the charger placement problem. The amalgamation of CSO with ALO can enhance the performance of ALO, thereby preventing it from getting stuck in the local optima. Emam et al. [38] proposed a modified Reptile search algorithm (mRSA), which combines the RSA algorithm with the Runge kutta optimizer (RUN). The mRSA mitigates the limitations of RSA, which include being stuck in local optima areas and having an insufficient balance between exploitation and exploration.

Despite several improvement measures taken to enhance the effectiveness of meta-heuristic optimization algorithm in solving practical problems, challenges persist in dealing with high-dimensional problems. High-dimensional optimization problems usually involve large-scale sets of parameters or variables, leading to a significant increase in the dimensionality of the solution space. In high-dimensional space, the number of local optimal solutions increases as the solution space expands. This is due to the fact that the solution space of high-dimensional problems is more extensive and the local optimal solutions have more diverse forms in each dimension. The existence of more local optimal solutions in the solution space increases the possibility that the algorithms stay in local optimal solutions during the search process. As a result, the algorithm is more likely to fall into local optima, which in turn increases the difficulty of the problems. Researchers have developed a number of algorithms specifically for high-dimensional optimization problems to address these challenges. MiarNaeimi et al. [39] proposed the Horse herd optimization algorithm (HOA) inspired by the herding behavior of horses for high-dimensional optimization problems. HOA performs well in solving high-dimensional complex problems, due to the large number of control parameters it extracts from the behavior of horses of different ages. Ma et al. [40] proposed a two-stage hybrid ACO for high-dimensional feature selection (TSHFS-ACO). These optimization algorithms can alleviate some of the challenges associated with high-dimensional optimization problems, but there are still limitations. When solving high-dimensional problems, although the two-stage strategy of the TSHFS-ACO can alleviate the algorithm from getting into local optima, there still exist errors in feature selection, which may result in the removal of potentially useful features on the test set, thereby impacting the performance of this model. The HOA has a strong ability to balance the two phases of exploration and exploitation, but it suffers from premature convergence, which limits its effectiveness in finding optimal solutions. Therefore, a novel algorithm based on gibbon territory contention behavior is proposed to solve high-dimensional problems, called the gibbon algorithm.

Zhong et al. [41] introduced the Beluga Whale Optimization (BWO) algorithm, inspired by beluga whale behaviors, which effectively solves optimization problems through self-adaptive mechanisms, Levy flight integration, and competitive performance compared to 15 other metaheuristic algorithms, validated across various benchmark functions and real-world engineering problems. Inspired by African vultures' foraging and navigation behaviors, Abdollahzadeh et al. [42] introduced the African Vultures Optimization Algorithm (AVOA), demonstrating superior performance in solving optimization problems compared to existing algorithms across various benchmark functions and engineering design problems, supported by statistical evaluation. In addition, to solve high dimensional complex optimization problems, Harris Hawks Optimization (HHO) [43], Dung Beetle Optimizer (DBO) [44], Whale Optimization Algorithm (WOA) [45], Runge Kutta Optimizer (RUN) [46], Krill Herd (KH) [47], Artificial Gorilla Troops Optimizer (GTO) [48], Novel Hippo Swarm Optimization (NHSO) [49] are also proposed.

3 Materials and Methods

3.1 Behavior of Gibbon

The gibbon is one of the primates belonging to the family Hominidae. They are mainly found in the tropical rainforests of South East Asia, including Thailand, Indonesia, Malaysia and Southern China. They are social animals used to living in groups. As such, they generally form family units and colonies within their populations, maintaining the structure and order of the group through social interaction. Family units generally consist of a pair of adult males and females and their minor children. Within a family unit, there may be multiple adult males with certain social hierarchies and competitive relationships, and it is usually the alpha male that dominates the decision-making and defense of the family unit. In order to compete for food resources and habitat environments, they establish their own territories in the forest and defend them. The size of the territory depends on the availability of resources and the number of family units. The quality of territory has a direct impact on gibbon reproduction and survival, thus a particular kind of territorial competition is present in gibbon populations. Family units mark and claim territories through musical calls, which are unique to their species and an important form of communication between them. When competing for territory, the alpha male will roam to determine the distribution of resources and environmental conditions in different territories. This means moving quickly between trees by swinging his long arms to gain a territorial view. In a given area, multiple family units may form a larger group, called a "colony". There may be some degree of competition or cooperation between the various family units in the herd.

3.2 Wild Gibbon Optimization Algorithm

Inspired by the competition and survival patterns of gibbon populations, the behavior of gibbons was simulated, and the Wild Gibbon Optimization algorithm (WGOA) was designed, which relates the superiority and inferiority process resulting from gibbon population competition to an optimized objective function. During the simulation, it is assumed that the family contains three members: male gibbons, female gibbons, and child gibbons. The physiological factors such as mass and volume of the members are ignored, and they are represented as a point in space, respectively. The value of a gibbon's preference for a territory depends on the food abundance, and habitat superiority of that territory, which is related to the optimal value of the objective function. The territorial replacement behavior of gibbons arising from population competition is then modeled as a process of solution updating.

In the gibbon family, the male gibbon is considered to be the leader of social behaviors such as territorial conservation and population competition. Male gibbons evaluate their environment by roaming in trees to facilitate locating a superior survival environment. This assessment process is modeled as a search process for solutions and mainly works on local search. Female gibbon is able to maintain the stability of family relationships and therefore can keep the balance during particle updating. Child gibbon can achieve roaming over a larger area of the family territory and can improve the global property in the particle search process. After finding a superior survival environment, male gibbons lead their families to migrate, allowing both male and child gibbons to live around the superior environment.

In simulation experiments, particles without weight and volume are used to simulate the behavior of gibbons. Each particle is endowed with three layers of memory, with the first layer representing male gibbons, the second layer representing female gibbons, and the third layer representing juvenile gibbons. Each particle participates in evolution as a collective entity of a gibbon family. Based on the environmental assessment process and family migration process of the gibbon population, two search strategies are proposed: the community search strategy and the community competition strategy. The cooperation of these two strategies can balance the local search while expanding the search scope, thus avoiding falling into local optimum and improving the search accuracy. The pseudo-code for the basic steps of WGOA is shown in Algorithm 1. The flowchart of WGOA is shown in Fig. 1.

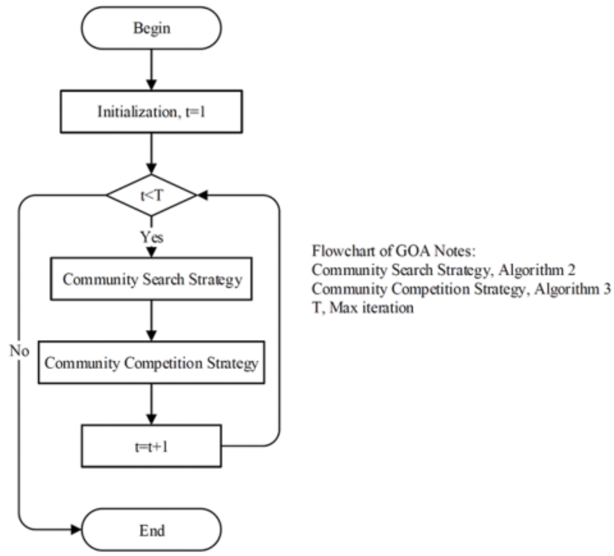

Figure 1: The flowchart of the WGOA

Algorithm 1: Wild Gibbon Optimization Algorithm

Require: Fitness function, F
Require: Maximum number of evolution, T
Require: Gibbons, $Gib = Gib(1), Gib(2), ..., (n)$
Require: Leader of gibbons, $BestGib$
Require: Position of Personal historical best territories, $Pbest - history$
Require: Position of Global best historical house, $Gbest\ history$

```
mm = 3
Each Gib contains mm memory layers
For mm = 1, 3
Calculate the positions with F
Record the results as Gib(1) (mm), Gib(2) (mm), ..., and Gib(n) (mm)
End For
Sort the memories of each particle.
Best positions place in the first layer
Worst positions place in the last layer
t = 1
while t < T do
    Do the Community search strategy
    Do the Community competition strategy
    t = t + 1
end while
Choose the best from Gib
Output BestGib
```

3.3 Community Search Strategy

In the gibbon family, female and child gibbons follow the male gibbon, leading to the elimination of poorer territories due to the dominant role of male gibbons in territory exploration. In the experiments, particles without weight and volume are used to simulate the behavior of gibbons. Within the community search strategy, each particle with three memory layers simulates a gibbon family. Among them, the first layer of memory records the best position, representing male gibbons; the second layer of memory records the second-best position, representing female gibbons; the third layer of memory records the third-best position, representing juvenile gibbons.

In each iteration, the group is searched by the leader particle according to Eq. (1). If the leader particle discovers a better position, the group particles are updated to ensure the superiority of the group's position. Otherwise, the group particles remain unchanged. This process is completed through Eq. (2). At the same time, the other two particles in the group, represented by the female gibbon and the child gibbon, work to maintain a balance between global and local searches.

$$\alpha = \frac{Gib\,(ii)^{t}\,(kk) + BestGib^{t}\,(ll)}{2}$$
$$\beta = \left|Gib\,(ii)^{t}\,(kk) + BestGib^{t}\,(ll)\right|$$
$$GibCandidate(ii)^{t+1}(kk, ll) = GD(\alpha, \beta) \tag{1}$$

where $Gib(ii)^{t}$ is the position of the iith gibbon in the tth iteration, $BestGib^{t}$ is the best gibbon in the tth iteration, $kk = [1, 2, 3]$ and $ll = [1, 2, 3]$ stand for the memory layer of the Gib and the $BestGib$, $GibCandidate(ii)$ is the new position that explored by this iteration, and $GD()$ is Gaussian distribution with a mean α and a standard deviation β.

$$Gib(ii)^{t+1} = Best\left(3, (GibCandidate(ii)^{t+1}, Gib\left(ii\right)^{t+1}\right)\right) \tag{2}$$

where $Gib(ii)^{t+1}$ is the position of the iith gibbon in the $(t+1)$th iteration. $Best\,(3, array)$ is a function used to find the best 3 position form the input $array$.

The pseudo-code of community search strategy is shown in Algorithm 2. Under this strategy, the information of the particles will be fully utilized, making the global search more efficient.

Algorithm 2: Community Search Strategy

Require: Number of gibbons n
Require: Gibbons, $Gib = Gib(1), Gib(2), \ldots, (n)$
Require: Leader of gibbons, $BestGib$
1: $ii = 1$
2: **while** $ii < n$ **do**
3: Calculate the candidate positions of $Gib\ (ii)$ by Eq. (1)
4: Update position of $Gib\ (ii)$ in the next iteration with Eq. (2)
5: $ii = ii + 1$
6: **end while**

3.4 Community Competition Strategy

Male gibbons have a stronger fighting ability, allowing them to compete with other families for superior territories. After this competition, the victor will occupy the best territory known so far. According to this modal, in the community competition strategy, the position of the global optimal particle is updated. This strategy maintains three historical best positions to enhance local search. The number of these historical positions matches the number in the particle group of the community search strategy. The global optimal particle group is updated when the leader particle searches for a preferred location. Specifically, the new position found by the leader particle is compared with the current global optimal group positions. If the new position is superior, the global optimal group is updated, thereby ensuring favorable conditions for the group's position. Following this, the global best particle is updated using Eq. (3).

$$BestGibCandi^{t+1} = Best\left(Gib\,(1)^t\,(1)\,, Gib\,(2)^t\,(1)\,, \ldots, Gib\,(n)^t\,(1)\right)$$

$$BestGib^{t+1} = Best(BestGibCandi^{t+1}, BestGib^t) \tag{3}$$

where $BestGibCandi(ii)^{t+1}$ is the candidate position of the best gibbons in the $(t+1)$th iteration, $Gib(ii)^t(1)$ is the best memory of the iith gibbon in the tth iteration, $BestGib^{t+1}$ is the position of the best gibbons in the $(t+1)$th iteration. $Best\ (array)$ is a function used to find the best position form the input $array$.

