CCSA: Conscious Neighborhood-based Crow Search Algorithm for Solving Global Optimization Problems

Hoda Zamani[1, 2], Mohammad-Hossein Nadimi-Shahraki[1, 2, *], Amir H Gandomi[3,4]

[1]Faculty of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran
[2]Big Data Research Center, Najafabad Branch, Islamic Azad University, Najafabad, Iran
[3]Faculty of Engineering & Information Technology, University of Technology Sydney, Ultimo, NSW 2007, Australia
[4]School of Business, Stevens Institute of Technology, Hoboken, NJ 07030, USA
[*] Correspondence: nadimi@ieee.org

Abstract

In this paper, a conscious neighborhood-based crow search algorithm (CCSA) is proposed for solving global optimization and engineering design problems. It is a successful improvement to tackle the imbalance search strategy and premature convergence problems of the crow search algorithm. CCSA introduces three new search strategies called neighborhood-based local search (NLS), non-neighborhood based global search (NGS) and wandering around based search (WAS) in order to improve the movement of crows in different search spaces. Moreover, a neighborhood concept is defined to select the movement strategy between NLS and NGS consciously, which enhances the balance between local and global search. The proposed CCSA is evaluated on several benchmark functions and four applied problems of engineering design. In all experiments, CCSA is compared by other state-of-the-art swarm intelligence algorithms: BA, CLPSO, GWO, EEGWO, WOA, KH, ABC, GABC, and Best-so-far ABC. The experimental and statistical results show that CCSA is very competitive especially for large-scale optimization problems, and it is significantly superior to the compared algorithms. Furthermore, the proposed algorithm also finds the best optimal solution for the applied problems of engineering design.

Keywords: Optimization, Bio-inspired metaheuristic algorithm, Swarm intelligence algorithms, Crow search algorithm, Conscious neighborhood-based crow search algorithm.

1. Introduction

Global numerical optimization problems have challenges such as being complex nonlinear, multi-modality, hybrid, composite, and large-scale. Solving these problems by exhaustive search algorithms increases the computational and time complexities [1]. To overcome these complexities, metaheuristic algorithms present a local and randomization exploration framework to extract near-optimal solutions with cognitive computational complexity [2, 3]. They start their exploration process with the minimum knowledge of the

problem and through trial and error, using two local and global search strategies. Accessing new areas of the search space, the global search strategy results in diversity among solutions, while the local search strategy concentrates the exploration around the near-optimal solution [4].

Problems can be solved either in a continuous or discrete space, therefore, metaheuristic algorithms are also classified into two categories continuous and binary (discrete). The ant colony algorithm (ACO) [5] was proposed for solving the discrete problems. It is inspired by the extraordinary ability of the ants in finding the shortest path to the food source. Consistently, different methods are used to adopt a continuous metaheuristic algorithm to work in a binary search space [6]. It is worth mentioning here that both continuous and binary metaheuristic algorithms strive to create a proper balance between local and global search, which have a direct impact on the efficiency and convergence behavior.

Mother Nature, because of its longtime presence as the most significant problem solver, has the potential to inspire us to strike a balance between these two search strategies [7]. Accordingly, bio-inspired metaheuristic algorithms are inspired by nature. Their robustness and powerful adaptation has made them suitable for a wide range of complex problems [8]. One of the exciting sub-branches of this category is swarm intelligence (SI) algorithms, which are inspired by the optimization behavior in the life of insects, aquatic animals, terrestrial animals, and birds. Based on their behaviors, there have been proposed many successful algorithms such as particle swarm optimization (PSO) [9], the artificial bee colony (ABC) algorithm [10], whale optimization algorithm (WOA) [11], krill herd (KH) [2] and spider monkey optimization (SMO) algorithm [12].

Such the population-based algorithms have good exploration and sharing information between their swarms which usually increases their robustness. This ability or robustness makes them quite powerful to solve a wide range of application such as mechanical engineering design, pattern recognition, and signal processing. However, some of these algorithms such as PSO and ABC suffer from weak robustness and imbalance between local and global search strategies for complicated problems. Therefore, they have been improved to introducing better algorithms like comprehensive learning particle swarm optimization (CLPSO) [13] and gbest-guided ABC (GABC) [14]. Meanwhile, the slower convergence speed of some metaheuristic can be an important limitation for applying them when time is critical. Thus, algorithms such as NSABC [15] and qABC [16] have been introduced to increase the convergence speed.

Crow search algorithm (CSA) [17] is another successful SI algorithm, which was recently proposed, and it is based on the social behavior of the crows. The evaluation results of CSA

show that it can solve the continuous optimization problems, especially in science and engineering. However, it selects the movement strategy by a random comparison, which decreases the balance between local and global search and it converges to non-optimal solutions. Moreover, the results indicate that its robustness is weak to deal with a wide range of problem and this algorithm cannot escape from exit local optima especially in high dimensional problems.

To overcome these shortcomings, in this paper, an improvement of CSA named conscious neighborhood-based crow search algorithm (CCSA) is proposed. CCSA improves the movement of crows by introducing three new search strategies: Neighborhood-based Local Search (NLS), Non-Neighborhood based Global Search (NGS) and Wandering Around based Search (WAS). NLS improves the local search, and NGS increases the domain of global search, causing the CCSA to be less influenced by the asymmetrical search space of the various problems. WAS is inspired by another social behavior of crows in nature named wandering around. It provides another movement opportunity for those crows located in the flat zone or local optima which could not update their positions.

Moreover, in CCSA, a new neighborhood concept is defined to perceive the search space and select the movement strategies consciously. After determining the neighborhood of a crow by this definition, if the quality of neighbor crows is better than non-neighbor crows, then the crow selects NLS strategy. Otherwise, it moves towards the best crow out of its neighborhood using NGS strategy.

The efficiency of the proposed CCSA is experimentally evaluated and compared with other state-of-the-art swarm intelligence algorithms, which are named compared algorithms from now on. The compared algorithms consist of crow search algorithm (CSA) [17], bat algorithm (BA) [18,19], comprehensive learning particle swarm optimization (CLPSO) [13], gray wolf optimization (GWO) [20], exploration-enhanced GWO (EEGWO) [21], whale optimization algorithm (WOA) [11], krill herd (KH) [2], artificial bee colony (ABC) [22], gbest-guided ABC (GABC) [14] and Best-so-far ABC [23].

This evaluation is conducted by various experiments on benchmark function CEC 2017 [24] with different dimensions of 30, 50 and 100. The experimental results show that the proposed CCSA performs better than the compared algorithms on unimodal, simple multi-modal, hybrid and composition functions, especially with dimensions over 50 and 100. Moreover, another experiment set on benchmark functions CEC 2010 [25] with dimension 1000 proves the efficiency of the proposed CCSA for the large-scale global optimization problems. In addition, CCSA is statistically evaluated by tests of Mean Absolute Error (MAE)

and Friedman test revealed, and results showed that CCSA is superior to the compared algorithms. Finally, the applicability of the proposed algorithm for solving real application problems is also tested by four different engineering design problems. The results of this experiment set show that CCSA outperforms the compared algorithms for solving these engineering problems.

2. Related work

Bio-inspired metaheuristic algorithm is a new research paradigm that is powerful and efficiently used in solving modern nonlinear numerical global optimization problems [26]. As shown in Fig.1, bio-inspired metaheuristic algorithms can be classified into evolutionary metaheuristic and swarm intelligence (SI) algorithms.

Evolutionary metaheuristic algorithms are a sub-branch of evolutionary computation that have been formed based on Darwin's theory of biological evolution and mostly mimic evolutionary concepts in nature. Evolutionary metaheuristic algorithms use two main operators cross-over and mutation [7,27,28]. The cross-over is to combine the solutions during the optimization process and is the essential mechanism to exploit the search space. While the mutation operators are to change some of the solution, which emphasizes the exploration. There have been proposed some well-known evolutionary-based algorithms such as genetic algorithm (GA) [29], genetic programming (GP) [30], evolution strategy (ES) [31] and differential evolution (DE) [32]. Among them, the DE algorithm is an accurate, reasonably fast and robust optimizer for solving a wide range of optimization problems. However, DE cannot guarantee to find the global optimum, especially for complex problems. Therefore, there is still much attention to improve this algorithm; recently IDEI [28], DECMSA [33], NDE [34], QUATRE-EAR [35] and PaDE [36] were proposed.

The SI algorithms are created based on simple behavioral models of animals and organisms with each other and their surroundings. They can map complex optimization problems into simple behavioral models to find optimal solutions [37]. In these algorithms, each organism is a search agent, which explores the search space using the local and global search strategy defined in their algorithms. Although SI algorithms such as PSO [9], ABC [22] and WOA [11] have attracted the attention of many researchers for solving optimization problems, they may have weaknesses such as local optima trapping, premature convergence and the imbalance search strategy [14, 15, 17]. Therefore, there have been proposed many improvements to tackle their weaknesses.

One of the significant challenges which affect the efficiency of metaheuristic algorithm is the ability to strike a balance between local and global search [37,38]. This balance enhances

the algorithm to cross the local optima and reach the promising areas of the search space. To overcome this challenge, numerous studies have been conducted on various species of organisms regarding their optimization behavior in the nature for survival, leading to the creation of a new SI algorithm. As shown in Fig.1, SI algorithms can be categorized into four categories: insects, terrestrial animals, aquatic animals, and birds. In the following, some well-known SI algorithms of these categories are reviewed.

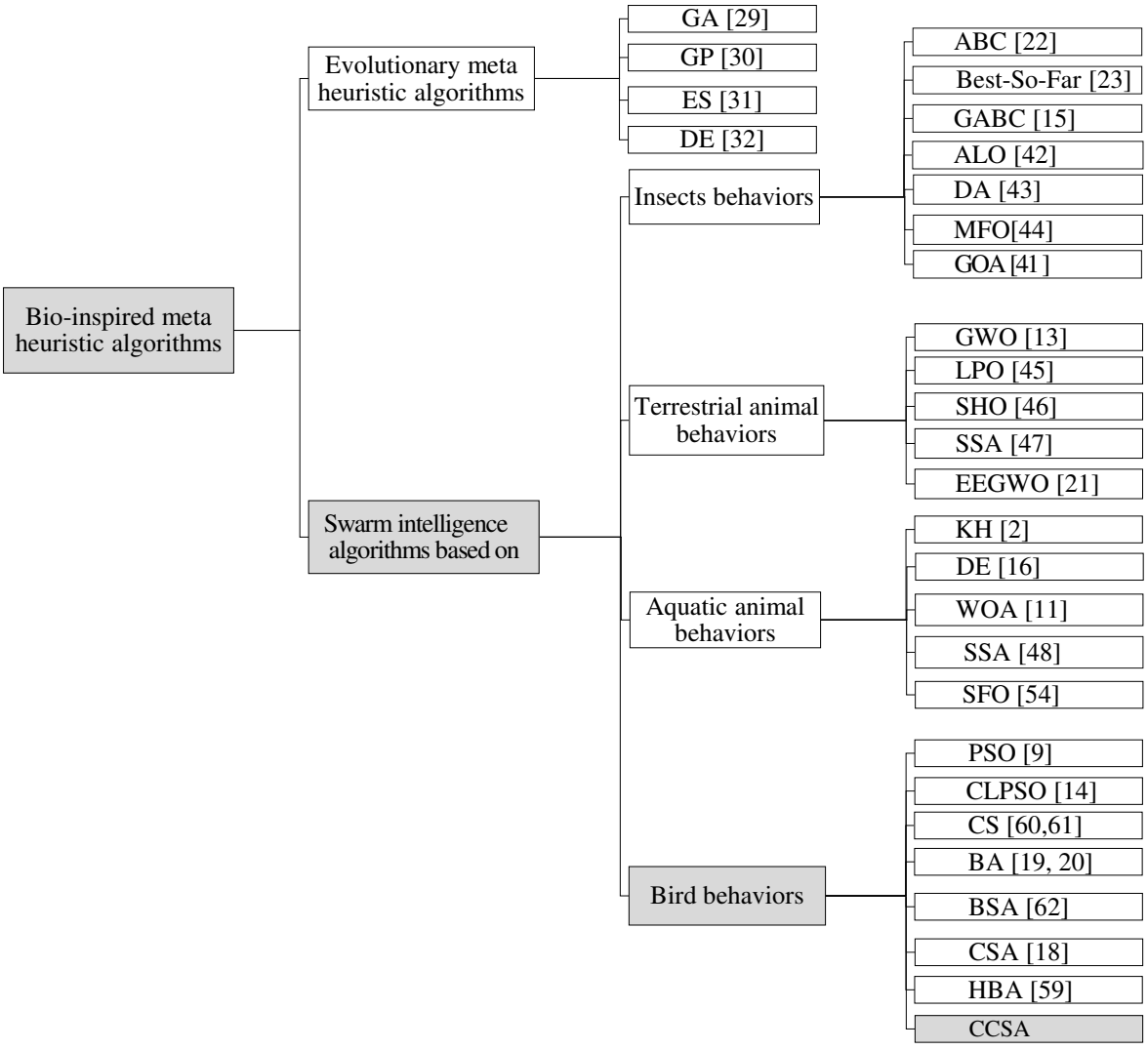


Fig. 1 Classification of bio-inspired metaheuristic algorithms

2.1. Swarm intelligence algorithms based on insects behaviors

Although insects have elementary behaviors and mental structures, self-organization, and cooperation they are regarded as the essential characteristic of their behavior in nature. They can perform several complex tasks in the best way by modeling these simple behaviors of

insects and solve the complex problems. Karaboga et al. [22] proposed the artificial bee colony (ABC) algorithm consist of three groups of bees: onlooker, employed and scout, and three different search strategies: local, global, and random. Although its evaluation results show that ABC is good at global search, its local search is weak, and it suffers from the imbalance between local and global search strategies. Tsai et al. [39] made the first improvement to increase the exploitation capability of the ABC by introducing interactive artificial bee colony (IABC) algorithm. It improves the ABC by utilizing the Newtonian law related to the gravitational force between masses and enhances the relationship of the employed and the onlooker bees. In order to enhance the balance between local and global search in the ABC algorithm, an extended version of this algorithm was proposed named gbest-guided ABC (GABC) [14]. In GABC, the global best (gbest) solution has also been added to the onlooker and employed bees phase to improve local search in this algorithm.

Accordingly, to improve the efficiency of the onlooker bees and to enhance local and global search, Best-so-far ABC algorithm [23] was proposed. In this algorithm, the information obtained from all employed bees is used by onlooker bees to decide about a candidate food source and choose the best-so-far position. Although the Best-so-far ABC algorithm improves the search strategy of the ABC algorithm, its evaluation results show that it still inherently suffers from a lack of balance between local and global search. Karaboga et al. [16] introduced qABC algorithm for enhancing the performance of the ABC algorithm in term of local search ability. They introduce a new control parameter named neighborhood radius by which onlooker bees choose the best food source in the neighborhood of the selected food source. Although their evaluation results show that the better local search ability in qABC algorithm, it suffers from the premature convergence problem. Thus, iqABC [40] was recently proposed by introducing different search schemas by which the qualities of the final solutions and convergence speeds are enhanced.

GOA algorithm was proposed [41] by simulating repulsion and attraction forces between the grasshoppers. Grasshoppers explore the search space using repulsion forces, whereas attraction forces encourage them to exploit the promising regions. GOA is equipped with a coefficient that adaptively decreases the comfort zone of the grasshoppers in order to strike a balance between local and global search. In 2015, Mirjalili introduced the Ant Lion Optimizer (ALO) [42] algorithm that mimics the hunting behavior of antlions using five main steps: the random walk of ants, building traps, entrapment of ants in traps, catching preys, and re-building traps. Its experimental results indicated that ALO benefits from high exploitation and convergence rate.

Dragonfly algorithm (DA) [43] is inspired by the static and dynamic swarming behaviors of dragonflies to explore and exploit the search space. It is equipped with five parameters to control cohesion, alignment, separation, attraction, and distraction of dragonflies in the swarm. Accordingly, the author used suitable operators and proposed binary and multi-objective versions as well. The results showed that all of the versions have high exploration and convergence because of using the static and dynamic swarming behavior of dragonflies respectively. MFO [44] was mainly inspired by the navigation method of moths in nature called transverse orientation. Moths fly in a straight line for long distances by this mechanism, although they are trapped in a winding path around artificial lights. In fact, this spiral convergence was the main inspiration of the MFO. It updates positions and obtains neighboring solutions around the flames to increase the local search ability. Furthermore, each moth assigned a flame to increases global search ability and decreases the probability of local optima.