The pseudo-code of community competition strategy is shown in Algorithm 3. This strategy improves the diversity of particles while facilitating the local exploration of particles. These two strategies can work together effectively to ensure that the global search range is increased while the search efficiency is improved. This cooperation promotes the balance of local and global search during iteration.

Algorithm 3: Community Competition Strategy

Require: Gibbons, $Gib = Gib(1), Gib(2), \ldots, (n)$
Require: Leader of gibbons, $BestGib$
Require: Number of gibbons, n
Require: Objective function, F
1: $jj = 1$
2: **while** $jj < n$ **do**
3: Calculate the position of $BestGib$ in the next iteration by Eq. (3)
4: $jj = jj + 1$
5: **end while**

This is followed by an analysis regarding the time complexity. From the pseudo-code of this WGOA, it can be seen that the algorithm mainly consists of particle swarm initialization, community search strategy and community competition strategy. For a particle swarm of size N, the time complexity of the algorithm for random initialization is $O(N)$. In both strategies, only the linear computation with a factor of three is added, which do not increase in computational complexity. Therefore, the time complexity of WGOA is equivalent to the time complexity of the particle swarm algorithm, both being $O(N)$.

4 Results

4.1 Experimental Methods

To better show the experimental results, CEC2017 and CEC2022 benchmark functions are selected as the test function. CEC2017 contains 29 test functions, which include uni-modal functions, simple multi-peaked functions, hybrid functions and composite functions. For the uni-modal functions, the algorithm has a low difficulty in finding the optimal value, and these functions have a unique global optimal value, which can show the ability of the algorithm to search globally. Simple multi-peaked functions, hybrid functions and composite functions all have multiple local optima and a unique global optimum. The algorithm is more difficult to find the superiority in these functions. The functions can test whether the algorithm can jump out of the local optimum, has high search efficiency and other comprehensive capabilities. The CEC2022, which contains 13 different benchmark functions, has been recognized for its ability to detect the effectiveness of algorithms in solving high-dimensional optimization problems. To ensure fairness, Function Evaluation Times (FES) are used to evaluate the performance of all algorithms. To show the performance of the WGOA algorithm, 9 excellent meta-heuristics are selected as a control group: Beluga Whale Optimization (BWO) [41], African Vultures Optimization Algorithm (AVOA) [42], Harris Hawks Optimization (HHO) [43], Dung Beetle Optimizer (DBO) [44], Whale Optimization Algorithm (WOA) [45], Runge Kutta Optimizer (RUN) [46], Krill Herd (KH) [47], Artificial Gorilla Troops Optimizer (GTO) [48] and RIME. To ensure equity, all algorithms were employed with their respective optimal parameters. To ensure the fairness of the experiment, the parameters in the algorithm will be consistent with the control group, where the particle population size is 100, every algorithm will stop when it reaches the max fitness-function evaluations (FES) times 100,000. Each group of experiments will be run 30 times independently to reduce the error caused by experimental chance.

4.2 Experimental Results

To compare the experimental and control groups with higher efficiency, we subtract the optimal value obtained by the experimental results from the theoretical optimal value of the function. The smaller the result obtained, the higher the accuracy of the algorithm is indicated. This result will be calculated using Eq. (4) and is denoted as PT.

$$PT = Practicaloptimum - Theoreticaloptimum \tag{4}$$

where *Practicaloptimum* stands for the results obtained by the algorithm after the final iteration, *Theoreticaloptimum* stands for the theoretical optimum solution.

The MEAN and STD of 30 sets of independent runs are recorded. Numerical results of CEC2017 are shown in Tables 1–5. Numerical results of CEC2022 are shown in Tables 6 and 7. Average rank of the CEC 2017 and CEC2020 are shown in Tables 8 and 9.

Table 1: PTs of the WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2017 F1–F6

	F1			F2		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	1.5677E+04	1.6460E+04	2	1.5117E+06	5.8701E+05	6
WOA	3.8130E+07	1.4923E+07	6	6.3278E+05	1.4586E+05	5
AVOA	**7.0409E+03**	**7.4188E+03**	1	2.0679E+29	1.1260E+30	9
HHO	2.0353E+08	2.4336E+07	8	**3.3689E+04**	**9.0611E+03**	1
DBO	7.0133E+07	5.1897E+07	7	3.1625E+05	2.2364E+04	4
BWO	2.3965E+11	5.9748E+09	10	3.0691E+05	1.7515E+04	3
RUN	3.1056E+04	1.4964E+04	3	1.2809E+19	5.4147E+19	8
KH	6.7754E+05	3.6529E+05	5	1.6835E+05	3.0296E+04	2

(Continued)

Table 1 (continued)

	F1			F2		
	Mean	Std	Rank	Mean	Std	Rank
GTO	2.7575E+10	8.1273E+09	9	2.5563E+08	1.1159E+09	7
RIME	2.2039E+05	5.2497E+04	4	7.8920E+48	4.2157E+49	10
	F3			F4		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	**1.7255E+02**	4.9743E+01	1	7.6376E+02	1.4101E+02	5
WOA	6.3948E+02	1.1579E+02	5	9.7206E+02	1.3303E+02	7
AVOA	1.9893E+04	6.9086E+03	9	2.5897E+02	4.7446E+01	2
HHO	4.4460E+02	8.8535E+01	3	9.1924E+02	5.7905E+01	6
DBO	4.7949E+02	1.3438E+02	4	1.0877E+03	1.6992E+02	8
BWO	7.7138E+04	4.2721E+03	10	1.5532E+03	**2.4414E+01**	10
RUN	6.4269E+02	2.4138E+02	6	**2.5697E+02**	4.8561E+01	1
KH	2.9256E+02	**4.7394E+01**	2	6.8078E+02	6.5403E+01	4
GTO	3.5120E+03	1.1616E+03	7	1.1205E+03	1.9394E+02	9
RIME	1.7095E+04	3.6351E+03	8	2.7812E+02	3.8458E+01	3
	F5			F6		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	**3.3550E+01**	8.9035E+00	1	8.0839E+02	1.4168E+02	4
WOA	7.9474E+01	1.0534E+01	5	2.5941E+03	1.5901E+02	8
AVOA	7.8227E+02	5.9745E+01	9	4.2041E+01	4.5169E+00	2
HHO	7.5938E+01	2.8957E+00	4	2.7393E+03	1.1084E+02	9
DBO	7.1517E+01	8.2798E+00	3	1.5373E+03	4.0682E+02	6
BWO	1.0769E+02	**2.1711E+00**	7	3.0502E+03	4.0082E+01	10
RUN	8.1528E+02	4.1280E+01	10	5.9126E+01	2.3011E+00	3
KH	5.9920E+01	3.4014E+00	2	1.3684E+03	1.2743E+02	5
GTO	8.7430E+01	1.2545E+01	6	2.1498E+03	2.6910E+02	7
RIME	3.4981E+02	5.4910E+01	8	**3.5690E+00**	**1.3945E+00**	1

Table 2: PTs of the WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2017 F7–F12

	F7			F8		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	6.9690E+02	1.2280E+02	2	3.1028E+04	9.9944E+03	6
WOA	1.0806E+03	1.1978E+02	5	3.4413E+04	7.5232E+03	7
AVOA	2.1387E+03	1.6353E+02	10	8.8201E+02	8.4873E+01	2
HHO	1.0479E+03	6.5269E+01	4	2.8504E+04	2.8439E+03	5
DBO	1.2171E+03	1.3817E+02	7	3.5978E+04	9.9184E+03	8
BWO	1.6927E+03	**4.2342E+01**	8	6.9486E+04	4.2354E+03	10

(Continued)

Table 2 (continued)

	F7			F8		
	Mean	Std	Rank	Mean	Std	Rank
RUN	1.8655E+03	2.6186E+02	9	8.8937E+02	7.9785E+01	3
KH	7.7637E+02	7.0616E+01	3	2.4353E+04	2.0390E+03	4
GTO	1.0811E+03	2.2076E+02	6	6.6399E+04	1.9632E+04	9
RIME	**4.4024E+02**	7.0102E+01	1	**3.6914E+02**	**6.0521E+01**	1
	F9			F10		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	2.1368E+04	8.3580E+03	7	2.9221E+03	1.6627E+03	2
WOA	1.9977E+04	2.7157E+03	5	6.1770E+03	1.9767E+03	3
AVOA	2.1459E+04	1.2188E+03	8	1.5063E+04	1.5755E+03	7
HHO	1.7961E+04	1.4556E+03	4	**1.9665E+03**	**2.0934E+02**	1
DBO	1.6820E+04	1.2688E+03	3	1.3515E+04	1.5218E+04	4
BWO	2.9499E+04	**6.3392E+02**	9	1.8062E+05	1.7467E+04	9
RUN	2.0892E+04	1.0554E+03	6	1.4015E+04	1.6551E+03	5
KH	1.4499E+04	1.0657E+03	2	1.9329E+04	8.0750E+03	8
GTO	3.0969E+04	3.7263E+03	10	2.3825E+05	1.5330E+05	10
RIME	**3.9488E+03**	4.5068E+03	1	1.4154E+04	1.2409E+03	6
	F11			F12		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	6.1830E+07	3.3193E+07	5	**4.9692E+03**	**5.3134E+03**	1
WOA	6.2045E+08	2.2725E+08	8	5.1688E+05	2.3925E+06	3
AVOA	1.2659E+03	2.5884E+02	2	1.0450E+07	4.0807E+06	8
HHO	2.9424E+08	9.7373E+07	6	3.3110E+06	7.5765E+05	4
DBO	4.3304E+08	2.4942E+08	7	5.7140E+06	7.3043E+06	6
BWO	1.6016E+11	1.1024E+10	10	3.4787E+10	2.4279E+09	10
RUN	**8.4926E+02**	**9.9614E+01**	1	7.7047E+06	2.2883E+06	7
KH	1.7966E+07	7.4078E+06	4	2.5157E+04	5.3910E+03	2
GTO	1.8062E+09	9.9299E+08	9	4.6604E+06	5.8679E+06	5
RIME	1.7574E+03	3.6115E+02	3	1.3046E+08	4.9179E+07	9

Table 3: PTs of the WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2017 F13–F18

	F13			F14		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	1.1175E+06	5.8295E+05	6	**3.2521E+03**	3.4868E+03	1
WOA	1.8007E+06	8.1298E+05	7	1.3555E+05	2.9979E+05	5
AVOA	3.9292E+04	1.5474E+04	2	2.4554E+05	1.1736E+05	6
HHO	8.4966E+05	2.6113E+05	4	8.1435E+05	3.1016E+05	7

(Continued)

Table 3 (continued)

	F13			F14		
	Mean	Std	Rank	Mean	Std	Rank
DBO	3.9007E+06	3.7220E+06	9	1.5755E+06	3.6424E+06	9
BWO	3.5458E+07	1.1002E+07	10	1.5787E+10	2.1423E+09	10
RUN	**2.8022E+04**	**1.1870E+04**	1	2.1031E+04	5.8728E+03	3
KH	8.7140E+05	3.4445E+05	5	7.1500E+03	**2.4382E+03**	2
GTO	2.2201E+06	1.5461E+06	8	5.2315E+04	3.6952E+04	4
RIME	9.5127E+04	4.1220E+04	3	8.4374E+05	4.3763E+05	8
	F15			F16		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	4.9185E+03	8.2894E+02	2	3.8093E+03	7.1820E+02	2
WOA	7.7092E+03	1.3522E+03	6	5.2774E+03	9.1474E+02	7
AVOA	2.4926E+04	1.2295E+04	9	4.8914E+03	7.4908E+02	6
HHO	5.2354E+03	7.2397E+02	3	4.3198E+03	7.4277E+02	5
DBO	6.0135E+03	8.6341E+02	4	5.4550E+03	8.7846E+02	8
BWO	1.6842E+04	1.0233E+03	7	8.4532E+05	3.7918E+05	10
RUN	1.8187E+04	2.3760E+03	8	4.0636E+03	**4.8366E+02**	3
KH	**4.2599E+03**	**5.2187E+02**	1	**3.5548E+03**	5.2781E+02	1
GTO	6.9138E+03	2.1154E+03	5	5.4699E+03	1.3644E+03	9
RIME	3.7635E+04	1.5407E+04	10	4.0727E+03	6.2767E+02	4
	F17			F18		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	4.7528E+06	3.7204E+06	8	7.9838E+03	7.8596E+03	2
WOA	2.0015E+06	8.1858E+05	5	1.3487E+07	5.6442E+06	9
AVOA	4.0356E+03	7.0556E+02	3	3.6792E+05	1.2191E+05	5
HHO	2.0541E+06	9.1889E+05	6	3.6142E+06	1.5835E+06	8
DBO	6.5477E+06	5.3686E+06	9	2.1052E+06	2.1435E+06	7
BWO	8.3552E+07	2.0975E+07	10	1.5805E+10	1.7126E+09	10
RUN	3.5319E+03	5.9796E+02	2	6.4992E+04	1.1775E+04	3
KH	1.2013E+06	4.3295E+05	4	**4.7389E+03**	**3.0637E+03**	1
GTO	3.1042E+06	3.3526E+06	7	3.3295E+05	2.8573E+05	4
RIME	**3.2502E+03**	**4.5328E+02**	1	1.5933E+06	5.9653E+05	6

Table 4: PTs of the WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2017 F19–F24

	F19			F20		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	**2.9705E+03**	5.6892E+02	1	**9.4915E+02**	1.1106E+02	1
WOA	4.4757E+03	5.2130E+02	5	1.8251E+03	2.1791E+02	6

(Continued)

Table 4 (continued)

	F19			F20		
	Mean	Std	Rank	Mean	Std	Rank
AVOA	8.3904E+03	4.8741E+03	8	3.8013E+03	7.4846E+02	10
HHO	3.6570E+03	4.4489E+02	3	1.6809E+03	1.5178E+02	5
DBO	3.8992E+03	6.0504E+02	4	1.5055E+03	1.6818E+02	4
BWO	5.1118E+03	**1.5528E+02**	6	2.3934E+03	**7.6848E+01**	7
RUN	1.6872E+05	1.1367E+05	10	2.6502E+03	4.4172E+02	8
KH	3.3372E+03	5.3436E+02	2	1.1991E+03	1.2050E+02	2
GTO	5.8085E+03	9.1755E+02	7	1.3624E+03	2.3766E+02	3
RIME	3.2574E+04	2.0310E+04	9	2.9207E+03	4.3967E+02	9
	F21			F22		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	2.1945E+04	8.0321E+03	8	**1.2019E+03**	**9.3203E+01**	1
WOA	2.1942E+04	2.3233E+03	7	2.5253E+03	2.0821E+02	6
AVOA	1.2277E+03	1.3200E+02	3	1.6835E+04	1.3892E+03	10
HHO	2.0294E+04	1.5673E+03	6	2.1995E+03	1.5807E+02	4
DBO	1.8413E+04	1.1721E+03	5	1.8726E+03	2.0519E+02	2
BWO	3.1303E+04	4.4623E+02	9	3.3068E+03	1.3140E+02	7
RUN	8.4620E+02	1.0216E+02	2	1.6506E+04	1.5905E+03	9
KH	1.7747E+04	1.2882E+03	4	2.2996E+03	2.1457E+02	5
GTO	3.2379E+04	3.9019E+03	10	1.8948E+03	2.2716E+02	3
RIME	**5.8369E+02**	**6.5612E+01**	1	1.5244E+04	1.1363E+03	8
	F23			F24		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	1.7227E+03	1.3124E+02	4	**7.9099E+02**	**5.7055E+01**	1
WOA	3.4818E+03	4.1824E+02	9	1.1266E+03	5.9195E+01	4
AVOA	1.4453E+03	1.4074E+02	3	2.4078E+03	2.4322E+02	8
HHO	3.1410E+03	3.0699E+02	7	1.0178E+03	7.4656E+01	3
DBO	2.5953E+03	3.5158E+02	6	1.3647E+03	1.3416E+03	6
BWO	5.4920E+03	2.9178E+02	10	2.1805E+04	9.4738E+02	10
RUN	9.7780E+02	6.8252E+01	2	1.5397E+03	1.1323E+02	7
KH	3.2856E+03	4.5343E+02	8	8.1371E+02	5.8881E+01	2
GTO	2.2856E+03	3.0515E+02	5	3.3629E+03	5.6783E+02	9
RIME	**8.1075E+02**	**5.2710E+01**	1	1.2438E+03	6.5469E+01	5

Table 5: PTs of the WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2017 F25–F29

	F25			F26		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	1.2850E+04	1.9719E+03	4	**5.0002E+02**	**5.1864E-04**	1
WOA	2.8387E+04	3.8823E+03	9	2.6027E+03	7.2273E+02	5
AVOA	8.1493E+02	4.4454E+01	2	1.8566E+04	2.1450E+03	9
HHO	2.1292E+04	1.6191E+03	7	1.5970E+03	3.1080E+02	4
DBO	1.6757E+04	3.7584E+03	5	1.2052E+03	2.0344E+02	2
BWO	4.4898E+04	1.2044E+03	10	7.4614E+03	6.3316E+02	8
RUN	**8.1361E+02**	**4.3220E+01**	1	2.1394E+04	5.6354E+03	10
KH	2.2457E+04	1.1833E+03	8	3.0045E+03	6.8529E+02	6
GTO	1.8719E+04	2.5344E+03	6	1.2209E+03	1.8820E+02	3
RIME	8.4668E+02	6.5230E+01	3	6.9515E+03	5.1394E+02	7
	F27			F28		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	**5.0002E+02**	**4.7272E-04**	1	3.7622E+03	6.0296E+02	4
WOA	8.8104E+02	5.7330E+01	5	1.1264E+04	1.8351E+03	9
AVOA	1.2792E+03	2.0989E+02	7	5.6797E+02	**3.2048E+01**	2
HHO	7.6602E+02	4.6493E+01	3	5.7401E+03	5.9383E+02	6
DBO	1.4117E+04	6.0294E+03	9	6.4891E+03	1.2113E+03	7
BWO	2.2864E+04	5.9721E+02	10	1.3306E+05	4.7235E+04	10
RUN	1.2784E+03	1.6283E+02	6	**5.6291E+02**	4.2581E+01	1
KH	6.4636E+02	4.1165E+01	2	5.4926E+03	6.6055E+02	5
GTO	4.3238E+03	9.7466E+02	8	8.4484E+03	2.0045E+03	8
RIME	7.8421E+02	3.9729E+01	4	6.5587E+02	4.8622E+01	3
	F29					
	Mean	Std	Rank			
WGOA	1.5855E+04	3.2028E+04	4			
WOA	2.0027E+08	9.8424E+07	9			
AVOA	4.8590E+03	**5.0543E+02**	2			
HHO	2.1301E+07	4.9017E+06	6			
DBO	2.2893E+07	2.3374E+07	7			
BWO	2.9081E+10	3.3400E+09	10			
RUN	5.8021E+03	7.4118E+02	3			
KH	1.2999E+06	5.3681E+05	5			
GTO	5.1322E+07	3.8151E+07	8			
RIME	**3.7331E+03**	5.7162E+02	1			

Table 6: PTs of the WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2022 F1–F6

	F1			F2		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	**5.1159E-14**	**2.2884E-14**	1	**1.3901E+01**	1.4514E+01	1
WOA	1.4858E+01	1.5330E+01	7	5.6630E+01	1.5234E+01	7
AVOA	2.5958E-13	9.8663E-14	2	2.6864E+01	2.5043E+01	2
HHO	1.9476E+00	7.7307E-01	5	5.5207E+01	1.8025E+01	5
DBO	1.8089E+01	3.1546E+01	8	5.9187E+01	2.4143E+01	9
BWO	7.4803E+03	2.0712E+03	9	5.8559E+01	**9.9755E+00**	8
RUN	2.6542E-04	2.9923E-04	3	3.0282E+01	2.3708E+01	3
KH	1.1759E+01	5.9322E+01	6	5.5283E+01	1.5073E+01	6
GTO	3.6622E+04	3.3693E+04	10	6.4862E+01	1.4499E+01	10
RIME	2.8662E-03	1.2520E-03	4	5.1721E+01	1.2818E+01	4
	F3			F4		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	**3.6394E-03**	1.5352E-02	1	5.1539E+01	1.5281E+01	2
WOA	5.4440E+01	1.3638E+01	10	1.0718E+02	3.3928E+01	10
AVOA	1.1188E+01	6.3203E+00	4	8.8847E+01	2.8227E+01	7
HHO	4.0496E+01	1.0076E+01	9	8.2735E+01	1.5019E+01	6
DBO	1.2562E+01	5.8152E+00	5	9.3127E+01	2.3445E+01	8
BWO	1.9513E+00	2.0463E-01	3	6.9310E+01	1.5179E+01	4
RUN	2.0894E+01	9.3235E+00	6	7.9442E+01	1.0956E+01	5
KH	2.7576E+01	8.9493E+00	7	5.4140E+01	1.3360E+01	3
GTO	3.4113E+01	1.4208E+01	8	9.5114E+01	2.8326E+01	9
RIME	2.1920E-02	**1.1610E-02**	2	**4.0921E+01**	**9.6251E+00**	1
	F5			F6		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	1.4787E+01	3.3232E+01	2	7.2065E+03	1.4311E+04	6
WOA	2.2439E+03	1.1357E+03	10	4.0025E+03	5.1540E+03	3
AVOA	1.3714E+03	3.6450E+02	7	4.8519E+03	5.8672E+03	4
HHO	1.4525E+03	2.4893E+02	8	8.2177E+03	6.4828E+03	7
DBO	3.6632E+02	2.5492E+02	3	8.1969E+04	3.9576E+05	10
BWO	5.9969E+02	5.3413E+02	5	3.0037E+04	1.7356E+04	9
RUN	7.9802E+02	2.2139E+02	6	**1.7821E+03**	**7.0591E+02**	1
KH	4.7308E+02	3.4549E+02	4	2.1868E+03	2.1341E+03	2
GTO	1.8871E+03	1.8669E+03	9	2.7473E+04	9.8438E+04	8
RIME	**4.9575E-01**	**6.9862E-01**	1	5.8755E+03	5.5991E+03	5

Table 7: PTs of the WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2022 F7–F12