2.2. Swarm intelligence algorithm based on terrestrial animal behavior

The second category of swarm intelligence algorithms mimics terrestrial animal behaviors such as searching for prey, information sharing, herd leadership, encircling and attacking prey. A well-known instance of this category is the gray wolf optimization algorithm (GWO) [20] inspired by the hunting mechanism and the leadership hierarchy of gray wolves. The evaluation results GWO show that although its local search is efficient, its global search is weak. To enhance the performance of GWO, Long et al. [21] proposed exploration-enhanced GWO (EEGWO) algorithm. It creates a proper balance between local and global search by improving the movement strategy. Also, the evaluation results of EEGWO show more efficiency in high-dimensional problems compared with GWO algorithm.

Lions are another species of these animals whose philosophy of group living and pride behavior in the herd has formed the main idea for introducing the lion pride optimizer (LPO) algorithm [45]. The efficiency of LPO is mostly from herd update strategy and competition among lions. The spotted hyena optimizer (SHO) [46] algorithm is proposed based on the spotted hyena behaviors. SHO uses two phases: searching for prey and attacking prey for exploration and extraction by which it creates a proper balance between local and global search.

Squirrel search algorithm (SSA) [47] is designed for unconstrained optimization problems by imitating of the dynamic foraging behavior of southern flying squirrels and their efficient way of locomotion known as gliding. Although its results showed that SSA finds the global optimum solutions in the low dimension optimization problems with good convergence, it losses the effectiveness in the large scale optimization problems.

2.3. Swarm intelligence algorithm based on aquatic animal behaviors

Behaviors such as movement, prey encircling and mating in aquatic animals play an important role in modeling and creating algorithms of this category. Some well-known algorithms of this category are salp swarm algorithm (SSA) [48], dolphin echolocation (DE) [49], krill herd (KH) [2] and whale optimization algorithm (WOA) [11].

WOA models the social behavior of humpback whales by three phases: encircling prey, bubble-net attacking method and search for prey. Its experimental evaluations show that WOA suffers from an imbalance between global and local search and premature convergence. Because of these weaknesses, WOA is trapped by the local optima. Meanwhile, WOA adapted for solving discrete problems such as feature selection from medical data [50]. Opposition-based learning WOA was proposed [51] to increase the performance of WOA by considering the 'opposite' position of whales for position updating. Moreover, the position of whales has been updated by levy flight and random flights [52, 53] as well by which the exploitation and convergence speed of WOA can be increased.

KH simulates the food searching behavior of krill through the three phases of foraging motion, motion induced by other krill individuals, and random physical diffusion. The local and global search strategies are implemented in two phases of motion induced by other krill individuals and foraging motion. By doing this, the KH algorithm can improve the balance between local and global search.

SailFish Optimizer (SFO) [54] is inspired by the group hunting behavior of sailfish. It simulates the search for prey, attack-alternation strategy, and hunting and catching prey of the hunting behavior of sailfishes. This algorithm uses two kinds of populations: sailfish population for exploitation around the best so far and sardine's population for exploration which results in low local optima stagnation. It has a proper exploration by saving a promising area in each iteration.

2.4. Swarm intelligence algorithm based on bird behaviors

The main idea behind this category is the survival behaviors of birds, such as nesting, mating habits, protection against predators, feeding and interaction with others. Particle swarm optimization (PSO) [9] is a famous instance of this category which is inspired by the social behavior of flocks, shoals of fish and swarms of insects searching for food. The main active elements in particles movements are the current position of the particle, the best personal position, the best group history, and velocity. Although PSO algorithm is efficient in solving unimodal problems, it is trapped in local optima when solving complex multimodal problems.

This is because having poor exploration and imbalance between local and global search. In order to exit from the local optima, Liang et al. proposed the comprehensive learning particle swarm optimization (CLPSO) [13] in which each particle effects from other particles in different dimensions. CAPSO was introduced [55] to enhance the convergence speed and global optimality of PSO. There is still much attention to improve PSO, then recently PSOTD [56], OSC-PSO [57] and CLPSO-LOT [58] were proposed.

Bat algorithm (BA) [18,19] is a combination of the PSO algorithm without the best personal histories along with a local search based on loudness and pulse rate. BA uses echolocation behavior to detect their prey, avoid hitting obstacles and find their nest. These approaches strengthen the local search ability by choosing a new solution based on the best available solutions. To overcome the premature convergence and improve the performance of BA on complex continuous optimization problems, an improved BA (HBA) was proposed by Liu et al. [59]. They developed HBA with three modifications, to improve the performance of BA on complex continuous optimization problems as follows: modification of the initial population to enhancing the diversity of the initial bat population in the search space; modification of location updating to improve the local search capability; and hybridization with extremal optimization for increasing the local search capability and the strengthen ability to tradeoff between local and global abilities. HBA shows the worse performance on some high-dimensional functions with the computation time.

Cuckoo search (CS) [60,61] algorithm is inspired by the life and egg-laying of cuckoo species. Bird swarm algorithm (BSA) [62] is inspired by foraging behavior, vigilance behavior and flight behavior of birds. Crow search algorithm (CSA) [17] is a population-based bio-inspired algorithm inspired by the social behavior of crows. In the next section, CSA is explained in detail.

3. Crow search algorithm

The social behaviors of crows are the main idea of the development of the crow search algorithm (CSA). Crows are smart birds that can hide their excess foods in hiding places and retrieve them when the foods are needed. In CSA, each crow seeks to steal the food resources of other crows, and they predict the behavior of the pilferer crow using their own experience of having been a thief. The awareness of the crow being followed by another crow plays a key role in search strategy selection. Consequently, there is considered an awareness probability (AP) value for each crow, which is compared with a random number r. As the pseudo-code of CSA shown in Fig. 2, then, based on the result of this comparison, two states may happen for the pilferer. In the first state $r_j \geq AP_j(t)$, which means the awareness probability of crow j (AP_j)

in iteration t is less than this random number. In such a case, according to Eq. (1), $\vec{m}_j(t)$ as the hiding place of crow j is stolen by crow i, where $\vec{x}_i(t)$ is the position of the pilferer crow, r_i *and* r_j are the random number with uniform distribution in the interval of [0, 1]. In addition, $fl_i(t)$ is the crow's flight length in the iteration t for creating a suitable balance between local and global search. The low value of fl encourages local search around $\vec{x}_i(t)$, while the high value of this parameter reduces the focus on local search, as the result of which the range of the global search is increased.

$$\vec{x}_i(t+1) = \vec{x}_i(t) + r_i \times fl_i(t) \times (\vec{m}_j(t) - \vec{x}_i(t)) \quad (1)$$

In the second state $r_j < AP_j(t)$, which means crow j is aware of being followed by the crow i. In this state, according to Eq. (2), the crow i is moved to a new random position in the search space.

$$\vec{x}_i(t+1) = \text{a random position of search space} \quad (2)$$

Like PSO, CSA explores the search space using a population of N crows (particles) which increase the probability of finding a near optimal solution. Although PSO uses the personal and global best solutions to increases the probability of convergence and finding a better solution [13,44], CSA selects randomly another crow (it may be itself) and moves towards its best position. This strategy increases the probability of escaping from the local optima and the diversity of generated solutions. Meanwhile, in CSA contrary to PSO only two main parameters flight length (fl) and awareness probability (AP) must be adjusted and tuned.

The evaluation results of CSA show that its efficiency is not suitable for solving multi-modal, hybrid, composition and large-scale problems. In fact, it suffers from lacking a proper balance between local and global search and having premature convergence, especially in large-scale problems. Mostly, this is because it selects search strategy unconsciously by comparing AP value with a random number. In addition, the parameter fl is manually set for all iterations by a constant value, which affects determining the flight length and striking a balance. Section 4 describes how the proposed CCSA tackles these weaknesses.

Algorithm 1 Crow search algorithm

Input: Population size N and the number of iteration MaxIt.
Output: Optimal hiding place of crow m, best fitness value fbest.

procedure CSA
Randomly initialize the position of a flock of N crows in the search space.
Evaluate the position of the crows.
Initialize the memory of each crow.
While t < MaxIt
For each search crow **do**
Randomly choose one of the crows to follow.
Define an awareness probability.
If $r_j \geq AP_j(t)$
$\vec{x}_i(t+1) = \vec{x}_i(t) + r_i \times fl_i(t) \times \left(\vec{m}_j(t) - \vec{x}_i(t)\right)$.
Else
$\vec{x}_i(t+1)$ = a random position of search space.
end if
end for
Check the feasibility of new positions.
Evaluate the fitness value of new positions.
Update the memory of crows.
end while
Return fitness value and the position of the best crow.
end procedure

Fig. 2 Pseudo code of CSA [17]

4. Conscious neighborhood-based crow search algorithm (CCSA)

The flowchart of the proposed CCSA is shown in Fig. 3. In this flowchart, initially, a finite set of $C = \{c_1, c_2, \dots, c_N\}$ consisting of N crows are distributed by uniform random distribution in D-dimensional problem space. Each crow c_i in the current iteration t is considered by a 4-tuple $c_i(t) = < \vec{x}_i(t), f_i(t), \vec{m}_i(t), fbest_i(t)$. In this 4-tuple, the vector $\vec{x}_i(t) = [x_{i1}, x_{i2}, \dots, x_{iD}]$ and $f_i(t)$ are the position and the fitness value of crow c_i in the iteration t respectively. Then, $fbest_i(t)$ is the best fitness value of crow c_i till the iteration t and the vector $\vec{m}_i(t) = [m_{i1}, m_{i2}, \dots, m_{iD}]$ represents the position where the crow c_i has obtained the $fbest_i$. In addition, by inspiring the intelligent behavior of crows and based on Definition 1, a hiding place m_i is considered by a 2-tuple $(\vec{m}_i, fbest_i)$ for crow c_i.

Definition 1 (hiding place of crow): Consider $m_i = (\vec{m}_i, fbest_i)$ is the hiding place of crow c_i. Then, in the first iteration, $\vec{m}_i \leftarrow \vec{x}_i(t = 1)$ and for the other iterations, $\vec{m}_i \leftarrow \vec{x}_i(s)$ such that $F(\vec{x}_i(s)) = \min\{F(\vec{x}_i(k)), k = 2, \dots, t\}$ where $F(\vec{x}_i(k))$ is the fitness function to evaluate the position of crow c_i in iteration k.

In CCSA we introduce three new strategies called Neighborhood-based Local Search (NLS), Non-Neighborhood based Global Search (NGS) and Wandering Around based Search (WAS) in order to enhance the efficiency of movement of crows in different problems space. Moreover, it uses a new conscious neighborhood concept defined by Definition 2 to select the

movement strategy between NLS or NGS consciously, and improve the balance between local and global search. For doing this, as shown in Fig. 3, first, CCSA generates neighborhood of crow c_i by Definition 2. Then, instead of CSA, CCSA selects either NLS or NGS consciously by comparing the fitness value of its neighbors with non-neighbors.

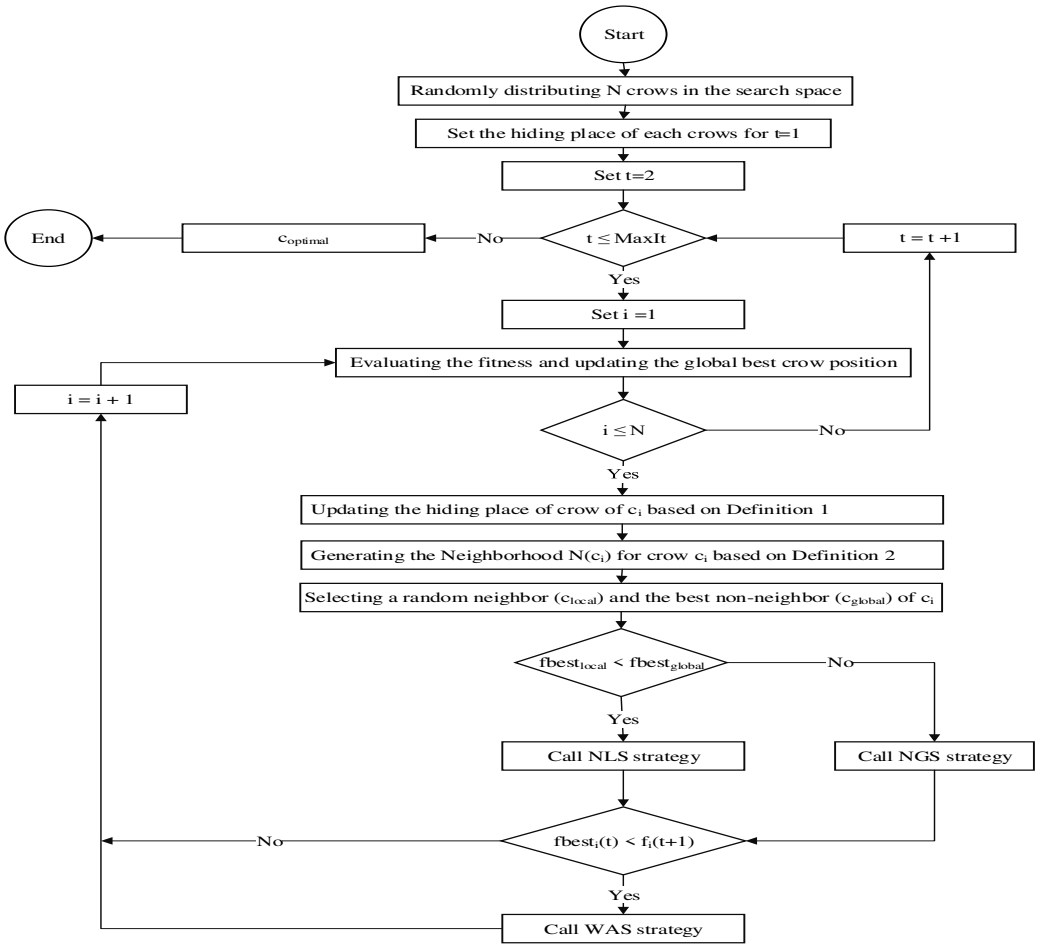


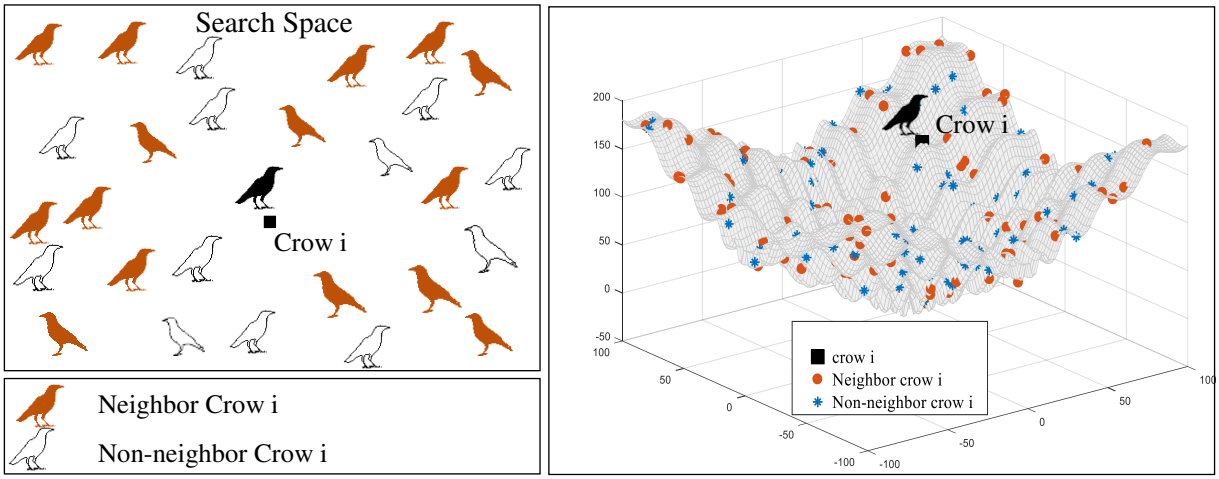
Fig. 3 Flowchart of the proposed CCSA

Definition 2 (conscious neighborhood): Given C is a set of N crows distributed in the problem space and crow $c_i \in C$. Then, a conscious neighborhood of crow c_i or N (c_i) is a subset Y of c_j that $d(\vec{x}_i,\vec{m}_j) \times w_{ij} < \mu$ for j=1,.., N and j≠i and accordingly, subset C-Y is non-neighborhood of c_i. In this definition, $d(\vec{x}_i,\vec{m}_j)$ is the Euclidean distance between position crow c_i and hiding place of crow c_j. Besides, w_{ij} and μ are computed by Eqs. 3 and 4 where f_i and $fbest_j$ are the fitness of position crow c_i and $\vec{m}_j$ respectively.

$$w_{ij} = \frac{\varepsilon + (f_i - fbest_j)}{\sum_{k=1}^{N}(f_i - fbest_k)} \quad (3)$$

$$\mu = \frac{1}{N}\sum_{j=1}^{N}(d(\vec{x}_i,\vec{m}_j) \times w_{ij}) \quad (4)$$

The conscious neighborhood concept is illustrated by Fig. 4 where Fig. 4(a) shows it in 2D graphical representation and Fig. 4(b) demonstrates this concept computed by Eqs. 3 and 4 in a real problem.



(a) 2D graphical representation (b) Real graphical representation

Fig. 4 Graphical representation of the neighborhood concept

4.1. Neighborhood-based local search (NLS) strategy

After generating the neighborhood of crow c_i, it should be exploited if the quality of the neighborhood is good. Thus, the NLS strategy is to improve the exploitation based on the conscious neighborhood as follows. A crow among the neighbor of crow c_i is randomly selected named c_{local}, and the best crow among non-neighbor with the lowest fitness is selected named c_{global}. Given, $fbest_{local}$ and $fbest_{global}$ are the best fitness of selected crows c_{local} and c_{global} respectively, if $fbest_{local} < fbest_{global}$, then c_i will approach to the hiding place of c_{local} and the new position $\vec{x}_i(t+1)$ of c_i is obtained by Eq.5. In fact, the neighbors can be considered as candidate solutions and be exploited.

$$\vec{x}_i(t+1) = \vec{x}_i(t) + r_i \times fl_i(t) \times (\vec{m}_{local}(t) - \vec{x}_i(t)) \quad (5)$$

Where $fl_i(t)$ is the flight length of the crow c_i in the current iteration t which is a decreased linearly variable, r_i is a random number with uniform distribution in an interval [0, 1], and $\vec{m}_{local}(t)$ is the hiding place of c_{local} in the current iteration. Fig. 5 illustrates the graphical representation of the NLS strategy, and the pseudo-code of the NLS is shown in Fig. 6. Both NLS and NGS uses a function named CC ($\vec{x}_i$ (t+1)) to check the newly obtained position of

crow i and correct it if necessary. If the newly position value is out of the problem space, then it resets the exceeded dimensions randomly within the problem space.

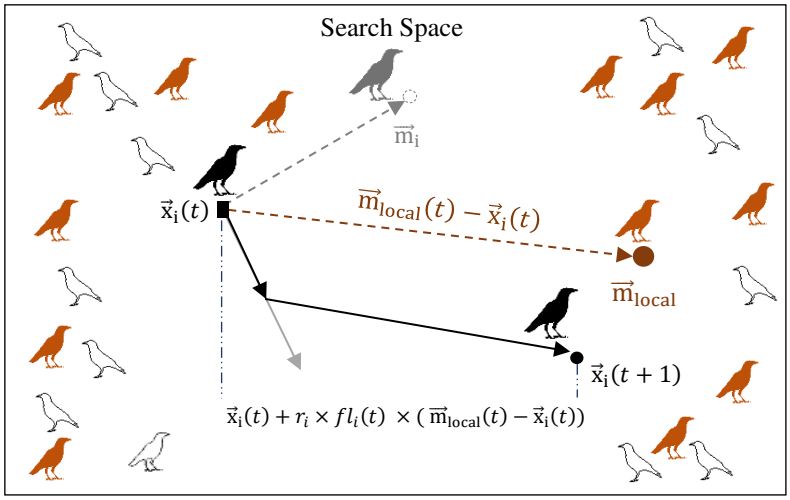


Fig. 5 Graphical representation of NLS strategy

Algorithm 2: Neighborhood-based local search (NLS)

Input: $\vec{x}_i$ (position of crow c_i) and $\vec{x}_{local}$ (position of crow c_{local}).
Output: the new position of crow c_i.
1: **Procedure NLS**
2: $\vec{x}_i(t+1) = \vec{x}_i(t) + r_i \times fl_i(t) \times (\vec{m}_{local}(t) - \vec{x}_i(t))$.
3: CC ($\vec{x}_i$ (t+1)).
4: **Return** the new position of crow c_i.
5: **end procedure**

Fig. 6 Pseudo-code of NLS strategy

4.2. Non-neighborhood based global search (NGS) strategy

CCSA uses NGS to improve the exploration and robustness based on the conscious neighborhood as follows. When the fitness value of $fbest_{local}$ is greater than and equal with $fbest_{global}$. Then, crow c_i extends its range of exploration into its non-neighborhood towards the c_{global} by selecting k dimensions randomly and changing their values using Eq. (6).

$$x_{ij}(t+1) = r_i \times fl_i(t) \times \left(m_{globalj}(t) - x_{ij}(t)\right) \quad (6)$$

Where, $x_{ij}(t+1)$ and $x_{ij}(t)$ are the values of dimension j of the new and current position of crow c_i respectively. Meanwhile, r_i is a random number with a uniform distribution between

0 and 1, $m_{globalj}$ is the value of dimension j of hiding place of c_{global}. The graphical representation and pseudo-code of NGS strategy is shown in Figs. 7 and 8 respectively.

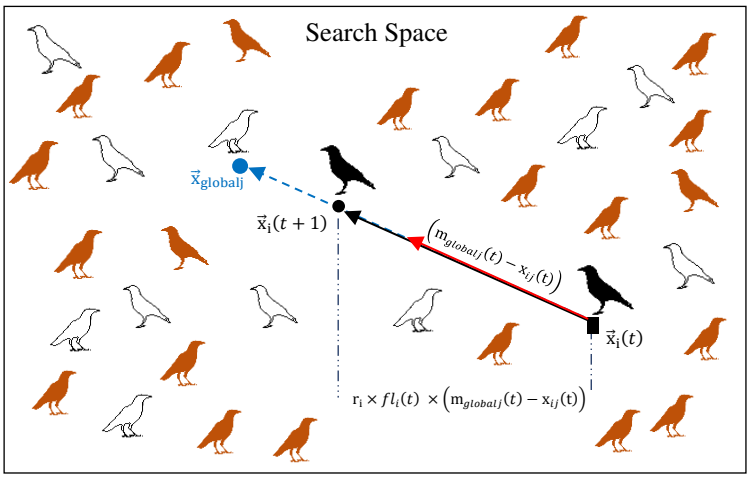


Fig. 7 Graphical representation of NGS strategy

	Algorithm 3: Non-neighborhood based global search (NGS)
	Input: $\vec{x}_i$ (position crow c_i) and $\vec{x}_{global}$ (position of crow c_{global}).
	Output: the new position of crow c_i.
1:	**Procedure NGS**
2:	Selecting k dimensions of $\vec{x}_i(t)$ randomly.
3:	**For** j = 1: k
4:	$x_{ij}(t+1) = r_i \times fl_i(t) \times \left(m_{globalj}(t) - x_{ij}(t)\right)$.
5:	**end for**
6:	CC ($\vec{x}_i$ (t+1)).
7:	**Return** the new position of crow c_i.
8:	**end procedure**

Fig. 8 Pseudo code of NGS strategy

4.3. Wandering around based search (WAS) strategy

CCSA introduces WAS based on an intelligent behavior of crows by which they analyze their surrounding environment and move to a better position by a number of jumps if their position is not good. In fact, the reason for using WAS is to correct the position of those crows that could not acquire a better fitness by using NLS or NGS. If the newly fitness of crow i is not less than the fitness of its hiding place, first, the position of the crow is set with its hiding place position ($\vec{x}_i(t) \leftarrow \vec{m}_i$). Then, crow i explore the search space by doing a random number of jumps (NJ) such that in each jump, k dimensions are randomly selected and their values are changed using Eq. (7).

$$x_{ij}(t+1) = m_{gbestj}(t) + r_i \times fl_i(t) \times (x_{rj}(t) - x_{ij}(t)) \quad (7)$$

Where, $x_{ij}(t+1)$ is the value of dimension j in the new position of crow i, $m_{gbestj}(t)$ is the value of dimension j of the best hiding place in the total population and $x_{rj}(t)$ is the dimension j of a random crow. The graphical representation and pseudo-code of the WAS strategy are shown in Figs. 9 and 10 respectively.

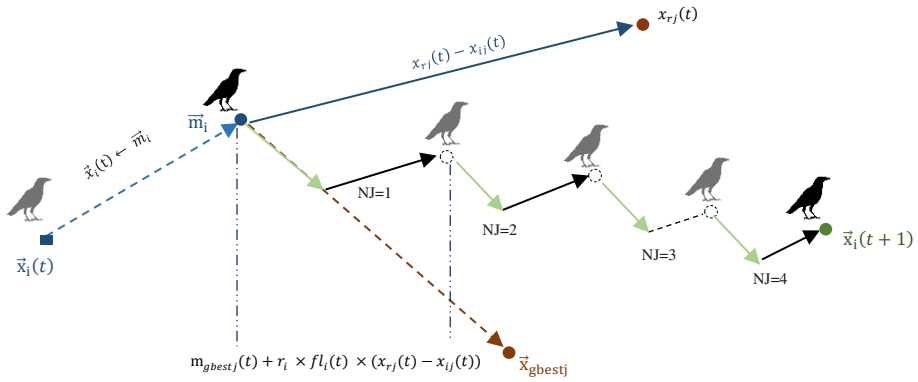


Fig. 9 Graphical representation of WAS strategy using four jumps

Algorithm 4: Wandering around based search (WAS)

Input: $\vec{x}_i$ (position crow c_i), $\vec{x}_r$ (random position crow c_r, $r \neq i$) and $\vec{x}_{gbest}$ (position of the best global crow).
Output: the new position of crow ci.
Procedure WAS
  $\vec{x}_i(t) = \vec{m}_i$.
  NJ = number of jumps.
  For f = 1:NJ
    Selecting k dimensions of $\vec{x}_i(t)$ randomly.
    For j= 1:k
      $x_{ij}(t+1) = m_{gbestj}(t) + r_i \times fl_i(t) \times (x_{rj}(t) - x_{ij}(t))$
    end for
    CC ($\vec{x}_i$ (t+1)).
  end for
  Return the new position of crow c_i.
end procedure

Fig. 10 Pseudo code of WAS strategy

The pseudo-code of CCSA is shown in Fig. 11 and its parameters description are presented in Table 1.

Algorithm 5: Conscious neighborhood-based crow search algorithm (CCSA)

Input: N (population size), D (dimension of search space) and MaxIt (the number of iteration).
Output: $c_{optimal}$ (optimal solution) with position $x_{optimal}=\{x_1, x_2, x_3, \ldots, x_D\}$.
Begin

1: Randomly distributing N crows in the search space.
2: Set the hiding place of each crow.
3: **For** t =2:MaxIt
4: **For** i = 1:N
5: Evaluating the fitness f ($\vec{x}_i(t)$) and updating the global best crow position.
6: Updating the hiding place of crow of c_i based on Definition 1.
7: Generating the neighborhood N (c_i) for crow c_i based on Definition 2.
8: Selecting a random neighbor (c_{local}) and the best non-neighbor (c_{global}) of c_i.
9: **If** $fbest_{local} < fbest_{global}$
10: $\vec{x}_i(t+1)$ = NLS ($\vec{x}_i(t), \vec{x}_{local}(t)$).
11: **else**
12: $\vec{x}_i(t+1)$ = NGS ($\vec{x}_i(t), \vec{x}_{global}(t)$).
13: **end if**
14: **If** $fbest_i(t) < f_i(t+1)$
15: $\vec{x}_r$ = Randomly generated crow c_r, $r \neq i$.
16: $\vec{x}_i(t+1)$ = WAS ($\vec{x}_i(t)$, $\vec{x}_r(t), \vec{x}_{gbest}(t)$).
17: **end if**
18: **end for**
19: **end for**
20: **Return** $c_{optimal}$
21: **end procedure**

Fig. 11 The pseudo code of CCSA

Table 1 Nomenclature used in CCSA

Parameters	Description
N	Population size.
c_i	Crow c_i.
t and MaxIt	Current and maximum number of iterations.
$\vec{x}_i(t)$ and $\vec{m}_i(t)$	Current position of crow c_i and its hiding place position.
$f_i(t)$ and $fbest_i$	Fitness value of current and hiding place positions of crow c_i.
$\vec{x}_r(t)$	A random crow position.
$\vec{x}_{gbest}(t)$	The position of the best global crow in all population.
$fl_i(t)$	Flight length of crow c_i in iteration t.
r_i	Random number with uniform distribution in the interval of [0, 1].
NJ	The number of jumps.
$N(c_i)$	The neighborhood of crow c_i.

4.4. Computational complexity analysis

As Fig. 11 shown CCSA algorithm consists of two main phases: initialization (lines 1-2) and movement (lines 3-19). The computational complexity of the initialization phase for distributing N crows in the search space with D dimensions is O(ND). Meanwhile, in the movement phase, the computational complexity of generating the neighborhood of each crow (line 7), using either NLS or NGS (lines 9-13) and then WAS (lines 14-17) is O(ND+D+D). Thus, the computational complexity of the movement phase for N crows for all iterations (T) is O(TN(ND+D+D)) which results the overall computational complexity of CCSA is equal to $O(TN^2D)$. Although there are two similar initialization and movement phases in the CSA algorithm, the computational complexity of the CSA algorithm is O(TND). This is because, as

Fig. 2 shown, the computational burden of its initialization (lines 2-4) and movement (line 5-18) phases is O(ND) and O(TND) respectively. In the next section, experimental results verify the benefits of the additional cost caused by the introduced conscious neighborhood.

5. Experimental evaluation and results

Since no metaheuristic algorithm can solve all optimization problems, in this section, the numerical efficiency of the proposed CCSA was experimentally evaluated by several experiments. First, CCSA was evaluated for solving global optimization problems by benchmark functions CEC 2017 [24] with different dimensions of 30, 50 and 100. Then, the efficiency of CCSA was evaluated for solving large-scale global optimization problems by benchmark functions CEC 2010 [25] with dimensions 1000. In all experiments, the efficiency of CCSA was compared with CSA and some of the state-of-the-art swarm intelligence algorithms: BA [18], CLPSO [13], GWO [20], EEGWO [21], WOA [11], KH [2], ABC[22], GABC [14] and Best-so-far ABC [23], which were named compared algorithms. Finally, CCSA and these algorithms were statistically compared by Mean Absolute Error (MAE) and Friedman tests.

5.1. Benchmark functions

To evaluate the numerical efficiency of CCSA, the benchmark functions CEC 2017 was used. CEC 2017 consists of four groups of different functions: unimodal, simple multi-modal, hybrid and composition. This diversity of functions is suitable to compare the proposed CCSA with the compared algorithms in terms of its ability to escape from local optima, convergence behavior, local search, and global search.

In this benchmark, the first group consists of three unimodal functions F_1-F_3, which have only a global optimization, and they are a unimodal, non-separable, symmetric, and smooth but narrow ridge. Thus, they are suitable for evaluating the ability of algorithms in terms of local search and convergence speed. In the second group, there are seven multi-modal functions F_4-F_{10}, which have a large number of local optima. These functions have the required characteristics to test the global search capabilities of the proposed CCSA. The third and fourth groups consist of ten hybrid functions F_{11}-F_{20}, and ten composition functions F_{21}-F_{30} respectively. Due to possessing the characteristic of maintaining the continuity around the local and global optima, their functions can evaluate the proposed algorithm in terms of the balance between local and global search and premature convergence.