	F7			F8		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	4.1525E+01	1.4704E+01	3	2.1486E+01	8.6583E-01	2
WOA	1.6103E+02	4.7676E+01	9	4.5938E+01	2.3066E+01	9
AVOA	8.0620E+01	3.1980E+01	5	2.7885E+01	7.4866E+00	5
HHO	9.4890E+01	2.8364E+01	6	4.3127E+01	1.2934E+01	8
DBO	6.9988E+01	3.4356E+01	4	3.0668E+01	7.6034E+00	6
BWO	3.3800E+01	**5.4200E+00**	2	2.3308E+01	**4.0673E-01**	3
RUN	9.6125E+01	2.9179E+01	7	2.5906E+01	3.1172E+00	4
KH	1.0973E+02	1.8141E+01	8	3.9228E+01	3.6102E+01	7
GTO	2.6124E+02	1.1976E+02	10	1.4235E+02	1.4138E+02	10
RIME	**2.7681E+01**	8.9808E+00	1	**2.1284E+01**	4.4547E-01	1
	F9			F10		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	**1.6534E+02**	**2.9621E-13**	1	1.5661E+02	1.1885E+02	6
WOA	1.8120E+02	5.2072E-01	7	1.3278E+03	1.0443E+03	10
AVOA	1.8078E+02	8.0723E-09	2	2.6289E+02	2.4370E+02	7
HHO	1.8138E+02	2.9318E-01	8	3.3453E+02	3.5019E+02	8
DBO	1.8145E+02	2.0522E+00	9	**1.0050E+02**	**1.0607E-01**	1
BWO	1.8100E+02	1.5593E-01	6	1.0975E+02	3.4526E+01	4
RUN	1.8078E+02	1.1474E-03	4	1.1085E+02	3.8776E+01	5
KH	1.8079E+02	2.1736E-02	5	3.3617E+02	5.0264E+02	9
GTO	1.8264E+02	2.0246E+00	10	1.0200E+02	8.0501E-01	2
RIME	1.8078E+02	3.7357E-04	3	1.0566E+02	5.2349E+01	3
	F11			F12		
	Mean	Std	Rank	Mean	Std	Rank
WGOA	3.1667E+02	3.7905E+01	4	**2.0000E+02**	**3.0964E-04**	1
WOA	3.3152E+02	7.9007E+01	5	3.0983E+02	5.9608E+01	8
AVOA	3.1333E+02	7.3030E+01	2	2.6564E+02	1.7424E+01	6
HHO	3.8712E+02	1.4355E+02	9	3.2118E+02	5.3942E+01	9
DBO	**3.0448E+02**	1.5254E+02	1	2.7884E+02	3.7238E+01	7
BWO	4.0704E+02	**1.5717E+01**	10	2.4006E+02	3.7017E+00	3
RUN	3.1345E+02	3.4532E+01	3	2.4962E+02	7.6971E+00	4
KH	3.3579E+02	1.2266E+02	6	3.3431E+02	4.4116E+01	10
GTO	3.3904E+02	9.6021E+01	7	2.6340E+02	1.6562E+01	5
RIME	3.4681E+02	8.1833E+01	8	2.3956E+02	4.3115E+00	2

Table 8: Average rankings of WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2017

Algorithms	+/−/=	Mean	Rank
WGOA	∼	3.207	1
WOA	24/2/3	5.655	6
AVOA	16/11/2	6.207	8
HHO	21/3/5	4.897	5
DBO	24/1/4	5.862	7
BWO	28/1/0	8.966	10
RUN	16/10/3	4.862	4
KH	17/7/5	3.655	2
GTO	28/1/0	6.931	9
RIME	15/12/2	4.759	3

Table 9: Average rankings of WGOA, WOA [41], AVOA [42], HHO [43], DBO [44], BWO [45], RUN [46], KH [47], GTO [48] and RIME [22] for CEC2022

Algorithms	+/−/=	Mean	Rank
WGOA	∼	2.500	1
WOA	11/0/1	7.917	9
AVOA	9/1/2	4.417	4
HHO	11/0/1	7.333	8
DBO	9/1/2	5.917	6
BWO	10/2/0	5.500	5
RUN	8/3/1	4.250	3
KH	10/1/1	6.083	7
GTO	10/1/1	8.167	10
RIME	6/3/3	2.917	2

The FES-convergence figures of CEC2017 are shown in Figs. 2–7. Friedman test results for the GWOA algorithm and the control group algorithms in CEC2017 and CEC2022 are shown in Fig. 8. The results of The FES-convergence figures of CEC2022 are shown in Figs. 9–11. In Figs. 2–7 and 9–11, the horizontal axis represents the FES, and the vertical axis represents the error.