Many efficient metaheuristic algorithms often lose their efficiency when applied to large-scale problems. They suffer from the "curse of dimensionality", by which their performance decreases dramatically as the dimensionality of the search space increases. Therefore, the proposed algorithm was also evaluated for solving the large-scale global optimization problems conducted by benchmark functions CEC 2010 [25] with dimensions 1000. This benchmark consists of twenty benchmark functions with different properties: unimodal, multi-modal, shifted, separable, fully nonseparable and scalability in the different range space.

5.2. Experimental setup

The MATLAB 2014b programming language was used to implement CCSA and the compared algorithms. To make sure that the comparison is fair, all these algorithms were run under the same conditions on the pc with Intel Core i7, 3.4 GHz CPU and 8 GB memory in windows 7.

The efficiency of CCSA was evaluated for solving different problems unimodal, simple multi-modal, hybrid and composition of the benchmark functions CEC 2017. The values of the common parameters such as population size (N) and the maximum number of iterations (MaxIt) were set to 200 and1500 respectively. Due to the random nature of the metaheuristic algorithms, the algorithms in all experiments of the efficiency evaluation and statistical tests were run 30 times for each function in different dimension 30, 50 and 100. The efficiency was computed by average (Avg), standard deviation (SD) and minimum (Min) of the best obtained optimal solution until the last iteration in each run. In the proposed CCSA, a few parameters must be adjusted. The flight length fl is linearly decreased from 2 to 0.9, ε is set by 0.02 and based on the results of our experiments the number of jumps (NJ) can be randomly set between 1 and 50.

The required parameters of the compared algorithms were set as same as their original algorithms as follows. In ABC, the limit control parameter was set to 100. Since this value can change depending on the dimension of search space, thus it was set to N×D for Best-so-far ABC and GABC. In CLPSO, the cognitive (c_1) and social components (c_2) were set to 1.49445, and the inertia weight was considered between 0.2 and 0.9. In BA, Pulse rate (r) and Loudness (A) were set to 0.5. Moreover, the minimum and maximum frequency were set to 0 and 2 respectively. In KH algorithm, the maximum diffusion speed, velocity of foraging and maximum induced speed were set to 0.005, 0.02 and 0.01 respectively. Consistently, the inertia weights of the motion and foraging motion were linearly decreased from 0.9 to 0.1 and C_t was set to 0.5 by which the krill individuals can search the problem space carefully. In the

exploration and exploitation phases of WOA, control parameter $\vec{a}$ was linearly decreased from 2 to 0 over iterations and shape of the logarithmic spiral b was set to 1. In EEGWO, the constant coefficient values b_1 and b_2 were set to 0.1 and 0.9 respectively. Also, non-linear modulation index μ and an initial and a final were set to 1.5, 2 and 0 respectively. In CSA, as the original work, AP= 0.1 and fl=2.

5.3. Numerical efficiency evaluation

In this section, through various experiments conducted by different functions of the benchmark CEC 2017, the efficiency of CCSA was evaluated. Its results were compared with the compared algorithms concerning local and global search ability and escape from local optima. The experimental results are shown in Tables 2-5 in which the bold values show the winning algorithms or the best solutions. Moreover, at the end of each table, the comparison of the results by the numbers of win (W), tie (T) and loss (L) of each algorithm is shown.

5.3.1. Evaluation of local searchability

Due to the properties of functions F_1-F_3, they can evaluate the local search ability of CCSA and other compared algorithms. The results of local search evaluation for different dimensions 30, 50 and 100 are shown in Table 2. The experimental results show CCSA is superior to the compared algorithms on functions F_1 and F_2 with dimensions 30 and 50 and on all functions with dimension 100.

5.3.2. Evaluation of global searchability

The functions F_4-F_{10} as simple multi-modal functions in CEC 2017 benchmark have a global optimization and several local optima. In addition, the number of local optima is exponentially increased by increasing the dimensions. Then, these functions are a good criterion for evaluating global search ability of optimization algorithms. The efficiency of the proposed algorithm for these functions was evaluated in three separate experiments for different dimensions 30, 50 and 100. The experimental results of proposed CCSA and other compared algorithms are shown in Table 3. The results show that CCSA is superior to the compared algorithms on all functions F_4-F_{10} for all three different dimensions 30, 50 and 100.

In addition, the efficiency of CCSA and compared algorithms for hybrid functions F_{11}-F_{20} were evaluated and compared by the different dimensions. The results of these experiments are shown in Table 4. In dimension 30, CCSA had a better efficiency for nine of ten hybrid

functions, and it is more efficient than the compared algorithms for dimensions 50 and 100 on all functions.

Table 2 Comparison of optimization results obtained from the unimodal benchmark functions

F	D	Index	CCSA	CSA	BA	CLPSO	GWO	EEGWO	WOA	KH	ABC	GABC	Best-so-far ABC
F_1	30	Avg	2.515E+03	2.542E+03	1.717E+10	6.169E+05	8.871E+08	5.726E+10	2.480E+06	1.716E+09	4.348E+02	1.464E+03	6.766E+03
		SD	1.957E+03	1.874E+03	6.206E+09	1.513E+05	8.705E+08	3.622E+09	1.451E+06	9.407E+09	2.853E+02	1.094E+03	1.370E+03
		Min	**1.018E+02**	2.666E+02	5.995E+09	4.434E+05	4.421E+07	4.982E+10	4.824E+05	7.541E+03	1.815E+02	1.177E+02	3.568E+03
	50	Avg	1.222E+03	3.554E+05	5.583E+10	7.296E+07	3.627E+09	1.101E+11	5.272E+07	2.406E+06	2.398E+04	2.757E+04	9.087E+06
		SD	1.028E+03	1.192E+05	1.105E+10	1.238E+07	2.726E+09	4.755E+09	2.874E+07	2.558E+06	1.679E+04	2.918E+04	2.35E+06
		Min	**2.148E+02**	1.995E+05	2.481E+10	4.816E+07	1.745E+08	1.011E+11	9.504E+06	8.774E+04	1.654E+03	1.418E+03	5.070E+06
	100	Avg	7.003E+03	3.477E+08	1.811E+11	3.937E+09	2.481E+10	2.671E+11	1.753E+09	1.096E+10	4.520E+05	3.095E+05	1.676E+09
		SD	2.481E+03	7.768E+07	3.245E+10	4.341E+08	7.258E+09	6.997E+09	4.974E+08	3.601E+09	4.450E+05	2.460E+05	3.011E+08
		Min	**3.235E+03**	1.980E+08	9.765E+10	2.842E+09	1.413E+10	2.449E+11	8.725E+08	4.183E+09	4.895E+04	2.541E+04	1.016E+09
F_2	30	Avg	2.042E+02	3.571E+13	4.878E+45	1.337E+21	3.506E+27	1.018E+48	1.696E+24	1.272E+41	2.079E+11	1.125E+12	7.869E+15
		SD	1.294E+01	8.456E+13	1.618E+46	3.373E+21	1.203E+28	2.601E+48	5.332E+24	3.984E+41	3.826E+11	5.214E+12	1.520E+16
		Min	**2.000E+02**	5.852E+10	4.139E+36	1.0478E+18	1.054E+15	8.862E+42	6.743E+17	2.627E+21	2.651E+07	6.510E+06	4.829E+12
	50	Avg	4.134E+02	5.076E+31	2.463E+82	4.282E+48	4.390E+50	1.214E+82	1.367E+64	1.259E+78	2.936E+28	1.867E+28	8.621E+39
		SD	1.759E+02	2.190E+32	1.340E+83	7.856E+48	2.292E+51	4.546E+82	6.210E+64	6.319E+78	7.792E+28	5.679E+28	3.340E+40
		Min	**2.990E+02**	1.909E+23	2.151E+68	5.953E+45	4.928E+38	4.882E+71	1.866E+46	6.889E+54	1.059E+24	6.517E+17	5.257E+33
	100	Avg	7.921E+19	2.322E+101	5.545E+177	4.708E+132	5.569E+120	1.391E+176	2.299E+164	1.956E+176	1.575E+93	4.725E+93	5.756E+113
		SD	2.690E+20	8.315E+101	INF	1.276E+133	1.586E+121	6.343E+164	INF	INF	8.126E+93	2.558E+94	1.755E+114
		Min	**3.406E+09**	1.319E+86	1.414E+156	1.124E+123	2.351E+99	INF	2.603E+131	1.000E+152	3.198E+71	1.901E+63	1.798E+106
F_3	30	Avg	1.969E+03	4.749E+02	1.352E+05	7.230E+04	2.504E+04	8.798E+04	1.455E+05	4.477E+04	1.098E+05	1.192E+05	1.297E+05
		SD	7.909E+02	1.141E+02	8.637E+04	1.055E+04	7.862E+03	3.742E+03	6.015E+04	1.912E+04	1.549E+04	1.803E+04	1.429E+04
		Min	5.635E+02	**3.297E+02**	3.346E+04	5.300E+04	5.637E+03	7.771E+04	2.831E+04	2.205E+04	8.322E+04	6.766E+04	8.080E+04
	50	Avg	2.461E+04	1.235E+04	3.674E+05	1.991E+05	6.951E+04	3.659E+05	8.681E+04	1.190E+05	2.171E+05	2.465E+05	2.542E+05
		SD	5.441E+03	3.247E+03	3.094E+05	2.634E+04	1.521E+04	2.986E+05	4.003E+04	1.906E+04	2.053E+04	2.226E+04	3.065E+04
		Min	1.113E+04	**7.296E+03**	6.106E+04	1.471E+05	4.263E+04	1.802E+05	4.522E+04	8.715E+04	1.791E+05	1.976E+05	1.776E+05
	100	Avg	1.264E+05	1.303E+05	7.612E+05	5.638E+05	2.163E+05	3.881E+05	7.593E+05	3.742E+05	5.696E+05	6.239E+05	6.463E+05
		SD	1.865E+04	1.280E+04	4.109E+05	5.417E+04	2.713E+04	4.770E+04	2.013E+05	4.147E+04	3.621E+04	3.561E+04	4.354E+04
		Min	**9.528E+04**	1.034E+05	1.815E+05	4.470E+05	1.749E+05	3.271E+05	3.081E+05	3.147E+05	4.933E+05	5.323E+05	5.671E+05
Ranking	**30 W|T|L**		2|0|1	1|0|2	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3
	50 W|T|L		2|0|1	1|0|2	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3
	100 W|T|L		3|0|0	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3	0|0|3

5.3.3. Ability evaluation to escape from local minima

The ability of an optimization algorithm to exit from the local minima depends significantly on balance between local and global search. Then, the power of CCSA for balancing between local and global search was evaluated by composite benchmark functions. Table 5 shows the experimental results of this evaluation compared to other algorithms. As the results show, in the most functions for different dimensions, the ability of CCSA to escape from local minima is better than the compared algorithms.

Table 3 Comparison of optimization results obtained from the simple multi-modal benchmark functions