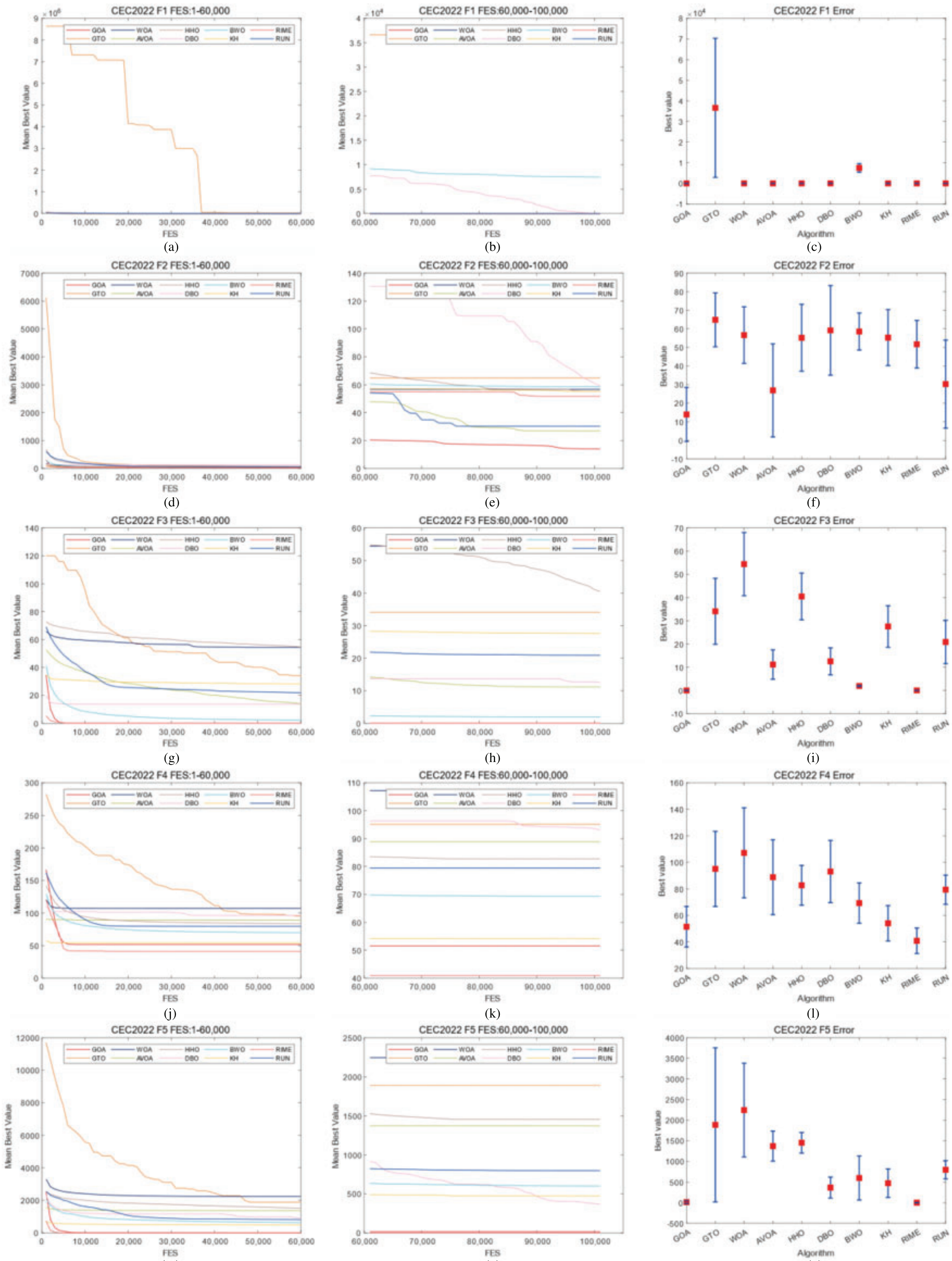

Figure 2: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2017 F1–F5

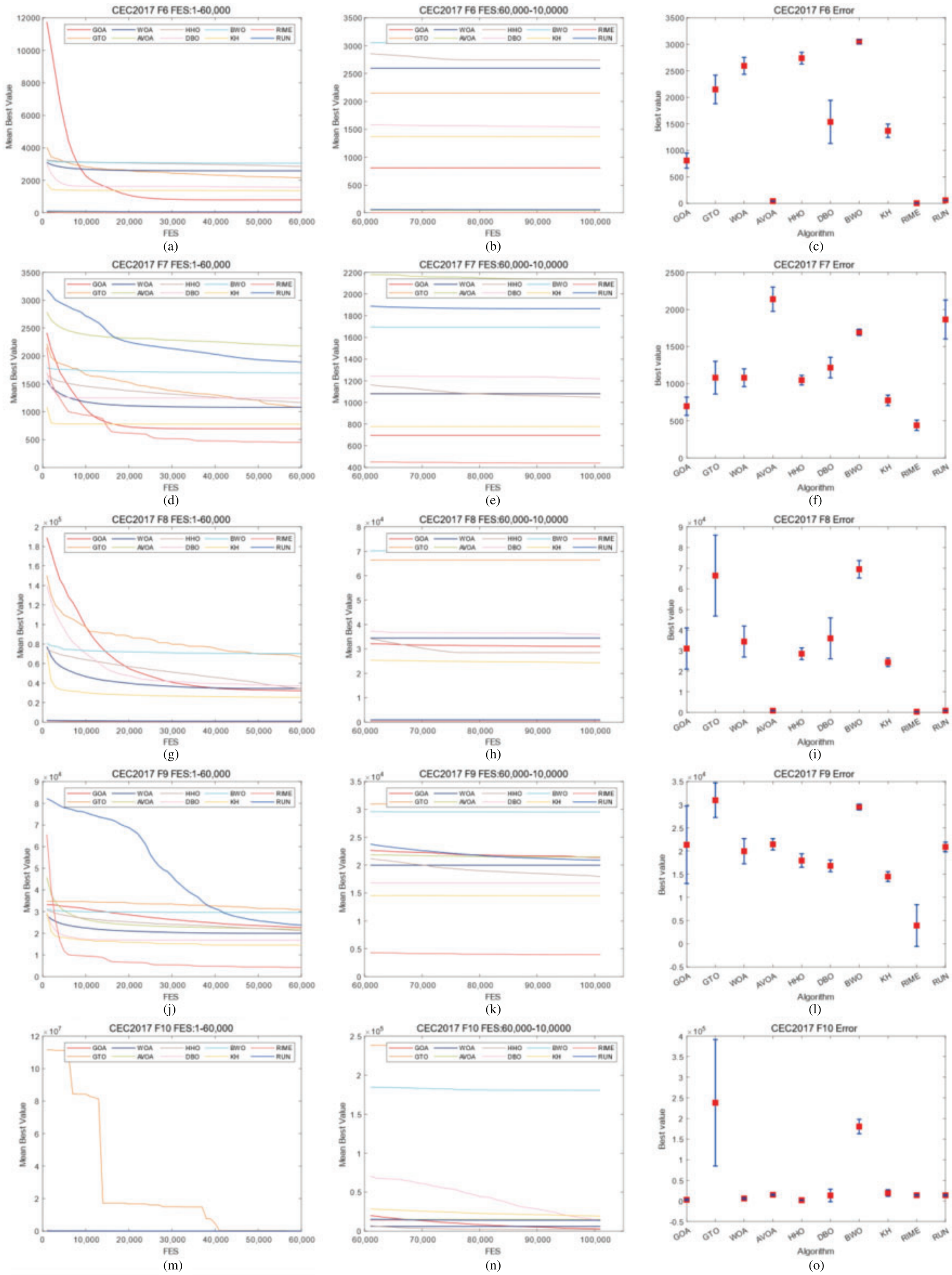

Figure 3: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2017 F6–F10

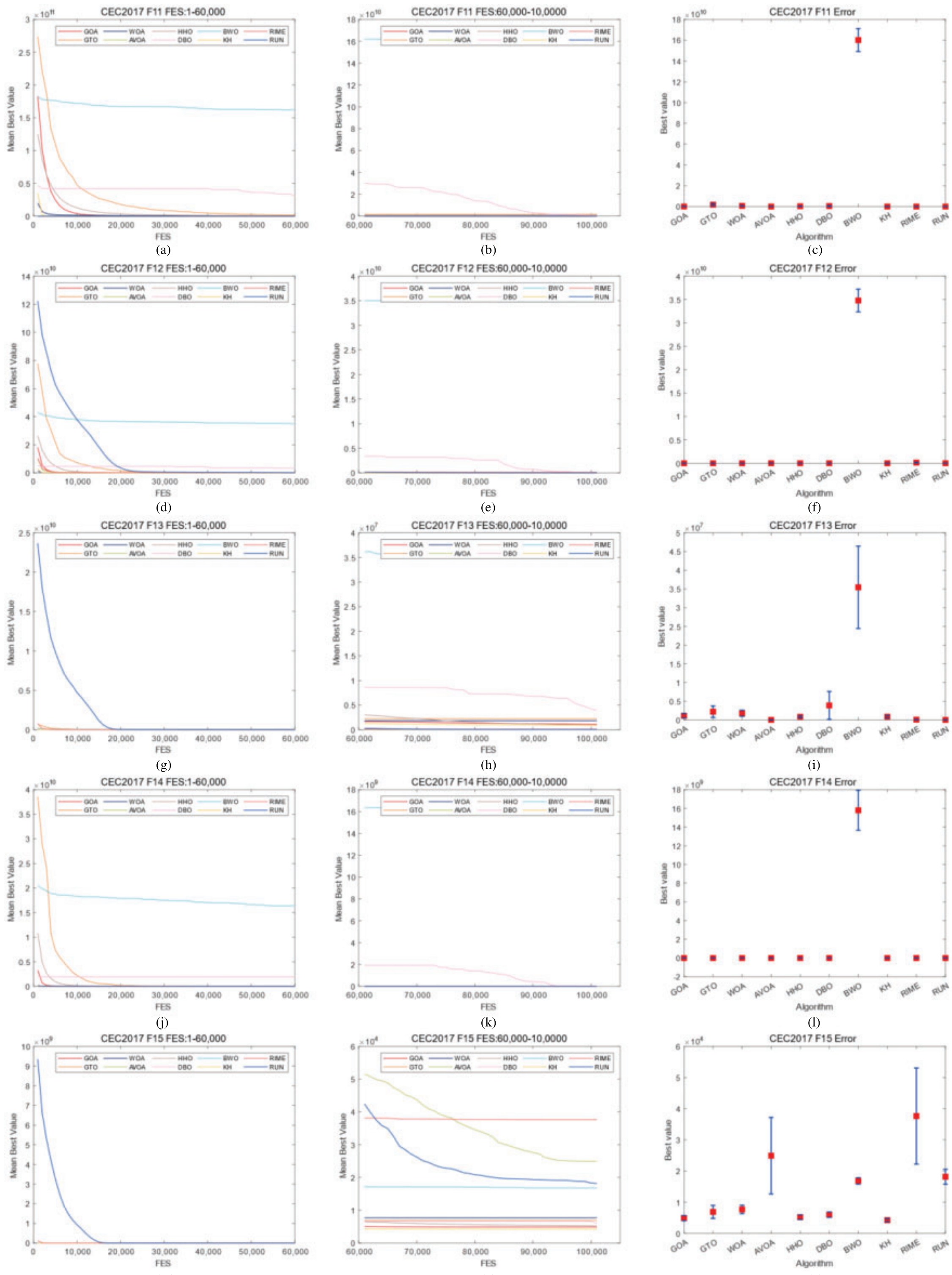

Figure 4: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2017 F11–F15

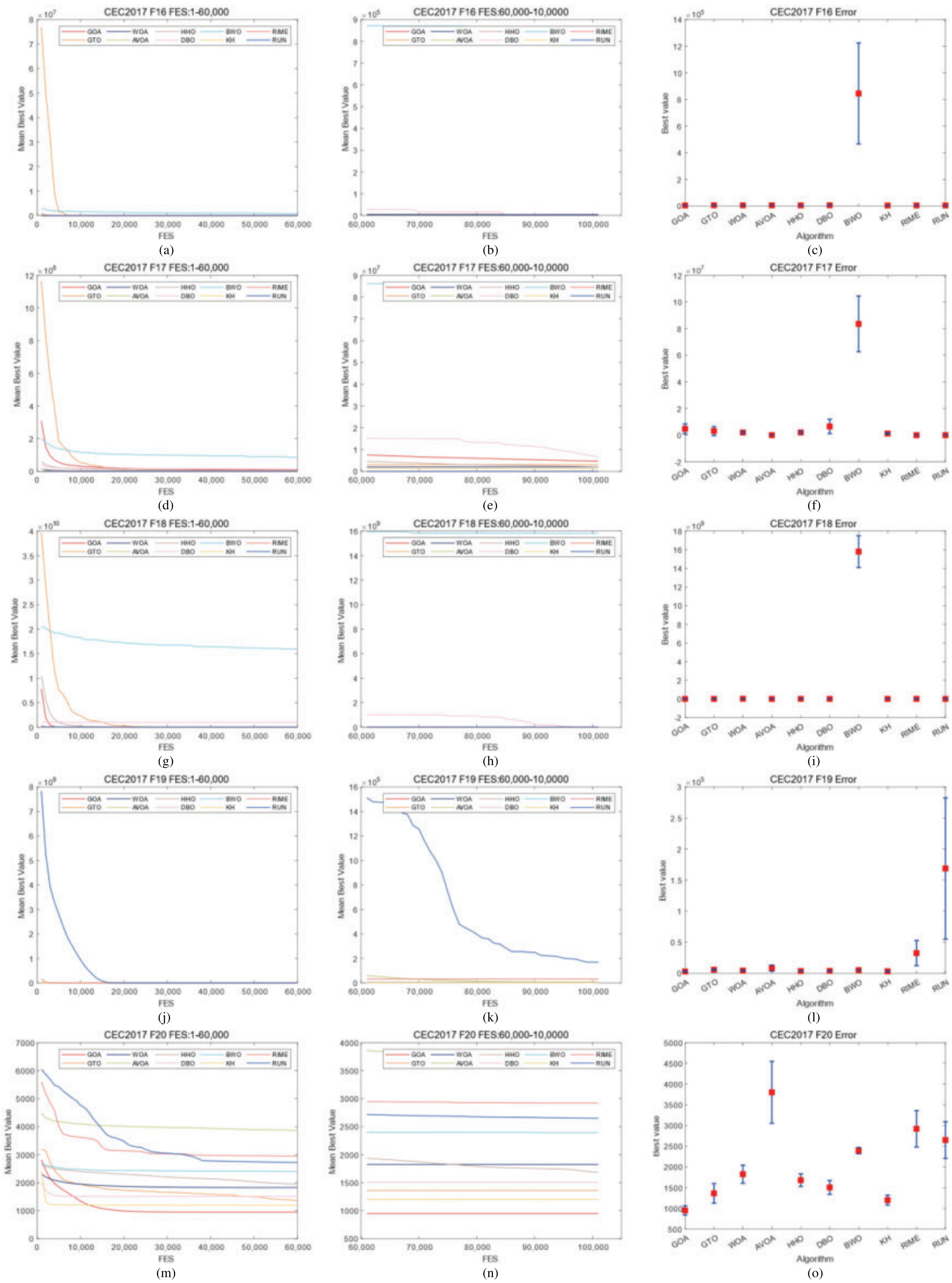

Figure 5: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2017 F16–F20

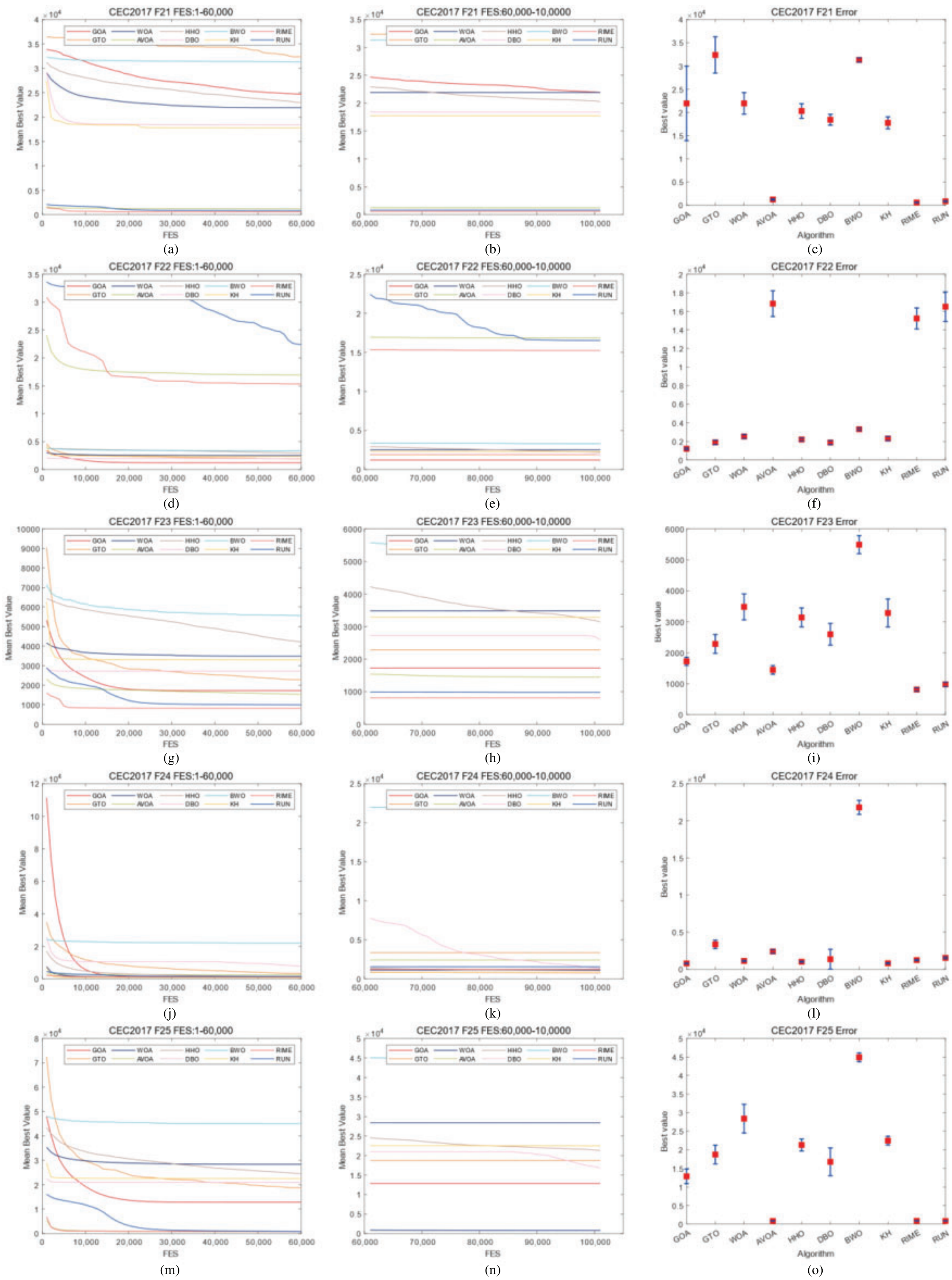

Figure 6: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2017 F21–F25