F	D	Index	CCSA	CSA	BA	CLPSO	GWO	EEGWO	WOA	KH	ABC	GABC	Best-so-far ABC
F_4	30	Avg	4.152E+02	5.023E+02	3.441E+03	5.294E+02	5.360E+02	1.725E+04	5.446E+02	5.046E+02	4.310E+02	4.490E+02	4.703E+02
		SD	2.122E+01	3.100E+01	1.480E+03	1.578E+01	2.866E+01	2.190E+03	3.373E+01	1.534E+01	2.310E+01	2.377E+01	1.583E+01
		Min	**4.000E+02**	4.172E+02	1.282E+03	4.628E+02	4.802E+02	1.366E+04	4.957E+02	4.748E+02	4.033E+02	4.080E+02	4.274E+02
	50	Avg	4.302E+02	6.302E+02	1.260E+04	7.530E+02	7.985E+02	3.717E+04	7.480E+02	5.811E+02	4.816E+02	4.735E+02	5.923E+02
		SD	1.593E+01	4.836E+01	3.802E+03	2.470E+01	1.424E+02	3.292E+03	6.975E+01	4.505E+0	2.283E+01	2.545E+01	2.099E+01
		Min	**4.012E+02**	5.000E+02	4.964E+03	7.041E+02	5.946E+02	2.928E+04	6.307E+02	4.919E+02	4.344E+02	4.369E+02	5.622E+02
	100	Avg	5.906E+02	1.198E+03	4.777E+04	2.050E+03	2.487E+03	1.076E+05	1.589E+03	2.425E+03	8.103E+02	7.482E+02	1.276E+03
		SD	1.842E+01	1.212E+02	1.316E+04	1.100E+02	5.815E+02	7.951E+03	2.223E+02	6.720E+02	5.067E+01	2.955E+01	9.792E+01
		Min	**5.513E+02**	1.012E+03	2.095E+04	1.810E+03	1.636E+03	8.855E+04	1.221E+03	1.557E+03	7.172E+02	6.953E+02	1.033E+03
F_5	30	Avg	5.311E+02	6.159E+02	7.267E+02	5.958E+02	5.869E+02	9.595E+02	7.783E+02	6.440E+02	5.848E+02	5.572E+02	5.890E+02
		SD	7.247E+00	2.400E+01	3.623E+01	7.863E+00	3.560E+01	2.085E+01	6.196E+01	7.491E+01	1.020E+01	7.316E+00	1.098E+01
		Min	**5.131E+02**	5.756E+02	6.343E+02	5.766E+02	5.484E+02	9.084E+02	6.540E+02	5.995E+02	5.625E+02	5.433E+02	5.594E+02
	50	Avg	5.790E+02	7.454E+02	8.730E+02	7.781E+02	6.732E+02	1.230E+03	9.388E+02	7.534E+02	7.137E+02	6.565E+02	7.358E+02
		SD	1.215E+01	4.017E+01	5.720E+01	1.442E+01	3.461E+01	1.887E+01	6.987E+01	2.948E+01	1.751E+01	1.921E+01	2.127E+01
		Min	**5.488E+02**	6.711E+02	7.776E+02	7.420E+02	5.970E+02	1.177E+03	8.168E+02	6.981E+02	6.693E+02	6.098E+02	6.914E+02
	100	Avg	7.765E+02	1.194E+03	1.412E+03	1.471E+03	1.033E+03	2.161E+03	1.529E+03	1.203E+03	1.288E+03	1.149E+03	1.353E+03
		SD	2.850E+01	6.247E+01	9.671E+01	3.265E+01	4.555E+01	2.386E+01	1.622E+02	5.674E+01	4.116E+01	4.212E+01	4.329E+01
		Min	**7.260E+02**	1.041E+03	1.248E+03	1.403E+03	9.358E+02	2.095E+03	1.228E+03	1.109E+03	1.211E+03	1.035E+03	1.259E+03
F_6	30	Avg	6.000E+02	6.203E+02	6.581E+02	6.003E+02	6.029E+02	7.013E+02	6.680E+02	6.386E+02	6.000E+02	6.000E+02	6.001E+02
		SD	3.070E-03	6.125E+00	8.323E+00	4.436E-02	2.024E+00	4.267E+00	1.068E+01	6.705E+00	1.271E-04	1.265E-05	2.667E-02
		Min	**6.000E+02**	6.069E+02	6.428E+02	6.002E+02	6.007E+02	6.898E+02	6.480E+02	6.245E+02	**6.000E+02**	**6.000E+02**	6.001E+02
	50	Avg	6.000E+02	6.333E+02	6.617E+02	6.029E+02	6.088E+02	7.123E+02	6.799E+02	6.508E+02	6.001E+02	6.000E+02	6.019E+02
		SD	3.514E-03	6.161E+00	6.233E+00	2.720E-01	2.636E+00	4.097E+00	1.036E+01	4.513E+00	1.853E-02	3.290E-03	2.272E-01
		Min	**6.000E+02**	6.218E+02	6.475E+02	6.023E+02	6.047E+02	7.026E+02	6.639E+02	6.419E+02	6.001E+02	**6.000E+02**	6.015E+02
	100	Avg	6.000E+02	6.520E+02	6.658E+02	6.178E+02	6.245E+02	7.167E+02	6.847E+02	6.607E+02	6.028E+02	6.018E+02	6.140E+02
		SD	2.495E-03	5.739E+00	3.272E+00	9.061E-01	3.803E+00	3.143E+00	8.585E+00	3.549E+00	5.942E-01	8.392E-02	9.203E-01
		Min	**6.000E+02**	6.427E+02	6.594E+02	6.157E+02	6.166E+02	7.085E+02	6.727E+02	6.557E+02	6.016E+02	**6.017E+02**	6.119E+02
F_7	30	Avg	7.629E+02	8.241E+02	1.286E+03	8.421E+02	8.222E+02	1.454E+03	1.215E+03	8.273E+02	8.068E+02	7.826E+02	8.104E+02
		SD	5.783E+00	2.991E+01	1.099E+02	7.982E+00	2.433E+01	3.998E+01	9.981E+01	2.636E+01	8.711E+00	7.264E+00	8.208E+00
		Min	**7.497E+02**	7.785E+02	1.047E+03	8.259E+02	7.808E+02	1.348E+03	1.020E+03	7.813E+02	7.892E+02	7.612E+02	7.920E+02
	50	Avg	8.322E+02	1.003E+03	1.882E+03	1.052E+03	9.829E+02	2.077E+03	1.687E+03	1.065E+03	9.411E+02	9.036E+02	9.909E+02
		SD	1.051E+01	5.068E+01	1.768E+02	1.582E+01	3.842E+01	4.253E+01	1.377E+02	4.735E+01	1.662E+01	1.491E+01	1.881E+01
		Min	**8.083E+02**	9.047E+02	1.554E+03	1.016E+03	8.946E+02	1.952E+03	1.329E+03	9.950E+02	8.893E+02	8.717E+02	9.537E+02
	100	Avg	1.031E+03	1.772E+03	3.868E+03	1.859E+03	1.670E+03	4.047E+03	3.264E+03	2.137E+03	1.530E+03	1.428E+03	1.882E+03
		SD	3.652E+01	1.614E+02	2.682E+02	2.621E+01	7.105E+01	5.176E+01	1.524E+02	1.010E+02	4.910E+01	2.697E+01	4.423E+01
		Min	**9.656E+02**	1.501E+03	3.264E+03	1.815E+03	1.558E+03	3.915E+03	2.932E+03	1.978E+03	1.419E+03	1.379E+03	1.795E+03
F_8	30	Avg	8.363E+02	8.936E+02	9.700E+02	9.000E+02	8.729E+02	1.167E+03	9.959E+02	9.045E+02	8.946E+02	8.643E+02	8.980E+02
		SD	7.906E+00	1.607E+01	2.896E+01	7.935E+00	2.034E+01	1.786E+01	5.054E+01	1.993E+01	1.072E+01	7.597E+00	1.267E+01
		Min	**8.159E+02**	8.597E+02	8.985E+02	8.842E+02	8.424E+02	1.118E+03	8.938E+02	8.627E+02	8.652E+02	8.440E+02	8.604E+02
	50	Avg	8.818E+02	1.048E+03	1.183E+03	1.081E+03	9.806E+02	1.558E+03	1.219E+03	1.069E+03	1.023E+03	9.648E+02	1.044E+03
		SD	1.017E+01	3.502E+01	5.770E+01	1.908E+01	2.521E+01	2.810E+01	7.732E+01	2.846E+01	1.474E+01	1.382E+01	1.548E+01
		Min	**8.607E+02**	9.602E+02	1.084E+03	1.030E+03	9.228E+02	1.482E+03	1.072E+03	1.001E+03	9.948E+02	9.320E+02	1.015E+03
	100	Avg	1.077E+03	1.534E+03	1.862E+03	1.764E+03	1.337E+03	2.648E+03	1.928E+03	1.589E+03	1.603E+03	1.461E+03	1.670E+03
		SD	2.390E+01	8.577E+01	8.071E+01	2.741E+01	1.135E+02	2.965E+01	1.062E+02	7.974E+01	6.448E+01	3.737E+01	3.156E+01
		Min	**1.037E+03**	1.333E+03	1.695E+03	1.689E+03	1.218E+03	2.585E+03	1.751E+03	1.425E+03	1.494E+03	1.381E+03	1.612E+03
F_9	30	Avg	9.001E+02	1.331E+03	5.608E+03	1.452E+03	1.231E+03	1.340E+04	7.802E+03	3.156E+03	2.112E+03	1.092E+03	2.871E+03
		SD	1.758E-01	2.736E+02	1.401E+03	1.401E+02	2.459E+02	9.302E+02	2.727E+03	4.542E+02	4.657E+02	8.268E+01	4.348E+02
		Min	**9.000E+02**	9.842E+02	3.371E+03	1.163E+03	9.419E+02	1.109E+04	3.580E+03	2.269E+03	1.398E+03	9.626E+02	2.117E+03
	50	Avg	9.011E+02	3.829E+03	1.361E+04	8.227E+03	4.089E+03	4.300E+04	2.304E+04	1.004E+04	1.061E+04	3.268E+03	9.792E+03
		SD	8.190E-01	9.026E+02	2.099E+03	9.653E+02	2.519E+03	3.291E+03	6.037E+03	1.827E+03	1.823E+03	5.518E+02	1.431E+03
		Min	**9.000E+02**	2.175E+03	9.923E+03	5.771E+03	1.714E+03	3.431E+04	1.432E+04	6.747E+03	6.838E+03	2.197E+03	7.116E+03
	100	Avg	9.312E+02	1.755E+04	2.583E+04	5.077E+04	2.129E+04	8.798E+04	4.749E+04	2.730E+04	5.491E+04	3.282E+04	4.871E+04
		SD	2.229E+01	3.452E+03	2.970E+03	4.223E+03	9.530E+03	3.037E+03	1.464E+04	7.015E+03	3.593E+03	3.093E+03	4.384E+03
		Min	**9.051E+02**	1.091E+04	1.895E+04	4.057E+04	9.667E+03	8.052E+04	2.796E+04	2.061E+04	4.732E+04	2.384E+04	3.845E+04
F_{10}	30	Avg	2.294E+03	4.481E+03	5.598E+03	5.001E+03	3.852E+03	9.297E+03	5.957E+03	5.230E+03	3.507E+03	3.221E+03	3.551E+03
		SD	2.633E+02	7.359E+02	6.223E+02	2.194E+02	5.301E+02	3.4215E+02	9.592E+02	1.066E+03	2.257E+02	2.125E+02	1.965E+02
		Min	**1.740E+03**	2.749E+03	4.575E+03	4.496E+03	2.849E+03	8.424E+03	4.111E+03	3.149E+03	2.893E+03	2.692E+03	3.145E+03
	50	Avg	3.744E+03	6.972E+03	8.828E+03	9.071E+03	7.183E+03	1.607E+04	1.015E+04	7.840E+03	5.936E+03	5.692E+03	6.410E+03
		SD	3.516E+02	7.636E+02	1.043E+03	3.480E+02	2.048E+03	3.757E+02	1.378E+03	9.335E+02	3.295E+02	4.212E+02	3.611E+02
		Min	**2.768E+03**	4.924E+03	6.158E+03	8.249E+03	4.977E+03	1.505E+04	7.176E+03	5.974E+03	5.201E+03	4.635E+03	5.554E+03
	100	Avg	9.416E+03	1.514E+04	1.708E+04	2.247E+04	1.426E+04	3.332E+04	2.248E+04	1.762E+04	1.443E+04	1.491E+04	1.654E+04
		SD	6.284E+02	1.256E+03	1.491E+03	6.286E+02	8.378E+02	6.755E+02	2.671E+03	3.398E+03	7.242E+02	6.757E+02	4.362E+02
		Min	**7.613E+03**	1.207E+04	1.456E+04	1.997E+04	1.296E+04	3.165E+04	1.808E+04	1.420E+04	1.252E+04	1.354E+04	1.573E+04
Ranking	**30**	**W|T|L**	6|1|0	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|1|6	0|1|6	0|0|7
	50	**W|T|L**	6|1|0	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|1|6	0|0|7
	100	**W|T|L**	7|0|0	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7	0|0|7

Table 4 Comparison of optimization results obtained from the hybrid benchmark functions