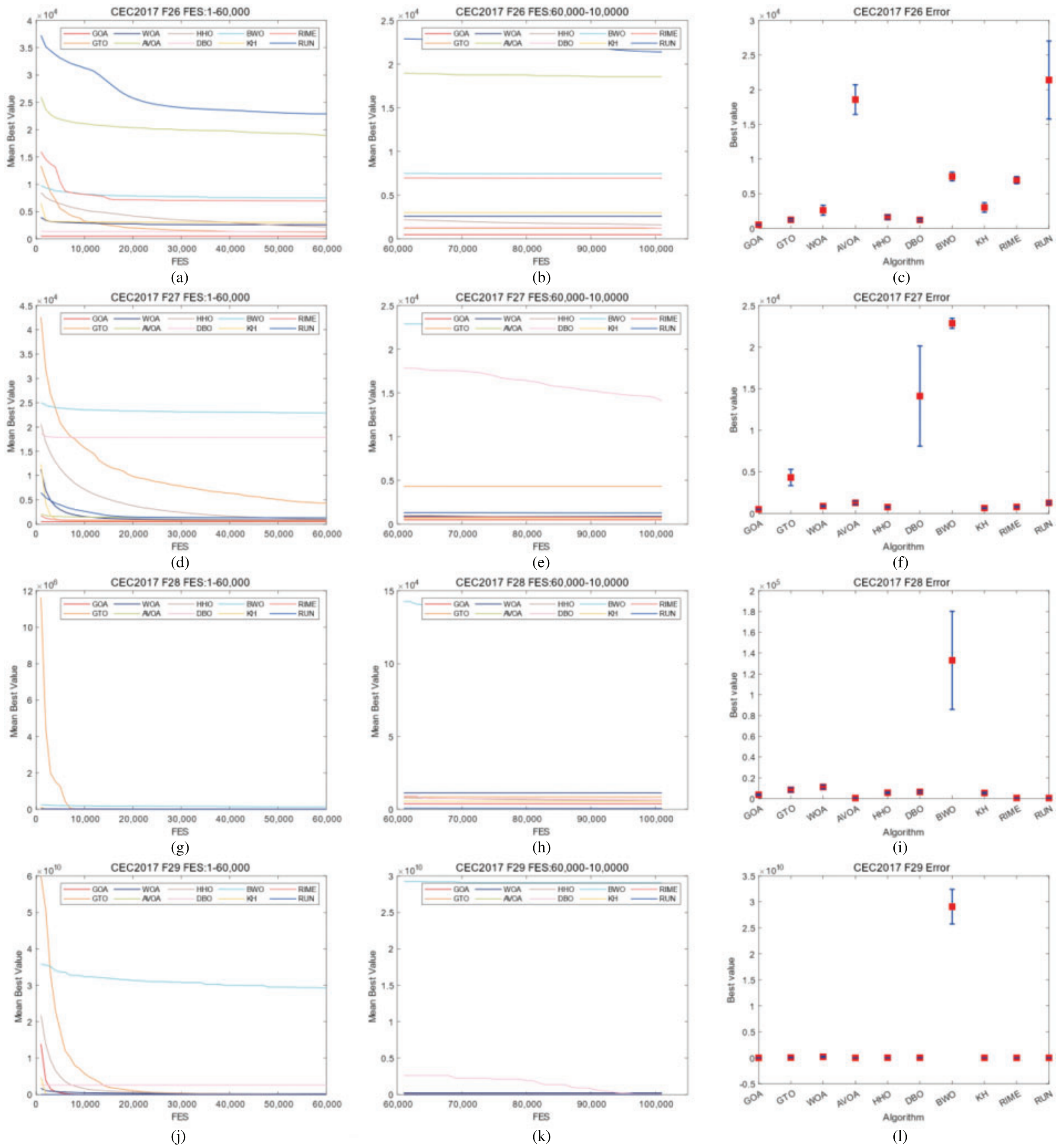

Figure 7: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2017 F26–F29

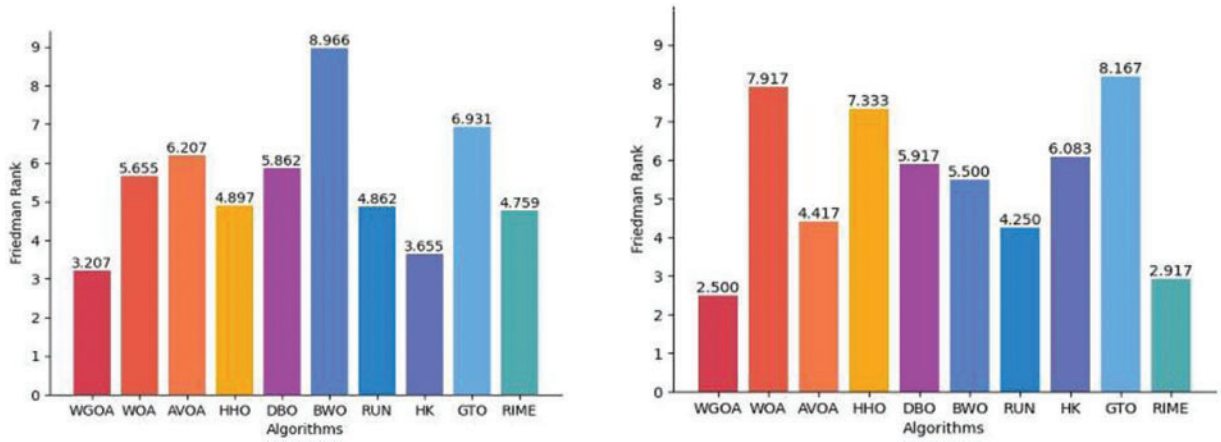

Figure 8: Friedman test results for the WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2017 (left) and CEC2022 (right)

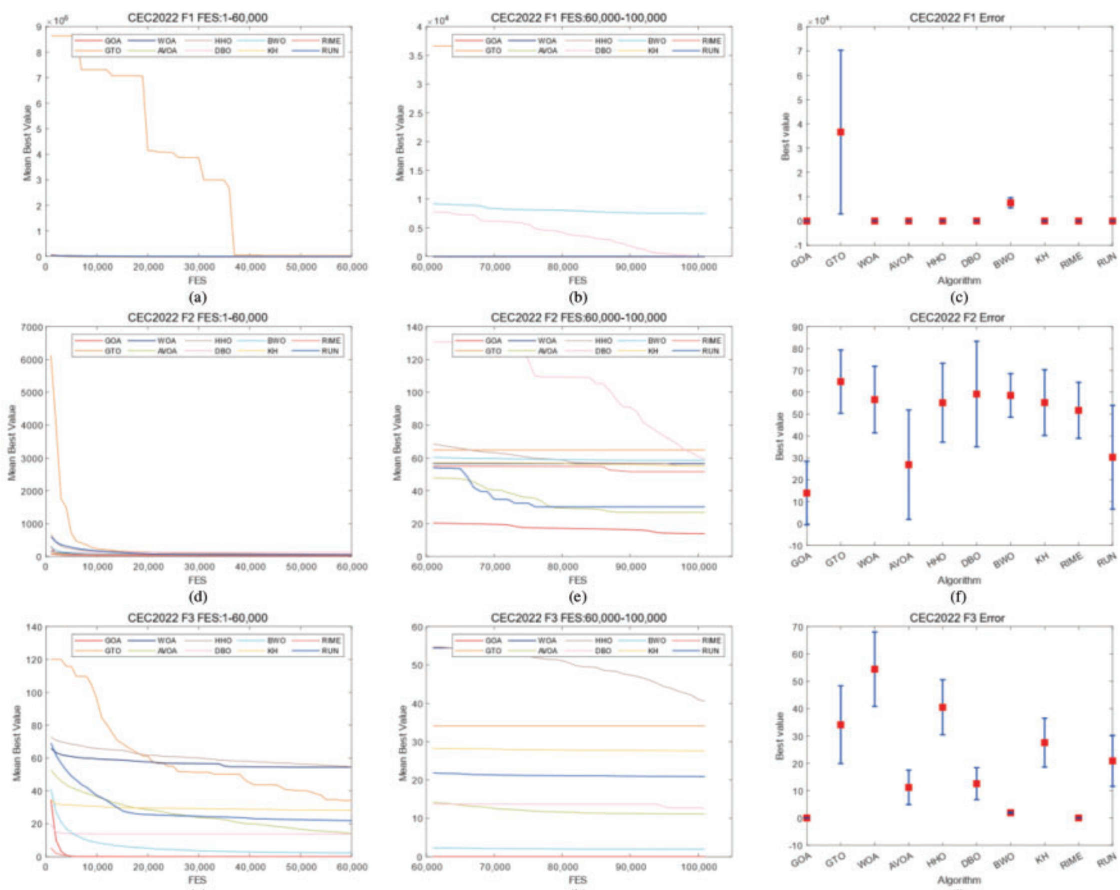

Figure 9: (Continued)

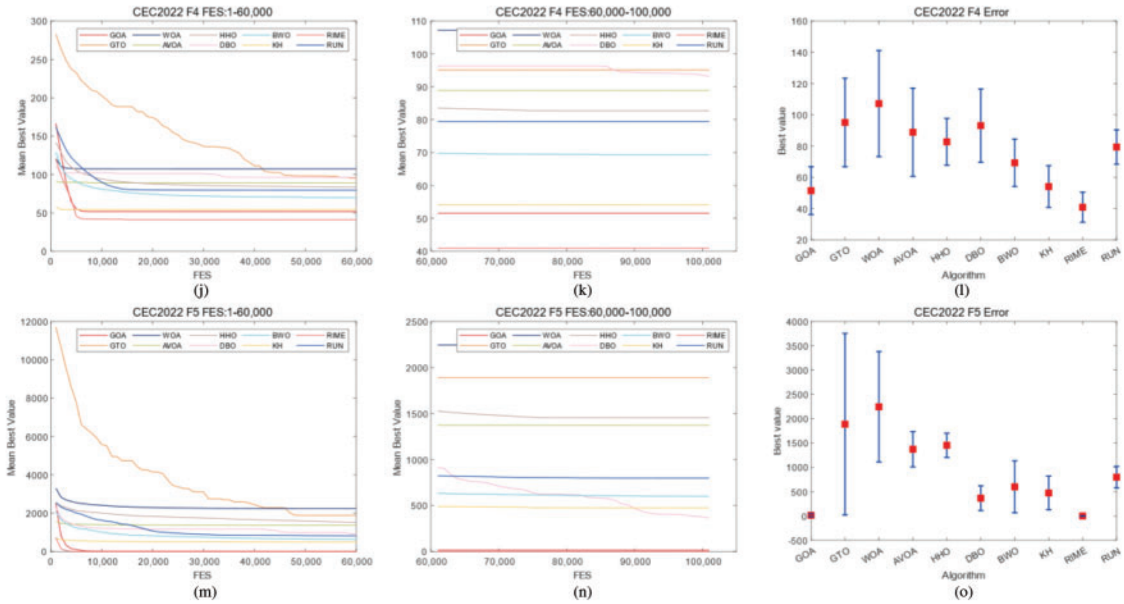

Figure 9: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2022 F1–F5

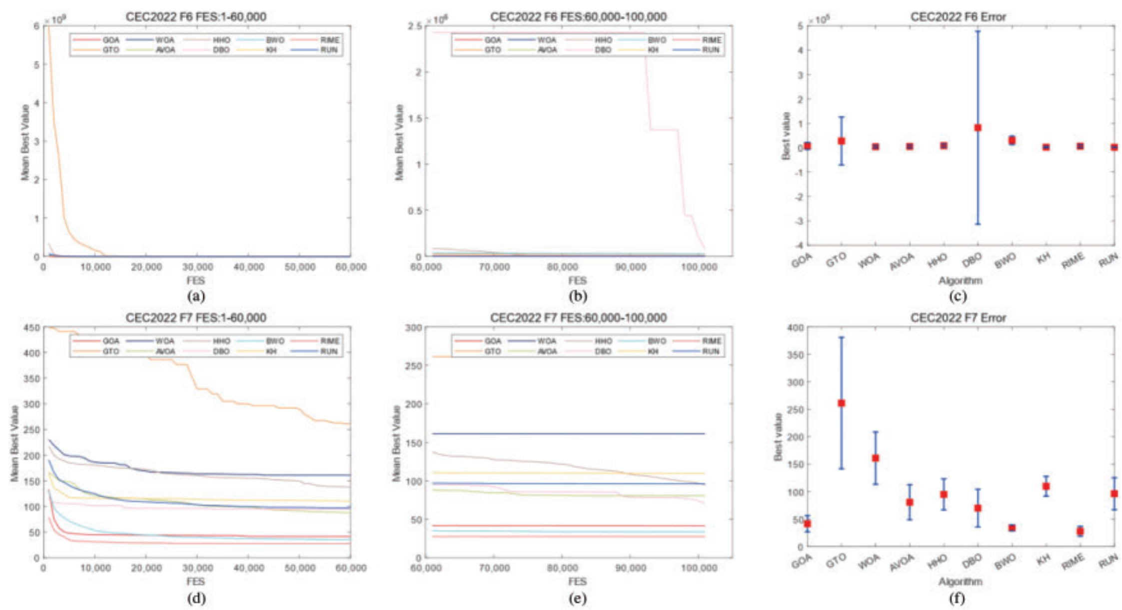

Figure 10: (Continued)