F	D	Index	CCSA	CSA	BA	CLPSO	GWO	EEGWO	WOA	KH	ABC	GABC	Best-so-far-ABC
F_{11}	30	Avg	1.114E+03	1.247E+03	4.814E+03	1.333E+03	1.330E+03	9.920E+03	1.494E+03	1.619E+03	1.746E+03	1.504E+03	1.990E+03
		SD	1.114E+01	4.828E+01	1.987E+03	2.736E+01	1.047E+02	1.903E+03	1.312E+02	2.1399E+02	3.108E+02	2.761E+02	5.763E+02
		Min	**1.104E+03**	1.166E+03	2.122E+03	1.286E+03	1.236E+03	5.168E+03	1.285E+03	1.3159E+03	1.224E+03	1.181E+03	1.294E+03
	50	Avg	1.173E+03	1.452E+03	2.215E+04	2.555E+03	2.393E+03	2.479E+04	1.699E+03	6.483E+03	4.821E+03	3.647E+03	9.057E+03
		SD	2.114E+01	7.515E+01	1.317E+04	3.042E+02	9.024E+02	1.455E+03	1.421E+02	2.0689E+03	1.880E+03	1.527E+03	2.624E+03
		Min	**1.135E+03**	1.298E+03	3.162E+03	1.918E+03	1.481E+03	2.153E+04	1.452E+03	3.360E+03	1.928E+03	1.804E+03	3.421E+03
	100	Avg	1.769E+03	8.264E+03	2.778E+05	9.913E+04	3.304E+04	4.984E+05	2.962E+04	1.023E+05	8.343E+04	7.319E+04	9.386E+04
		SD	2.398E+02	1.288E+03	1.968E+05	1.035E+04	1.069E+04	4.313E+05	1.073E+04	2.268E+04	1.507E+04	1.387E+04	1.682E+04
		Min	**1.370E+03**	5.830E+03	3.640E+04	7.926E+04	1.392E+04	1.854E+05	1.674E+04	5.463E+04	5.922E+04	4.913E+04	5.859E+04
F_{12}	30	Avg	2.694E+04	1.599E+06	9.949E+08	4.730E+06	2.516E+07	1.508E+10	3.285E+07	4.272E+08	9.470E+05	5.080E+05	2.229E+06
		SD	1.383E+04	1.262E+06	7.941E+08	1.687E+06	2.458E+07	2.915E+09	2.454E+07	2.294E+09	4.014E+05	2.734E+05	6.904E+05
		Min	**7.888E+03**	9.190E+04	6.339E+07	2.185E+06	2.278E+06	7.550E+09	2.544E+06	1.164E+06	2.354E+05	9.275E+04	9.835E+05
	50	Avg	2.069E+05	2.848E+07	1.671E+10	1.435E+08	2.331E+08	8.270E+10	2.376E+08	2.718E+09	6.983E+06	6.259E+06	4.548E+07
		SD	1.151E+05	1.582E+07	6.712E+09	1.995E+07	1.979E+08	8.814E+09	1.137E+08	1.476E+10	2.738E+06	2.316E+06	9.336E+06
		Min	**6.780E+04**	3.526E+06	4.744E+09	1.117E+08	2.073E+07	6.442E+10	3.681E+07	7.393E+06	2.183E+06	2.479E+06	2.456E+07
	100	Avg	9.028E+05	6.766E+08	9.719E+10	2.417E+09	3.552E+09	2.007E+11	1.312E+09	8.432E+08	8.027E+07	7.517E+07	1.562E+09
		SD	3.350E+05	2.510E+08	1.793E+10	3.063E+08	1.892E+09	1.218E+10	4.395E+08	4.520E+08	2.637E+07	2.600E+07	2.612E+08
		Min	**3.733E+05**	1.892E+08	5.691E+10	1.785E+09	5.329E+08	1.635E+11	5.125E+08	2.983E+08	4.238E+07	3.048E+07	1.043E+09
F_{13}	30	Avg	2.981E+03	2.340E+04	1.113E+07	2.135E+05	4.674E+06	1.358E+10	1.485E+05	2.706E+08	3.650E+04	3.066E+04	4.008E+05
		SD	1.781E+03	8.110E+03	4.570E+07	1.222E+05	2.113E+07	4.735E+09	1.051E+05	1.482E+09	1.693E+04	2.141E+04	1.698E+05
		Min	**1.331E+03**	1.047E+04	8.884E+03	7.228E+04	2.316E+04	3.820E+09	2.772E+04	1.139E+04	1.307E+04	4.022E+03	1.180E+05
	50	Avg	2.955E+03	5.134E+04	3.430E+09	8.468E+06	7.131E+07	4.768E+10	3.328E+05	1.836E+09	9.900E+04	2.468E+04	8.418E+06
		SD	1.766E+03	2.945E+04	2.774E+09	3.510E+06	9.487E+07	9.488E+09	2.744E+05	1.006E+10	6.601E+04	1.517E+04	3.330E+06
		Min	**1.375E+03**	1.794E+04	1.763E+05	2.446E+06	8.000E+04	2.640E+10	7.820E+04	1.733E+04	1.231E+04	7.108E+03	3.261E+06
	100	Avg	4.334E+03	4.205E+04	1.398E+10	2.752E+07	3.678E+08	4.921E+10	1.260E+06	3.230E+05	1.849E+05	7.767E+04	7.591E+07
		SD	2.500E+03	1.361E+04	5.938E+09	5.942E+06	4.572E+08	2.957E+09	1.011E+06	2.668E+05	1.250E+05	8.670E+04	2.150E+07
		Min	**1.551E+03**	1.866E+04	2.697E+09	1.713E+07	8.952E+05	4.186E+10	3.575E+05	6.782E+04	5.286E+04	8.143E+03	3.430E+07
F_{14}	30	Avg	1.473E+03	1.581E+03	5.579E+05	3.474E+04	1.162E+05	7.639E+06	8.756E+05	3.154E+05	1.409E+05	7.30E+04	2.396E+05
		SD	9.612E+01	6.163E+01	9.061E+05	1.893E+04	1.665E+05	4.582E+06	1.119E+06	2.585E+05	7.485E+04	5.427E+04	1.343E+05
		Min	**1.415E+03**	1.500E+03	4.418E+03	7.023E+03	4.683E+03	2.011E+06	5.098E+04	4.814E+03	3.254E+04	7.845E+03	4.939E+04
	50	Avg	4.830E+03	8.667E+03	7.051E+06	9.222E+05	5.086E+05	1.523E+08	8.733E+05	3.413E+06	1.307E+06	6.426E+05	1.767E+06
		SD	4.793E+03	7.2672E+03	7.269E+06	3.676E+05	7.507E+05	6.449E+07	5.695E+05	9.103E+06	6.204E+05	3.519E+05	8.467E+05
		Min	**1.622E+03**	1.958E+03	9.246E+04	3.973E+05	3.019E+04	3.961E+07	1.479E+05	3.732E+05	3.931E+05	1.177E+05	6.119E+05
	100	Avg	1.269E+05	2.693E+05	2.731E+07	1.801E+07	3.565E+06	1.581E+08	4.512E+06	6.817E+06	1.276E+07	9.08E+06	2.249E+07
		SD	6.377E+04	1.198E+05	2.092E+07	5.073E+06	2.635E+06	4.045E+07	2.015E+06	1.730E+06	4.182E+06	2.825E+06	6.379E+06
		Min	**2.795E+04**	1.041E+05	4.944E+06	9.314E+06	5.288E+05	7.909E+07	1.863E+06	3.473E+06	5.884E+06	3.263E+06	9.153E+06
F_{15}	30	Avg	1.572E+03	4.520E+03	6.927E+05	6.459E+03	1.900E+05	5.348E+08	6.966E+04	1.900E+04	9.555E+03	1.413E+04	9.534E+04
		SD	9.389E+01	1.511E+03	2.386E+06	2.303E+03	5.575E+05	1.909E+08	4.899E+04	8.677E+03	4.917E+03	1.019E+04	5.172E+04
		Min	**1.510E+03**	2.326E+03	9.045E+03	2.496E+03	7.366E+03	7.384E+07	1.544E+04	7.302E+03	1.953E+03	2.486E+03	1.839E+04
	50	Avg	3.537E+03	1.230E+04	2.954E+07	4.526E+05	3.890E+06	8.693E+09	8.793E+04	3.915E+08	2.765E+04	1.829E+04	1.701E+06
		SD	1.778E+03	5.402E+03	1.353E+08	2.595E+05	6.158E+06	2.288E+09	5.358E+04	2.144E+09	7.156E+03	1.367E+04	8.603E+05
		Min	**1.572E+03**	4.126E+03	8.125E+03	5.978E+04	2.057E+04	3.848E+09	3.430E+04	7.720E+03	2.028E+04	2.208E+03	4.261E+05
	100	Avg	2.420E+03	3.470E+04	3.821E+09	6.705E+06	7.859E+07	2.576E+10	1.119E+05	2.795E+04	1.703E+05	6.664E+04	2.792E+07
		SD	9.240E+02	1.264E+04	2.694E+09	2.525E+06	1.029E+08	3.272E+09	6.282E+04	5.531E+03	1.327E+05	9.058E+04	1.100E+07
		Min	**1.631E+03**	1.913E+04	1.366E+08	1.661E+06	4.607E+04	1.937E+10	6.096E+04	1.920E+04	1.466E+04	7.463E+03	1.276E+07
F_{16}	30	Avg	1.922E+03	2.380E+03	3.689E+03	2.252E+03	2.352E+03	6.912E+03	3.505E+03	3.188E+03	2.211E+03	2.078E+03	2.235E+03
		SD	1.171E+02	2.316E+02	5.459E+02	9.787E+01	2.398E+02	8.225E+02	4.066E+02	8.783E+02	1.331E+02	1.576E+02	1.254E+02
		Min	**1.638E+03**	1.939E+03	2.898E+03	2.003E+03	1.932E+03	5.319E+03	2.770E+03	2.408E+03	1.850E+03	1.780E+03	1.990E+03
	50	Avg	2.377E+03	3.065E+03	4.796E+03	3.191E+03	2.833E+03	1.080E+04	4.811E+03	3.753E+03	2.854E+03	2.859E+03	3.022E+03
		SD	1.623E+02	3.759E+02	5.697E+02	1.696E+02	3.579E+02	1.052E+03	8.071E+02	1.595E+03	2.201E+02	2.135E+02	1.955E+02
		Min	**2.032E+03**	2.431E+03	3.899E+03	2.823E+03	2.192E+03	8.766E+03	3.358E+03	2.745E+03	2.362E+03	2.242E+03	2.398E+03
	100	Avg	4.055E+03	6.221E+03	1.045E+04	8.111E+03	5.550E+03	2.532E+04	1.087E+04	7.042E+03	5.352E+03	5.626E+03	6.272E+03
		SD	3.253E+02	7.013E+02	1.184E+03	4.087E+02	9.919E+02	1.858E+03	1.941E+03	8.702E+02	3.164E+02	2.659E+02	2.498E+02
		Min	**3.300E+03**	4.521E+03	8.734E+03	7.267E+03	3.734E+03	2.227E+04	8.341E+03	5.035E+03	4.781E+03	5.087E+03	5.672E+03
F_{17}	30	Avg	1.743E+03	1.956E+03	2.889E+03	1.831E+03	1.936E+03	4.895E+03	2.506E+03	2.221E+03	1.892E+03	1.824E+03	1.905E+03
		SD	2.981E+01	1.053E+02	4.419E+02	3.028E+01	1.512E+02	1.021E+03	2.351E+02	1.949E+02	7.049E+01	5.534E+01	6.026E+01
		Min	**1.714E+03**	1.782E+03	2.032E+03	1.773E+03	1.767E+03	3.554E+03	1.946E+03	1.896E+03	1.792E+03	1.756E+03	1.801E+03
	50	Avg	2.186E+03	2.952E+03	4.669E+03	2.857E+03	2.696E+03	1.203E+04	3.910E+03	3.431E+03	2.731E+03	2.686E+03	2.793E+03
		SD	8.392E+01	2.264E+02	7.663E+02	1.319E+02	3.139E+02	3.233E+03	3.257E+02	3.148E+02	1.377E+02	1.472E+02	1.302E+02
		Min	**2.045E+03**	2.493E+03	3.671E+03	2.497E+03	2.212E+03	6.754E+03	3.352E+03	2.686E+03	2.462E+03	2.340E+03	2.517E+03
	100	Avg	3.359E+03	5.116E+03	5.527E+04	6.783E+03	4.385E+03	1.098E+07	7.562E+03	2.147E+04	4.898E+03	4.889E+03	6.126E+03
		SD	2.454E+02	4.409E+02	4.997E+04	3.911E+02	4.341E+02	5.047E+06	9.255E+02	8.492E+04	3.116E+02	3.112E+02	3.426E+02
		Min	**2.777E+03**	4.274E+03	1.144E+04	5.705E+03	3.137E+03	1.861E+06	6.176E+03	4.805E+03	4.213E+03	3.953E+03	5.384E+03
F_{18}	30	Avg	1.217E+04	1.738E+04	3.754E+06	2.558E+05	5.327E+05	1.178E+08	2.201E+06	5.259E+05	3.388E+05	1.821E+05	3.890E+05
		SD	5.831E+03	9.833E+03	5.014E+06	1.121E+05	3.574E+05	6.033E+07	2.341E+06	8.758E+05	1.317E+05	8.682E+04	1.300E+05
		Min	3.558E+03	**2.217E+03**	8.665E+04	9.957E+04	7.997E+04	3.815E+07	1.374E+05	8.565E+04	9.718E+04	7.404E+04	1.366E+05
	50	Avg	4.197E+04	1.028E+05	3.687E+07	4.033E+06	1.935E+06	2.175E+08	6.962E+06	4.624E+06	2.135E+06	1.115E+06	3.438E+06
		SD	1.525E+04	4.066E+04	3.812E+07	1.916E+06	1.304E+06	7.982E+07	6.496E+06	3.369E+06	9.077E+05	5.291E+05	1.346E+06
		Min	**1.494E+04**	4.014E+04	1.722E+05	6.679E+05	1.955E+05	6.998E+07	1.956E+06	1.278E+06	5.251E+05	2.909E+05	9.779E+05
	100	Avg	1.573E+05	3.830E+05	2.972E+07	1.891E+07	3.954E+06	2.884E+08	4.305E+06	4.116E+06	1.075E+07	6.416E+06	1.858E+07
		SD	4.327E+04	1.895E+05	2.509E+07	5.602E+06	1.886E+06	1.047E+08	1.996E+06	1.682E+06	2.746E+06	2.495E+06	4.145E+06
		Min	**9.160E+04**	1.891E+05	3.348E+06	8.117E+06	1.010E+06	1.141E+08	1.514E+06	1.900E+06	2.846E+06	2.172E+06	1.162E+07
F_{19}	30	Avg	2.002E+03	7.177E+03	1.401E+06	4.388E+03	2.421E+05	9.891E+08	2.100E+06	3.328E+05	1.780E+04	1.602E+04	5.779E+04
		SD	1.858E+02	6.929E+03	2.335E+06	1.608E+03	3.013E+05	4.047E+08	1.835E+06	3.565E+05	1.039E+04	1.314E+04	2.729E+04
		Min	**1.908E+03**	2.054E+03	2.329E+04	2.224E+03	7.217E+03	4.775E+08	2.199E+04	2.227E+04	2.838E+03	2.511E+03	2.480E+04
	50	Avg	2.332E+03	7.467E+04	1.340E+07	6.440E+04	2.300E+06	5.102E+09	3.268E+06	4.519E+05	4.848E+04	2.334E+04	1.649E+05
		SD	9.228E+02	8.324E+04	1.901E+07	3.220E+04	5.777E+06	1.344E+09	2.690E+06	3.709E+05	1.533E+04	1.422E+04	6.756E+04
		Min	**1.923E+03**	1.687E+04	7.185E+04	2.432E+04	1.203E+05	1.727E+09	2.170E+05	4.519E+04	1.960E+04	2.742E+03	6.816E+04
	100	Avg	3.205E+03	1.023E+06	3.480E+09	9.111E+06	5.703E+07	2.575E+10	2.580E+07	1.527E+09	3.664E+05	7.553E+04	3.377E+07
		SD	1.562E+03	9.691E+05	2.173E+09	3.167E+06	5.439E+07	2.943E+09	1.128E+07	5.810E+09	1.956E+05	9.861E+04	1.292E+07
		Min	**1.968E+03**	1.158E+05	1.938E+08	3.225E+06	3.113E+06	1.814E+10	6.023E+05	2.801E+05	8.428E+04	1.615E+04	1.280E+07
F_{20}	30	Avg	2.046E+03	2.329E+03	2.904E+03	2.206E+03	2.328E+03	3.247E+03	2.660E+03	2.537E+03	2.247E+03	2.197E+03	2.222E+03
		SD	5.215E+01	8.615E+01	2.154E+02	6.202E+01	1.499E+02	1.626E+02	1.976E+02	1.700E+02	8.792E+01	7.295E+01	7.172E+01
		Min	**2.004E+03**	2.258E+03	2.494E+03	2.070E+03	2.116E+03	2.771E+03	2.339E+03	2.267E+03	2.11E+03	2.077E+03	2.092E+03
	50	Avg	2.328E+03	2.802E+03	3.666E+03	2.801E+03	2.858E+03	4.539E+03	3.639E+03	3.314E+03	2.943E+03	2.788E+03	2.873E+03
		SD	1.144E+02	2.167E+02	3.761E+02	1.412E+02	3.015E+02	1.704E+02	4.725E+02	3.226E+02	1.217E+02	1.380E+02	1.412E+02
		Min	**2.112E+03**	2.328E+03	2.975E+03	2.559E+03	2.299E+03	4.050E+03	2.783E+03	2.636E+03	2.569E+03	2.524E+03	2.476E+03
	100	Avg	3.662E+03	4.666E+03	6.307E+03	5.339E+03	4.514E+03	8.324E+03	6.208E+03	5.307E+03	5.239E+03	5.081E+03	5.307E+03
		SD	2.481E+02	4.340E+02	3.953E+02	2.467E+02	7.693E+02	3.930E+02	5.920E+02	6.505E+02	2.251E+02	2.965E+02	2.761E+02
		Min	**3.108E+03**	3.888E+03	5.590E+03	4.931E+03	3.164E+03	7.173E+03	4.746E+03	3.852E+03	4.719E+03	4.407E+03	4.509E+03
Ranking	**30**	**W|T|L**	9|0|1	1|0|9	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10
	50	**W|T|L**	10|0|0	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10
	100	**W|T|L**	10|0|0	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10

Table 5 Comparison of optimization results obtained from the composition benchmark functions

F	D	Index	CCSA	CSA	BA	CLPSO	GWO	EEGWO	WOA	KH	ABC	GABC	Best-so-far ABC
F_{21}	30	Avg	2.313E+03	2.398E+03	2.520E+03	2.379E+03	2.376E+03	2.800E+03	2.556E+03	2.427E+03	2.304E+03	2.311E+03	2.343E+03
		SD	5.103E+01	2.161E+01	4.481E+01	3.930E+01	4.077E+01	2.993E+01	4.199E+01	7.347E+01	7.602E+01	6.167E+01	6.819E+01
		Min	**2.200E+03**	2.350E+03	2.440E+03	2.267E+03	2.332E+03	2.727E+03	2.490E+03	2.372E+03	2.224E+03	2.225E+03	2.229E+03
	50	Avg	2.383E+03	2.514E+03	2.776E+03	2.585E+03	2.469E+03	3.302E+03	2.889E+03	2.557E+03	2.529E+03	2.477E+03	2.556E+03
		SD	1.411E+01	4.267E+01	9.493E+01	1.695E+01	2.597E+01	6.687E+01	9.921E+01	3.325E+01	2.034E+01	2.262E+01	1.826E+01
		Min	**2.351E+03**	2.445E+03	2.594E+03	2.538E+03	2.409E+03	3.170E+03	2.670E+03	2.484E+03	2.479E+03	2.382E+03	2.518E+03
	100	Avg	2.585E+03	3.120E+03	3.988E+03	3.323E+03	2.842E+03	5.230E+03	3.923E+03	3.727E+03	3.129E+03	2.998E+03	3.204E+03
		SD	3.624E+01	9.639E+01	2.355E+02	2.469E+01	6.117E+01	1.477E+02	1.905E+02	4.624E+02	4.526E+01	3.878E+01	4.509E+01
		Min	**2.511E+03**	2.863E+03	3.510E+03	3.283E+03	2.745E+03	4.806E+03	3.565E+03	3.282E+03	3.017E+03	2.912E+03	3.073E+03
F_{22}	30	Avg	2.300E+03	2.301E+03	6.383E+03	2.526E+03	4.418E+03	9.784E+03	6.355E+03	2.708E+03	2.318E+03	2.306E+03	2.336E+03
		SD	3.794E+00	1.256E+00	1.185E+03	8.174E+01	1.456E+03	5.427E+02	2.323E+03	1.570E+03	3.564E+00	4.109E+00	7.312E+00
		Min	**2.280E+03**	2.300E+03	3.443E+03	2.419E+03	2.312E+03	8.381E+03	2.315E+03	2.300E+03	2.311E+03	2.301E+03	2.326E+03
	50	Avg	2.971E+03	3.346E+03	1.041E+04	1.018E+04	8.885E+03	1.814E+04	1.160E+04	9.809E+03	6.447E+03	6.133E+03	7.466E+03
		SD	1.369E+03	2.686E+03	1.155E+03	2.032E+03	2.179E+03	4.406E+02	1.185E+03	6.751E+02	2.415E+03	2.333E+03	1.946E+03
		Min	**2.300E+03**	2.306E+03	8.564E+03	4.358E+03	6.391E+03	1.703E+04	9.473E+03	8.646E+03	2.341E+03	2.337E+03	2.534E+03
	100	Avg	1.062E+04	1.796E+04	2.042E+04	2.534E+04	1.748E+04	3.584E+04	2.482E+04	2.080E+04	1.766E+04	1.728E+04	1.899E+04
		SD	3.403E+03	5.383E+03	1.991E+03	5.042E+02	1.233E+03	4.936E+02	2.296E+03	2.504E+03	4.786E+02	7.412E+02	5.099E+02
		Min	**2.301E+03**	2.437E+03	1.622E+04	2.420E+04	1.539E+04	3.455E+04	2.079E+04	1.693E+04	1.667E+04	1.591E+04	1.797E+04
F_{23}	30	Avg	2.674E+03	2.841E+03	3.326E+03	2.757E+03	2.738E+03	3.733E+03	3.037E+03	3.015E+03	2.715E+03	2.707E+03	2.721E+03
		SD	5.200E+01	4.955E+01	1.317E+02	1.296E+01	5.199E+01	1.617E+02	9.167E+01	2.696E+02	2.103E+01	1.186E+01	1.837E+01
		Min	**2.401E+03**	2.753E+03	3.068E+03	2.709E+03	2.685E+03	3.445E+03	2.876E+03	2.817E+03	2.692E+03	2.687E+03	2.701E+03
	50	Avg	2.804E+03	3.206E+03	3.991E+03	3.062E+03	2.901E+03	4.915E+03	3.588E+03	3.633E+03	2.956E+03	2.924E+03	2.974E+03
		SD	7.758E+01	8.588E+01	2.269E+02	1.826E+01	2.915E+01	2.435E+02	1.552E+02	5.127E+02	6.455E+01	1.480E+01	6.395E+01
		Min	**2.401E+03**	3.065E+03	3.501E+03	3.031E+03	2.845E+03	4.428E+03	3.257E+03	3.237E+03	2.788E+03	2.882E+03	2.799E+03
	100	Avg	2.973E+03	4.383E+03	5.499E+03	3.738E+03	3.391E+03	7.716E+03	4.789E+03	5.437E+03	3.287E+03	3.216E+03	3.374E+03
		SD	2.702E+01	3.367E+02	3.102E+02	3.380E+01	5.721E+01	4.449E+02	2.209E+02	1.026E+03	4.052E+01	2.415E+01	3.160E+01
		Min	**2.915E+03**	3.877E+03	4.915E+03	3.664E+03	3.258E+03	6.702E+03	4.224E+03	4.561E+03	3.179E+03	3.156E+03	3.289E+03
F_{24}	30	Avg	2.870E+03	2.951E+03	3.471E+03	2.955E+03	2.890E+03	4.044E+03	3.181E+03	3.253E+03	2.706E+03	2.826E+03	2.796E+03
		SD	1.531E+01	3.895E+01	1.575E+02	4.635E+01	3.818E+01	1.964E+02	8.929E+01	2.999E+02	1.538E+02	1.581E+02	1.695E+02
		Min	2.845E+03	2.869E+03	3.175E+03	2.756E+03	2.853E+03	3.668E+03	3.003E+03	3.016E+03	2.614E+03	**2.612E+03**	2.617E+03
	50	Avg	3.026E+03	3.290E+03	4.191E+03	3.320E+03	3.094E+03	5.381E+03	3.668E+03	3.888E+03	3.412E+03	3.301E+03	3.312E+03
		SD	3.208E+01	9.406E+01	2.186E+02	2.383E+01	1.056E+02	2.830E+02	1.357E+02	4.175E+02	5.711E+01	4.551E+01	3.922E+01
		Min	**2.958E+03**	3.141E+03	3.746E+03	3.261E+03	3.002E+03	4.716E+03	3.382E+03	3.508E+03	3.290E+03	3.200E+03	3.183E+03
	100	Avg	3.560E+03	5.394E+03	8.789E+03	4.400E+03	3.928E+03	1.279E+04	5.977E+03	7.006E+03	4.003E+03	3.892E+03	4.107E+03
		SD	3.312E+01	3.254E+02	7.068E+02	3.347E+01	1.364E+02	9.982E+02	4.496E+02	1.976E+03	5.358E+01	3.335E+01	2.741E+01
		Min	**3.464E+03**	4.634E+03	7.445E+03	4.342E+03	3.731E+03	1.045E+04	5.186E+03	5.544E+03	3.826E+03	3.816E+03	4.027E+03
F_{25}	30	Avg	2.884E+03	2.914E+03	3.453E+03	2.907E+03	2.944E+03	5.538E+03	2.945E+03	2.919E+03	2.886E+03	2.886E+03	2.891E+03
		SD	6.165E-01	2.093E+01	2.288E+02	4.289E+00	1.990E+01	3.097E+02	3.010E+01	1.547E+01	1.200E+00	1.261E+00	1.210E+00
		Min	**2.883E+03**	2.884E+03	3.064E+03	2.898E+03	2.89E+03	4.991E+03	2.899E+03	2.889E+03	2.884E+03	2.884E+03	2.889E+03
	50	Avg	2.980E+03	3.114E+03	7.997E+03	3.248E+03	3.299E+03	1.520E+04	3.190E+03	3.104E+03	3.055E+03	3.023E+03	3.096E+03
		SD	2.093E+01	2.792E+01	1.378E+03	1.753E+01	1.167E+02	1.114E+03	4.272E+01	2.110E+01	1.919E+01	1.676E+01	1.532E+01
		Min	**2.959E+03**	3.041E+03	4.385E+03	3.206E+03	3.159E+03	1.196E+04	3.132E+03	3.044E+03	2.995E+03	2.991E+03	3.058E+03
	100	Avg	3.186E+03	3.860E+03	1.832E+04	4.875E+03	4.959E+03	2.870E+04	4.084E+03	4.038E+03	3.539E+03	3.448E+03	4.230E+03
		SD	4.509E+01	1.045E+02	2.780E+03	1.012E+02	3.690E+02	1.667E+03	1.189E+02	1.613E+02	4.108E+01	2.957E+01	8.839E+01
		Min	**3.079E+03**	3.642E+03	1.174E+04	4.661E+03	4.372E+03	2.299E+04	3.888E+03	3.733E+03	3.472E+03	3.393E+03	4.088E+03
F_{26}	30	Avg	2.869E+03	3.190E+03	7.961E+03	4.326E+03	4.445E+03	1.169E+04	7.349E+03	5.923E+03	2.898E+03	3.015E+03	3.389E+03
		SD	1.942E+02	9.577E+02	8.326E+02	3.941E+02	2.886E+02	4.630E+02	1.330E+03	1.108E+03	3.723E+01	2.849E+02	2.433E+02
		Min	**2.800E+03**	**2.800E+03**	6.255E+03	3.550E+03	3.963E+03	1.043E+04	3.481E+03	**2.800E+03**	2.840E+03	2.841E+03	2.996E+03
	50	Avg	3.231E+03	4.144E+03	1.376E+04	6.880E+03	5.727E+03	1.784E+04	1.273E+04	9.828E+03	3.896E+03	4.358E+03	5.074E+03
		SD	6.747E+02	2.197E+03	1.013E+03	1.778E+02	3.966E+02	4.991E+02	1.316E+03	1.662E+03	9.960E+02	8.445E+02	9.164E+02
		Min	**2.900E+03**	2.921E+03	1.128E+04	6.521E+03	4.928E+03	1.667E+04	9.864E+03	4.078E+03	3.040E+03	3.443E+03	3.979E+03
	100	Avg	8.424E+03	2.032E+04	4.476E+04	1.676E+04	1.247E+04	5.648E+04	3.224E+04	2.715E+04	1.254E+04	1.215E+04	1.478E+04
		SD	3.758E+02	6.305E+03	6.951E+03	3.457E+02	9.677E+02	1.939E+03	4.359E+03	5.715E+03	2.905E+03	1.253E+03	3.612E+02
		Min	7.676E+03	6.524E+03	3.609E+04	1.608E+04	1.085E+04	5.120E+04	2.238E+04	2.036E+04	**4.908E+03**	5.818E+03	1.401E+04
F_{27}	30	Avg	3.199E+03	3.360E+03	3.200E+03	3.233E+03	3.229E+03	4.905E+03	3.366E+03	3.456E+03	3.212E+03	3.207E+03	3.213E+03
		SD	6.103E+00	5.440E+01	8.012E-05	4.773E+00	1.194E+01	4.323E+02	8.122E+01	2.132E+02	4.458E+00	3.707E+00	4.228E+00
		Min	**3.185E+03**	3.258E+03	3.200E+03	3.222E+03	3.210E+03	4.040E+03	3.228E+03	3.326E+03	3.201E+03	3.199E+03	3.206E+03
	50	Avg	3.254E+03	4.132E+03	3.200E+03	3.643E+03	3.466E+03	8.082E+03	4.364E+03	4.880E+03	3.375E+03	3.327E+03	3.396E+03
		SD	2.293E+01	3.163E+02	8.660E-05	3.950E+01	6.295E+01	4.811E+02	3.935E+02	9.250E+02	2.678E+01	1.904E+01	3.081E+01
		Min	3.212E+03	3.715E+03	**3.200E+03**	3.547E+03	3.366E+03	6.862E+03	3.621E+03	3.943E+03	3.317E+03	3.283E+03	3.327E+03
	100	Avg	3.354E+03	4.885E+03	3.200E+03	4.379E+03	3.819E+03	1.520E+04	4.937E+03	8.055E+03	3.529E+03	3.482E+03	3.550E+03
		SD	1.721E+01	3.540E+02	7.336E-05	9.789E+01	1.003E+02	1.250E+03	6.336E+02	2.809E+03	3.258E+01	2.202E+01	2.115E+01
		Min	3.315E+03	4.242E+03	**3.200E+03**	4.126E+03	3.607E+03	1.252E+04	4.142E+03	5.069E+03	3.460E+03	3.434E+03	3.494E+03
F_{28}	30	Avg	3.133E+03	3.233E+03	3.307E+03	3.317E+03	3.340E+03	7.492E+03	3.309E+03	3.253E+03	3.216E+03	3.212E+03	3.277E+03
		SD	4.681E+01	2.273E+01	9.119E-05	7.890E+00	6.090E+01	4.713E+02	3.867E+01	2.542E+01	6.175E+00	5.546E+00	1.307E+01
		Min	**3.100E+03**	3.202E+03	3.300E+03	3.300E+03	3.221E+03	6.537E+03	3.231E+03	3.199E+03	3.203E+03	3.205E+03	3.241E+03
	50	Avg	3.259E+03	3.413E+03	3.300E+03	4.080E+03	3.741E+03	1.354E+04	3.570E+03	3.509E+03	3.336E+03	3.293E+03	3.501E+03
		SD	3.632E-02	4.972E+01	6.639E-05	1.113E+02	2.119E+02	7.835E+02	9.093E+01	6.137E+02	1.716E+01	1.379E+01	6.798E+01
		Min	**3.259E+03**	3.293E+03	3.300E+03	3.867E+03	3.452E+03	1.164E+04	3.429E+03	3.335E+03	3.299E+03	3.269E+03	3.409E+03
	100	Avg	3.355E+03	4.079E+03	3.300E+03	9.445E+03	6.264E+03	3.608E+04	4.516E+03	6.059E+03	3.750E+03	3.580E+03	8.729E+03
		SD	1.368E+01	1.520E+02	5.419E-05	4.483E+02	8.476E+02	1.144E+03	2.318E+02	6.979E+02	7.367E+01	2.403E+01	1.329E+03
		Min	3.328E+03	3.888E+03	**3.300E+03**	8.453E+03	5.098E+03	3.362E+04	4.089E+03	4.724E+03	3.634E+03	3.523E+03	5.892E+03
F_{29}	30	Avg	3.318E+03	3.855E+03	5.433E+03	3.571E+03	3.593E+03	8.893E+03	4.789E+03	4.227E+03	3.529E+03	3.438E+03	3.574E+03
		SD	3.954E+01	1.723E+02	5.608E+02	5.949E+01	9.701E+01	1.634E+03	3.299E+02	2.440E+02	5.690E+01	5.260E+01	5.669E+01
		Min	**3.243E+03**	3.632E+03	4.604E+03	3.417E+03	3.437E+03	6.989E+03	4.131E+03	3.745E+03	3.398E+03	3.334E+03	3.468E+03
	50	Avg	3.364E+03	5.025E+03	1.145E+04	4.390E+03	4.186E+03	8.244E+04	7.318E+03	5.576E+03	4.040E+03	3.721E+03	3.991E+03
		SD	9.855E+01	4.051E+02	2.600E+03	1.641E+02	2.598E+02	5.540E+04	9.797E+02	5.709E+02	1.479E+02	1.279E+02	1.190E+02
		Min	**3.218E+03**	4.095E+03	6.253E+03	4.021E+03	3.813E+03	1.801E+04	5.596E+03	4.961E+03	3.763E+03	3.472E+03	3.697E+03
	100	Avg	5.285E+03	9.439E+03	2.697E+04	1.054E+04	7.316E+03	6.969E+05	1.431E+04	1.026E+04	7.427E+03	6.535E+03	8.442E+03
		SD	3.146E+02	6.810E+02	1.080E+04	4.319E+02	4.519E+02	2.777E+05	1.789E+03	1.434E+03	3.637E+02	3.193E+02	4.498E+02
		Min	**4.590E+03**	8.393E+03	1.265E+04	9.461E+03	6.714E+03	1.906E+05	1.102E+04	8.373E+03	6.776E+03	6.039E+03	7.701E+03
F_{30}	30	Avg	7.931E+03	1.488E+05	3.719E+07	9.198E+04	3.965E+06	1.935E+09	9.906E+06	1.975E+08	2.264E+04	1.228E+04	5.418E+04
		SD	1.313E+03	1.080E+05	5.262E+07	3.625E+04	2.571E+06	5.672E+08	6.675E+06	5.001E+08	6.028E+03	3.961E+03	1.704E+04
		Min	**5.985E+03**	2.504E+04	4.080E+04	3.205E+04	3.430E+05	1.029E+09	1.464E+06	3.711E+05	1.231E+04	6.546E+03	2.528E+04
	50	Avg	7.621E+05	3.413E+07	3.740E+08	1.847E+07	8.182E+07	7.971E+09	9.342E+07	7.253E+07	1.019E+06	1.012E+06	2.840E+06
		SD	8.736E+04	9.782E+06	2.692E+08	4.622E+06	2.546E+07	1.731E+09	2.752E+07	2.770E+07	1.172E+05	2.019E+05	6.412E+05
		Min	**5.922E+05**	1.207E+07	1.731E+07	1.102E+07	3.747E+07	4.628E+09	3.183E+07	2.884E+07	8.375E+05	6.900E+05	1.699E+06
	100	Avg	1.197E+04	6.820E+07	1.106E+10	8.306E+07	3.657E+08	4.304E+10	4.485E+08	1.089E+08	3.453E+05	8.764E+04	8.160E+07
		SD	2.287E+03	4.220E+07	3.575E+09	1.516E+07	2.942E+08	4.262E+09	2.620E+08	6.428E+07	2.194E+05	4.842E+04	2.113E+07
		Min	**8.657E+03**	1.854E+07	4.521E+09	4.492E+07	1.621E+07	3.481E+10	1.697E+08	2.625E+07	7.121E+04	2.208E+04	4.485E+07
	30	**W|T|L**	9|0|1	0|2|8	1|0|9	0|0|10	0|0|10	0|0|10	0|0|10	0|1|9	0|1|9	1|1|8	0|0|10
Ranking	**50**	**W|T|L**	10|0|0	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10
	100	**W|T|L**	7|0|3	0|0|10	2|0|8	0|0|10	0|0|10	0|0|10	0|0|10	0|0|10	1|0|9	0|0|10	0|0|10

5.4. Convergence analysis

In this experiment, the convergence behavior of CCSA was compared with other state-of-the-art swarm intelligence algorithms. Figures 12-15 show the convergence curves for solving different categories unimodal, simple multi-modal, hybrid and composition, respectively. In all curves, the convergence of CCSA is better than compared algorithms.