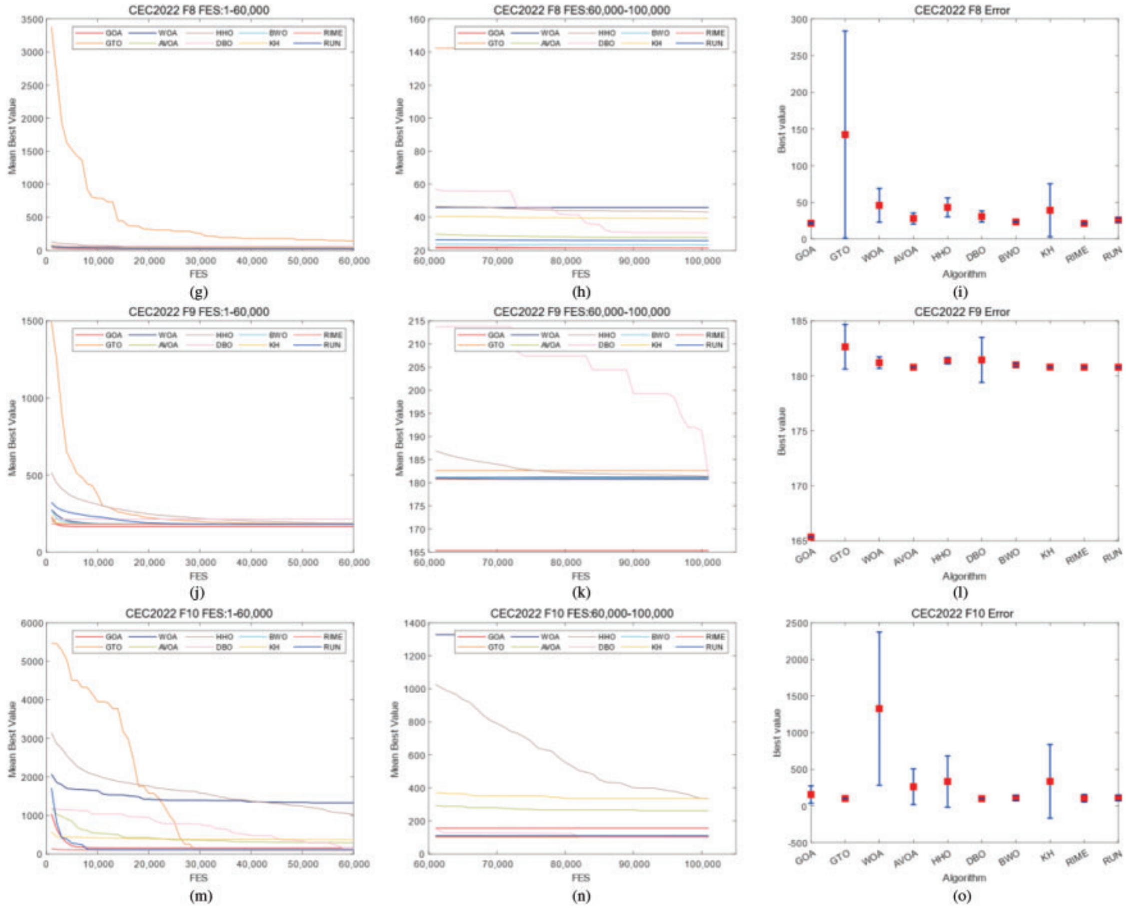

Figure 10: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2022 F6–F10

To further verify the performance of GWOA, a Wilcoxon rank-sum test under 0.05 level was conducted. The experimental results for CEC2017 are presented in Table 8, and the experimental results for CEC2022 are presented in Table 9.

4.3 Discussion

In the CEC2017 benchmark functions, WGOA ranked first in mean values among 10 functions and second among 6 functions. WGOA also achieved the first position in standard deviation among 5 functions and the second position among 4 functions. In a suite comprising 29 test functions and a selection of 10 competing algorithms, the WGOA achieved the foremost position with an average ranking of 3.207, marking it as the preeminent algorithm amongst all contenders.

In the CEC2022 benchmark functions, WGOA obtained the first rank in mean values among 5 functions and the second rank in mean values among 3 functions. WGOA achieved the first position in standard deviation among 3 functions and the second position in standard deviation among 2 functions. Across a total of 12 test functions and a cohort of 10 evaluation algorithms, the WGOA secured the leading average rank of 2.500, situating it at the apex of the algorithmic hierarchy.

The exceptional performance of the Wild Gibbon Optimization Algorithm (WGOA) is attributed to the synergistic interplay between community search and community competition strategies. The community search is dedicated to enhancing the algorithm's diversity by furnishing it with a plethora of candidate positions in each iteration. Conversely, the community competition aims to augment the algorithm's precision by exploring more promising solutions within the local search space.

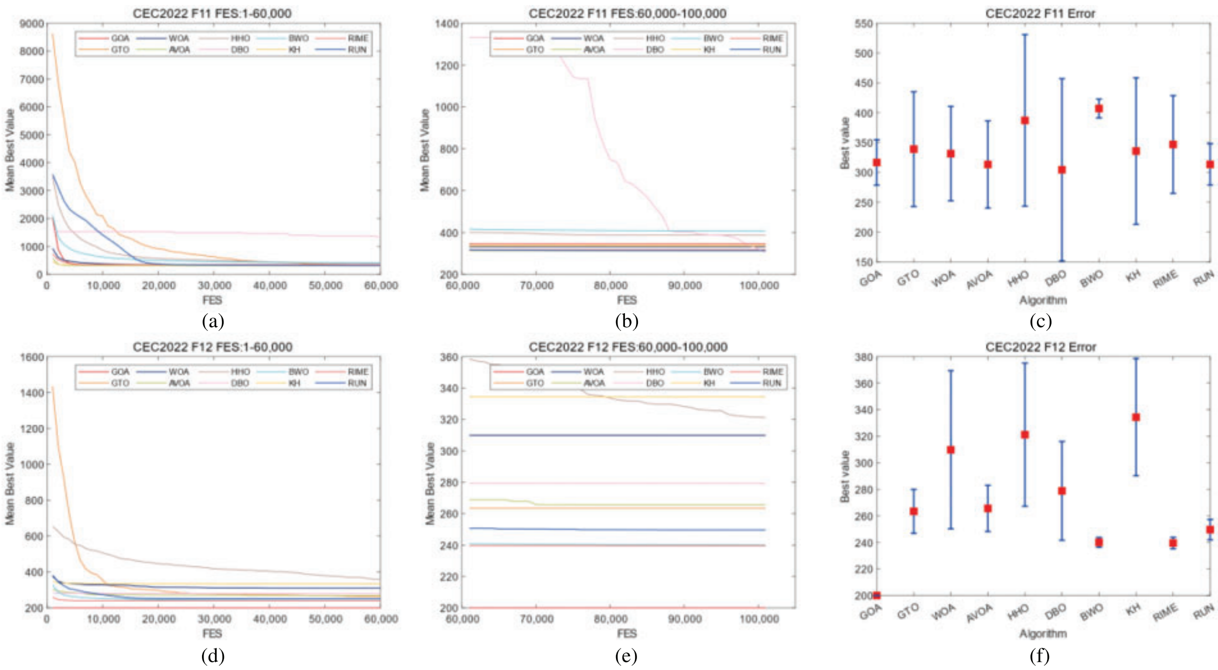

Figure 11: Convergence curves and error bars of WGOA, WOA, AVOA, HHO, DBO, BWO, RUN, KH, GTO and RIME on CEC2022 F11–F12

In CEC2017, WGOA outputs best results in F3, F5, F12, F14, F19, F20, F22, F24, F26, and F27. In CEC2022, WGOA outputs best results in F1, F2, F3, F9, F12. However, During the processing of functions F6 and F10 of CEC2022, the WGOA exhibited slow convergence rates. This can be attributed to the algorithm's search strategies failing to adequately balance exploitation and exploration when addressing such problems. Given the limitations of WGOA in handling these types of issues, integrating an adaptive mechanism into WGOA represents a significant avenue for future work.

5 Conclusions

In this paper, a novel meta-heuristic algorithm called the Wild Gibbon Optimization Algorithm (WGOA) is proposed. The algorithm is inspired by the competition among gibbon populations and is used to solve high-dimensional optimization problems. This algorithm combines a community search strategy and a community competition strategy. The community search strategy is used in the evolution of individual optima, and the community competition strategy is used in the evolution of global optima. The cooperation of these two strategies can be used to balance local search and global search, which can improve the high-dimensional search accuracy. Compared to the original algorithm, both strategies make only linear dimensional changes with $O(N)$ time complexity in terms of computational complexity. In this study, the CEC2017 and CEC2022 benchmark function are selected to evaluate the algorithm, and nine excellent optimization algorithms are selected to compare with WGOA. In the

CEC2017, the WGOA achieved the foremost position with an average ranking of 3.207, marking it as the preeminent algorithm amongst all contenders. In the CEC2022, the WGOA secured the leading average rank of 2.500, situating it at the apex of the algorithmic hierarchy. These experimental results indicate that the algorithm has a superior ability in solving optimization problems.

However, on some hybrid functions, the Wild Gibbon Optimization Algorithm (WGOA) does not achieve optimal performance and tends to fall into local optima. Furthermore, an increase in the computational burden may occur as the number of members in the gibbon family continues to grow. Therefore, incorporating an adaptive mechanism into WGOA constitutes an important area of future work. On another note, applying WGOA to real-world problems such as neural network parameter optimization and multi-objective feature selection represents a viable direction for future research.

Acknowledgement: The research was conducted under the auspices of the Hosei International Fund (HIF) Foreign Scholars Fellowship.

Funding Statement: This research was funded by Natural Science Foundation of Hubei Province Grant Numbers 2023AFB003, 2023AFB004; Education Department Scientific Research Program Project of Hubei Province of China Grant Number Q20222208; Natural Science Foundation of Hubei Province of China (No. 2022CFB076); Artificial Intelligence Innovation Project of Wuhan Science and Technology Bureau (No. 2023010402040016); JSPS KAKENHI Grant Number JP22K12185.

Author Contributions: Conceptualization, Jia Guo; methodology, Jia Guo and Jin Wang; software, Ke Yan; validation, Qiankun Zuo; formal analysis, Ruiheng Li; investigation, Zhou He; resources, Jia Guo and Jin Wang; data curation, Jia Guo; writing—original draft preparation, Jia Guo; writing—review and editing, Jin Wang; visualization, Zhou He and Dong Wang; supervision, Yuji Sato; project administration, Jia Guo and Qiankun Zuo; funding acquisition, Jia Guo. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Dataset available on request from the authors.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. W. Li, D. Y. Xu, J. Geng, and W. C. Hong, "A ship motion forecasting approach based on empirical mode decomposition method hybrid deep learning network and quantum butterfly optimization algorithm," *Nonlinear Dyn*, vol. 107, no. 3, pp. 2447–2467, 2022. doi: 10.1007/s11071-021-07139-y.

[2] B. S. Yıldız *et al.*, "A novel hybrid arithmetic optimization algorithm for solving constrained optimization problems," *Knowl.-Based Syst.*, vol. 271, pp. 110554, 2023. doi: 10.1016/j.knosys.2023.110554.

[3] S. Li, S. Zhao, Y. Zhang, J. Hong, and W. Chen, "Source-free unsupervised adaptive segmentation for knee joint MRI," *Biomed. Signal Process. Control*, vol. 92, pp. 106028, 2024. doi: 10.1016/j.bspc.2024.106028.

[4] J. Hong, Y. D. Zhang, and W. Chen, "Source-free unsupervised domain adaptation for cross-modality abdominal multi-organ segmentation," *Knowl.-Based Syst.*, vol. 250, pp. 109155, 2022. doi: 10.1016/j.knosys.2022.109155.

[5] X. Li *et al.*, "EAFP-Med: An efficient adaptive feature processing module based on prompts for medical image detection," *Expert. Syst. Appl.*, vol. 247, pp. 123334, 2024. doi: 10.1016/j.eswa.2024.123334.

[6] J. Wang, S. Y. Lu, S. H. Wang, and Y. D. Zhang, "RanMerFormer: Randomized vision transformer with token merging for brain tumor classification," *Neurocomputing*, vol. 573, pp. 127216, 2024. doi: 10.1016/j.neucom.2023.127216.

[7] X. Li *et al.*, "DBTN: An adaptive neural network for multiple-disease detection via imbalanced medical images distribution," *Appl. Intell.*, vol. 54, no. 2, pp. 2188–2210, 2024. doi: 10.1007/s10489-023-05165-4.
[8] J. Guo, G. Zhou, K. Yan, B. Shi, Y. Di and Y. Sato, "A novel hermit crab optimization algorithm," *Sci. Rep.*, vol. 13, pp. 9934, 2023. doi: 10.1038/s41598-023-37129-6.
[9] K. Sörensen, "Metaheuristics—The metaphor exposed," *Int. Trans. Operations Res.*, vol. 22, no. 1, pp. 3–18, 2015. doi: 10.1111/itor.12001.
[10] E. S. M. El-kenawy, N. Khodadadi, S. Mirjalili, A. A. Abdelhamid, M. M. Eid and A. Ibrahim, "Greylag goose optimization: Nature-inspired optimization algorithm," *Expert. Syst. Appl.*, vol. 238, pp. 122147, 2024. doi: 10.1016/j.eswa.2023.122147.
[11] B. Abdollahzadeh *et al.* "Puma optimizer (PO): A novel metaheuristic optimization algorithm and its application in machine learning," *Cluster Comput.*, 2024. doi: 10.1007/s10586-023-04221-5.
[12] S. Memarian, N. Behmanesh-Fard, P. Aryai, M. Shokouhifar, S. Mirjalili and M. del C. Romero-Ternero , "TSFIS-GWO: Metaheuristic-driven takagi-sugeno fuzzy system for adaptive real-time routing in WBANs," *Appl. Soft Comput.*, vol. 155, pp. 111427, 2024. doi: 10.1016/j.asoc.2024.111427.
[13] J. Guo *et al.*, "A twinning bare bones particle swarm optimization algorithm," *PLoS One*, vol. 17, no. 5 May, pp. 1–30, 2022. doi: 10.1371/journal.pone.0267197.
[14] C. Huang, X. Li, and Y. Wen, "AN OTSU image segmentation based on fruitfly optimization algorithm," *Alex. Eng. J.*, vol. 60, no. 1, pp. 183–188, 2021. doi: 10.1016/j.aej.2020.06.054.
[15] D. Chen, Y. Ge, Y. Wan, Y. Deng, Y. Chen and F. Zou, "Poplar optimization algorithm: A new meta-heuristic optimization technique for numerical optimization and image segmentation," *Expert. Syst. Appl.*, vol. 200, pp. 117118, 2022. doi: 10.1016/j.eswa.2022.117118.
[16] S. Chakraborty and K. Mali, "SuFMoFPA: A superpixel and meta-heuristic based fuzzy image segmentation approach to explicate COVID-19 radiological images," *Expert. Syst. Appl.*, vol. 167, pp. 114142, 2021. doi: 10.1016/j.eswa.2020.114142.
[17] X. Deng, R. Li, L. Zhao, K. Wang, and X. Gui, "Multi-obstacle path planning and optimization for mobile robot," *Expert. Syst. Appl.*, vol. 183, pp. 115445, 2021. doi: 10.1016/j.eswa.2021.115445.
[18] Md R. Islam, P. Protik, S. Das, and P. K. Boni, "Mobile robot path planning with obstacle avoidance using chemical reaction optimization," *Soft Comput.*, vol. 25, no. 8, pp. 6283–6310, 2021. doi: 10.1007/s00500-021-05615-6.
[19] R. Chen, H. Shen, and Y. Lai, "A metaheuristic optimization algorithm for energy efficiency in digital twin," *Internet of Things and Cyber-Phys. Syst.*, vol. 2, pp. 159–169, 2022. doi: 10.1016/j.iotcps.2022.08.001.
[20] M. Khanali *et al.* "Multi-objective optimization of energy use and environmental emissions for walnut production using imperialist competitive algorithm," *Appl. Energy*, vol. 284, pp. 116342, 2021. doi: 10.1016/j.apenergy.2020.116342.
[21] A. F. Güven and M. M. Samy, "Performance analysis of autonomous green energy system based on multi and hybrid metaheuristic optimization approaches," *Energy Convers. Manag.*, vol. 269, pp. 116058, 2022. doi: 10.1016/j.enconman.2022.116058.
[22] H. Su *et al.* "RIME: A physics-based optimization," *Neurocomputing*, vol. 532, pp. 183–214, 2023. doi: 10.1016/j.neucom.2023.02.010.
[23] P. C. Song, S. C. Chu, J. S. Pan, and H. Yang, "Simplified phasmatodea population evolution algorithm for optimization," *Complex Intell. Syst.*, vol. 8, no. 4, pp. 2749–2767, 2022. doi: 10.1007/s40747-021-00402-0.
[24] D. Połap and M. Woźniak, "Red fox optimization algorithm," *Expert. Syst. Appl.*, vol. 166, pp. 114107, 2021. doi: 10.1016/j.eswa.2020.114107.
[25] X. L. Li, R. Serra, and J. Olivier, "A multi-component PSO algorithm with leader learning mechanism for structural damage detection," *Appl. Soft Comput.*, vol. 116, pp. 108315, 2022. doi: 10.1016/j.asoc.2021.108315.
[26] H. Zhang and Q. Peng, "PSO and K-means-based semantic segmentation toward agricultural products," *Future Gener. Comput. Syst.*, vol. 126, pp. 82–87, 2022. doi: 10.1016/j.future.2021.06.059.

[27] D. Li, W. Guo, A. Lerch, Y. Li, L. Wang and Q. Wu, "An adaptive particle swarm optimizer with decoupled exploration and exploitation for large scale optimization," *Swarm Evol. Comput.*, vol. 60, 2021. doi: 10.1016/j.swevo.2020.100789.

[28] L. Zhang, T. Gao, G. Cai, and K. L. Hai, "Research on electric vehicle charging safety warning model based on back propagation neural network optimized by improved gray wolf algorithm," *J. Energy Storage*, vol. 49, pp. 104092, 2022. doi: 10.1016/j.est.2022.104092.

[29] E. A. Atta, A. F. Ali, and A. A. Elshamy, "A modified weighted chimp optimization algorithm for training feed-forward neural network," *PLoS One*, vol. 18, no. 3, pp. e0282514, Mar. 2023. doi: 10.1371/journal.pone.0282514.

[30] J. Guo, J. Li, Y. Sato, and Z. Yan, "A gated recurrent unit model with fibonacci attenuation particle swarm optimization for carbon emission prediction," *Processes*, vol. 12, pp. 1063, 2024. doi: 10.3390/pr12061063.

[31] N. Bacanin *et al.*, "Hybridized sine cosine algorithm with convolutional neural networks dropout regularization application," *Sci. Rep.*, vol. 12, no. 1, pp. 6302, 2022. doi: 10.1038/s41598-022-09744-2.

[32] M. Shokouhifar, "FH-ACO: Fuzzy heuristic-based ant colony optimization for joint virtual network function placement and routing," *Appl. Soft Comput.*, vol. 107, pp. 107401, 2021. doi: 10.1016/j.asoc.2021.107401.

[33] L. Qian, M. Khishe, Y. Huang, and S. Mirjalili, "SEB-ChOA: An improved chimp optimization algorithm using spiral exploitation behavior," *Neural Comput. Appl.*, vol. 36, no. 9, pp. 4763–4786, 2024. doi: 10.1007/s00521-023-09236-y.

[34] L. Jovanovic *et al.*, "Multi-step crude oil price prediction based on LSTM approach tuned by salp swarm algorithm with disputation operator," *Sustainability*, vol. 14, pp. 14616, Nov. 2022. doi: 10.3390/su142114616.

[35] C. Tang, W. Sun, M. Xue, X. Zhang, H. Tang and W. Wu, "A hybrid whale optimization algorithm with artificial bee colony," *Soft Comput.*, vol. 26, no. 5, pp. 2075–2097, 2022. doi: 10.1007/s00500-021-06623-2.

[36] S. Thawkar, S. Sharma, M. Khanna, and L. Kumar Singh , "Breast cancer prediction using a hybrid method based on butterfly optimization algorithm and ant lion optimizer," *Comput. Biol. Med.*, vol. 139, pp. 104968, 2021. doi: 10.1016/j.compbiomed.2021.104968.

[37] S. Deb and X. Z. Gao, "A hybrid ant lion optimization chicken swarm optimization algorithm for charger placement problem," *Complex Intell. Syst.*, vol. 8, no. 4, pp. 2791–2808, 2022. doi: 10.1007/s40747-021-00510-x.

[38] M. M. Emam, E. H. Houssein, and R. M. Ghoniem, "A modified reptile search algorithm for global optimization and image segmentation: Case study brain MRI images," *Comput. Biol. Med.*, vol. 152, pp. 106404, 2023. doi: 10.1016/j.compbiomed.2022.106404.

[39] F. MiarNaeimi, G. Azizyan, and M. RasKHi, "Horse herd optimization algorithm: A nature-inspired algorithm for high-dimensional optimization problems," *Knowl.-Based Syst.*, vol. 213, pp. 106711, 2021. doi: 10.1016/j.knosys.2020.106711.

[40] W. Ma, X. Zhou, H. Zhu, L. Li, and L. Jiao, "A two-stage hybrid ant colony optimization for high-dimensional feature selection," *Pattern Recognit.*, vol. 116, pp. 107933, 2021. doi: 10.1016/j.patcog.2021.107933.

[41] C. Zhong, G. Li, and Z. Meng, "Beluga whale optimization: A novel nature-inspired metaheuristic algorithm," *Knowl. Based Syst.*, vol. 251, pp. 109215, 2022. doi: 10.1016/j.knosys.2022.109215.

[42] B. Abdollahzadeh, F. S. Gharehchopogh, and S. Mirjalili, "African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems," *Comput. Ind. Eng.*, vol. 158, pp. 107408, 2021. doi: 10.1016/j.cie.2021.107408.

[43] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, 2019. doi: 10.1016/j.future.2019.02.028.

[44] J. Xue and B. Shen, "Dung beetle optimizer: A new meta-heuristic algorithm for global optimization," *J. Supercomput.*, vol. 79, no. 7, pp. 7305–7336, 2023. doi: 10.1007/s11227-022-04959-6.

[45] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, 2016. doi: 10.1016/j.advengsoft.2016.01.008.

[46] I. Ahmadianfar, A. A. Heidari, A. H. Gandomi, X. Chu, and H. Chen, "RUN beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method," *Expert. Syst. Appl.*, vol. 181, pp. 115079, 2021. doi: 10.1016/j.eswa.2021.115079.

[47] A. H. Gandomi and A. H. Alavi, "Krill herd: A new bio-inspired optimization algorithm," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 17, no. 12, pp. 4831–4845, 2012. doi: 10.1016/j.cnsns.2012.05.010.

[48] B. Abdollahzadeh, F. Soleimanian Gharehchopogh, S. Mirjalili, "Artificial gorilla troops optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems," *Int. J. Intell. Syst.*, vol. 36, no. 10, pp. 5887–5958, Oct. 2021. doi: 10.1002/int.22535.

[49] G. Zhou, J. Du, J. Guo, and G. Li, "A novel hippo swarm optimization: For solving high-dimensional problems and engineering design problems," *J. Comput. Des. Eng.*, 2024. doi: 10.1093/jcde/qwae035.