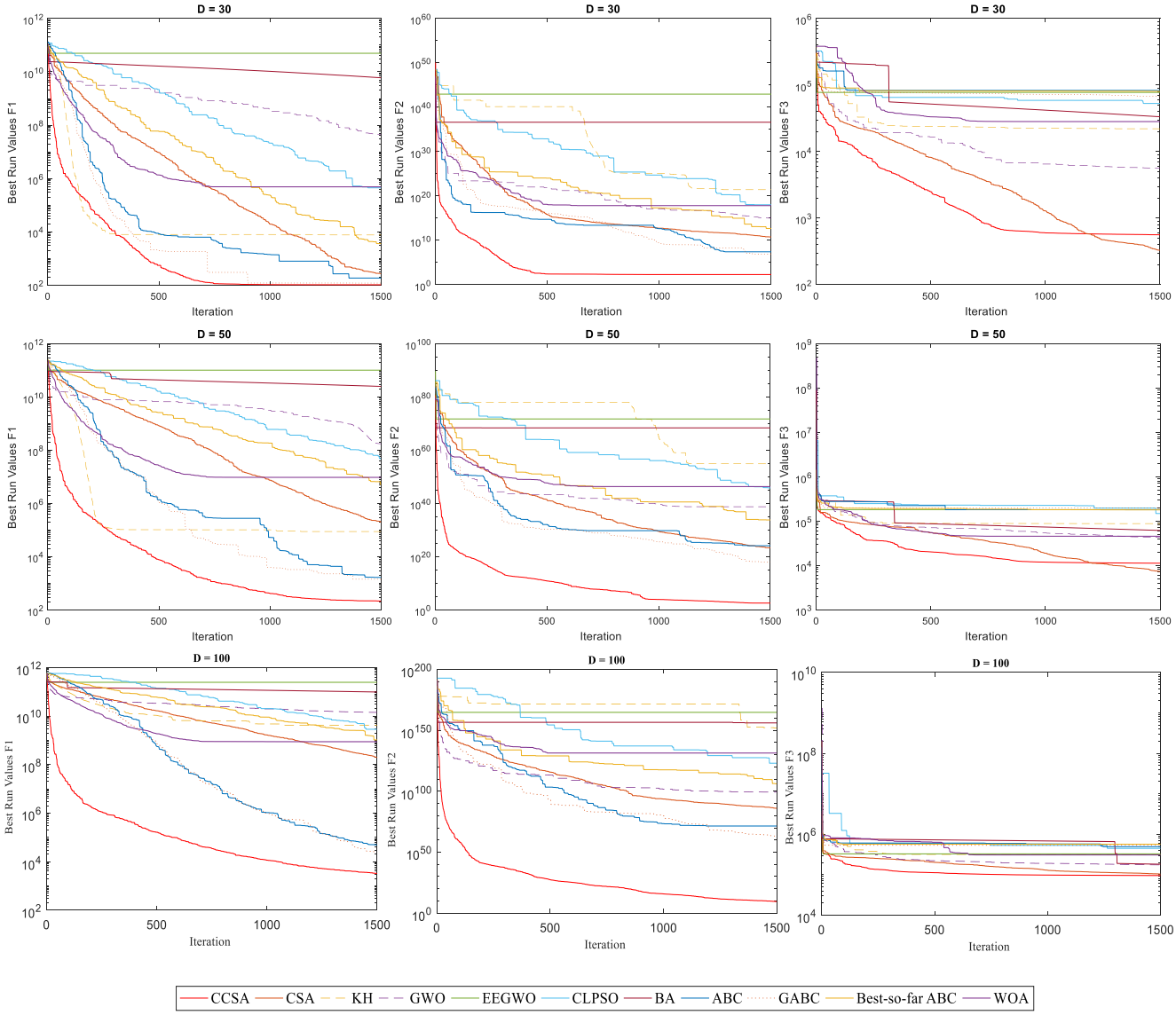


Fig. 12 Convergence analysis of unimodal functions with different dimensions

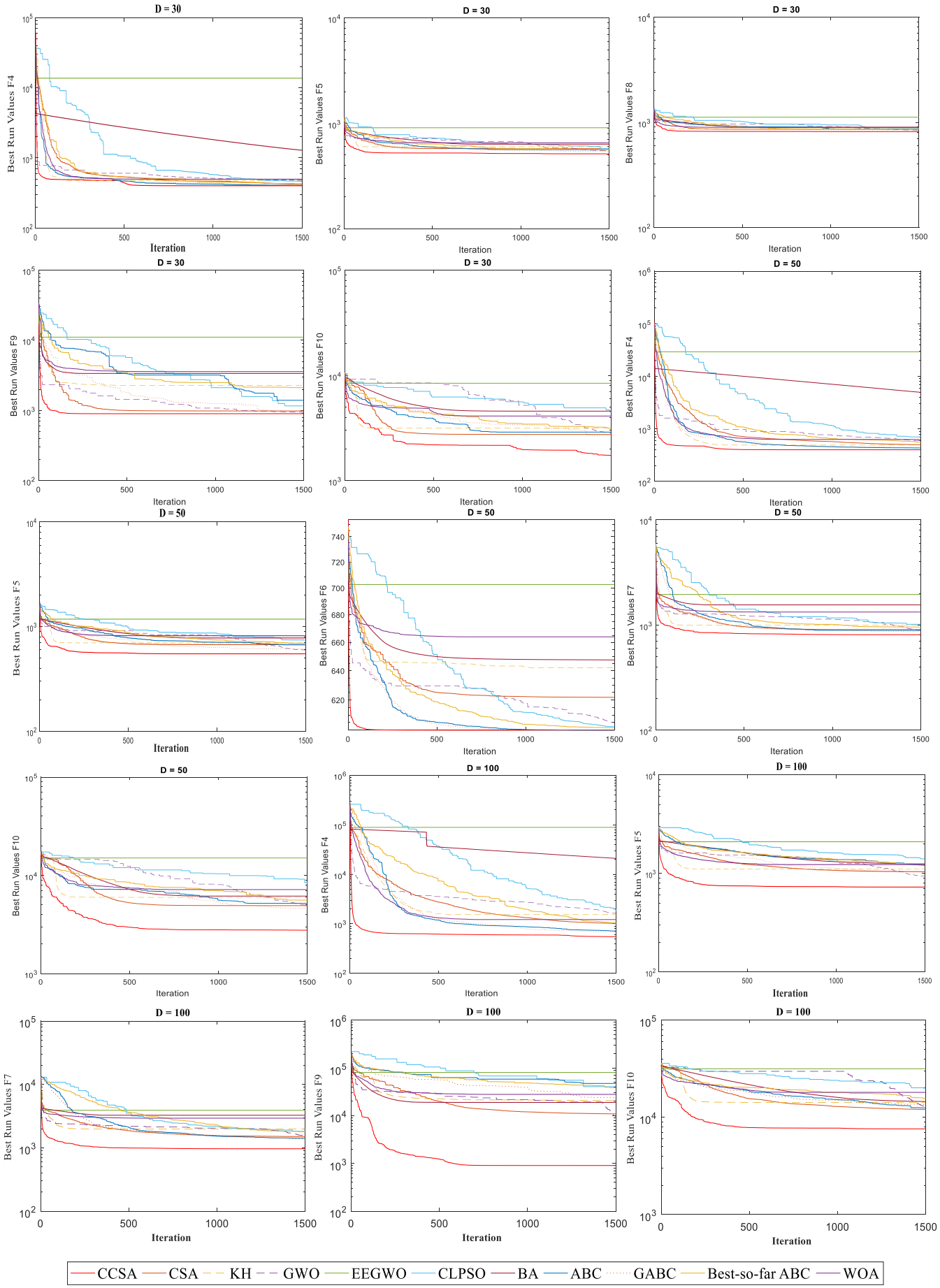


Fig. 13 Convergence analysis of simple multi-modal test functions with different dimensions

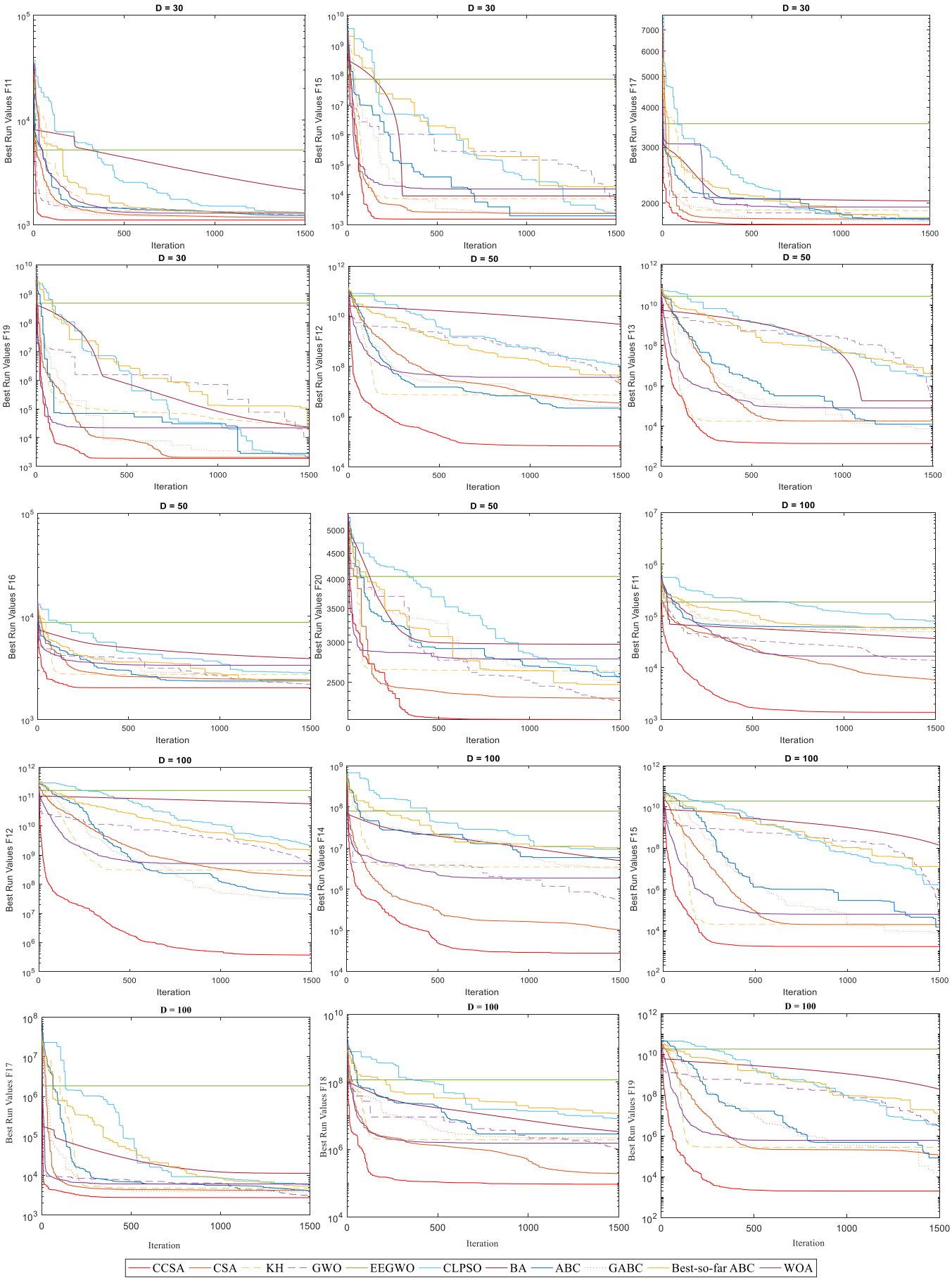


Fig. 14 Convergence analysis on hybrid test functions with different dimensions

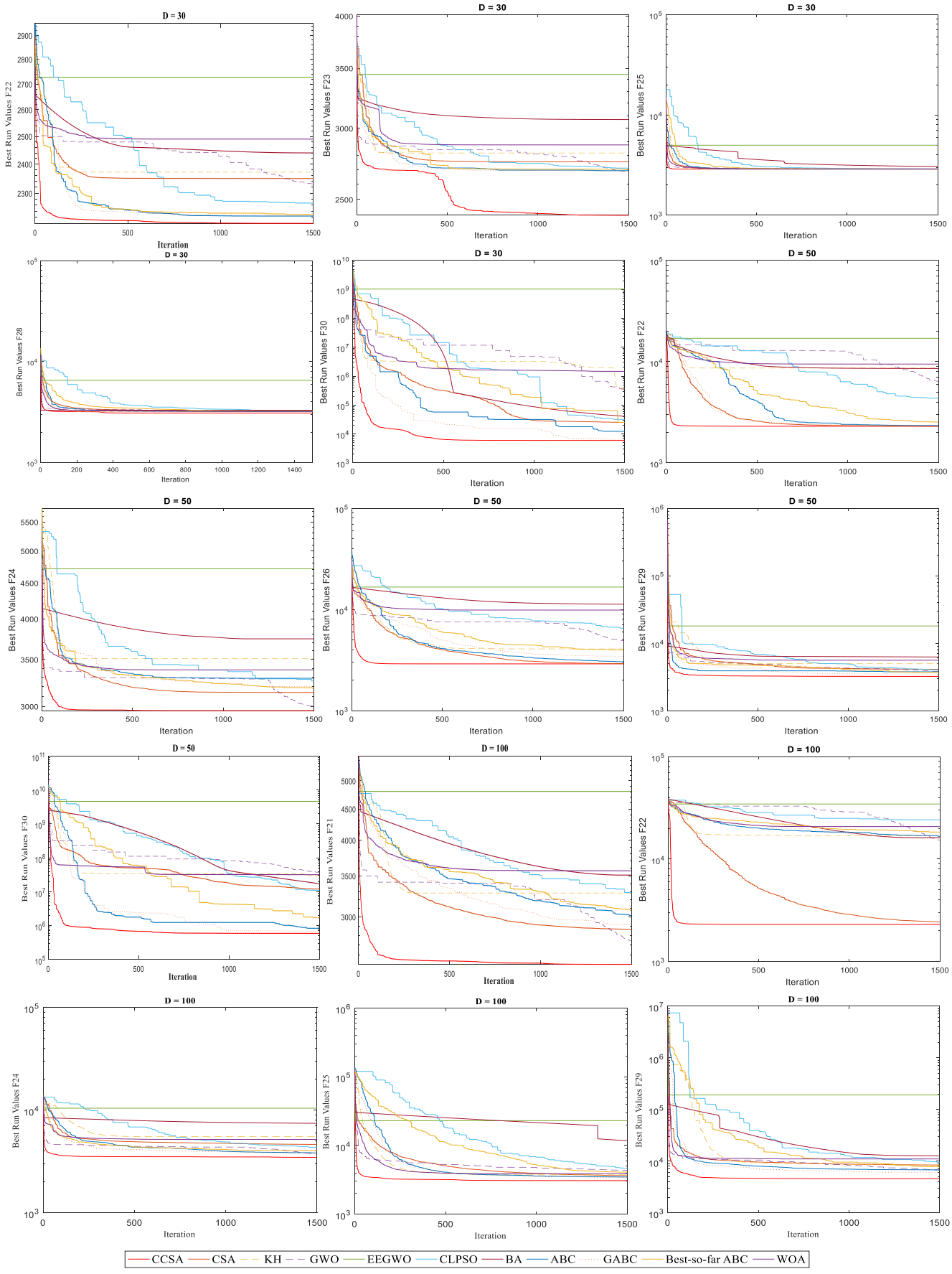


Fig. 15 Convergence analysis on composition test functions with different dimensions

5.5. Efficiency evaluation for large-scale global optimization

Although some bio-inspired optimization algorithms have functional search capabilities for solving low-dimensional problems (D <100), their efficiency is decreased when the dimension is increased in large-scale global optimization problems. In this experiment set, the efficiency of CCSA was compared with the compared algorithms by several experiments on benchmark functions CEC 2010 with dimension 1000. This experiment was run 15 times with MaxIt = 300 and population size N=30. The experimental results are shown in Table 6, and their convergence curves are shown in Fig. 16. The results show that the proposed CCSA is more efficient than compared algorithms for solving large-scale problems.

Table 6 Comparison of optimization results obtained from CEC 2010 benchmark functions with D=1000

F	Index	CCSA	CSA	BA	CLPSO	GWO	EEGWO	WOA	KH	ABC	GABC	Best-so-far ABC
F_1	Avg	1.440E+09	1.134E+11	2.314E+11	2.441E+11	6.510E+10	1.908E+11	1.252E+11	1.771E+11	3.148E+11	3.025E+11	3.078E+11
	SD	2.439E+08	6.931E+09	2.145E+10	2.125E+10	5.477E+09	3.285E+09	6.213E+09	8.614E+09	1.844E+10	1.079E+10	1.271E+10
	Min	**1.170E+09**	1.009E+11	1.953E+11	2.116E+11	5.380E+10	1.827E+11	1.121E+11	1.631E+11	2.800E+11	2.784E+11	2.835E+11
F_2	Avg	1.719E+11	1.933E+11	1.910E+11	1.935E+11	1.829E+11	1.966E+11	1.822E+11	1.958E+11	1.951E+11	1.930E+11	1.926E+11
	SD	3.972E+08	8.086E+08	2.854E+09	7.048E+08	6.775E+08	6.716E+08	1.147E+09	9.188E+08	6.799E+08	5.435E+08	4.830E+08
	Min	**1.715E+11**	1.915E+11	1.861E+11	1.921E+11	1.817E+11	1.954E+11	1.803E+11	1.945E+11	1.932E+11	1.921E+11	1.916E+11
F_3	Avg	6.603E+10	1.617E+11	1.841E+11	1.754E+11	1.122E+11	1.911E+11	1.273E+11	1.911E+11	1.872E+11	1.784E+11	1.767E+11
	SD	5.768E+08	2.803E+09	2.009E+10	7.186E+09	3.093E+09	3.213E+09	3.660E+09	3.889E+09	6.522E+09	4.239E+09	3.930E+09
	Min	**6.548E+10**	1.577E+11	1.149E+11	1.593E+11	1.067E+11	1.857E+11	1.193E+11	1.808E+11	1.768E+11	1.711E+11	1.697E+11
F_4	Avg	1.454E+09	1.165E+11	2.453E+11	2.478E+11	6.160E+10	1.925E+11	1.296E+11	1.782E+11	3.206E+11	3.025E+11	3.053E+11
	SD	1.960E+08	3.893E+09	2.012E+10	1.494E+10	5.737E+09	3.357E+09	6.509E+09	5.631E+09	1.887E+10	1.485E+10	1.511E+10
	Min	**1.138E+09**	1.080E+11	2.029E+11	2.102E+11	5.199E+10	1.835E+11	1.162E+11	1.661E+11	2.853E+11	2.735E+11	2.751E+11
F_5	Avg	4.237E+08	1.935E+11	1.897E+11	1.932E+11	1.830E+11	1.968E+11	1.821E+11	1.960E+11	1.947E+11	1.931E+11	1.925E+11
	SD	5.066E+07	5.587E+08	3.634E+09	7.306E+08	4.957E+08	7.565E+08	1.528E+09	6.349E+08	9.296E+08	7.446E+08	7.399E+08
	Min	**3.389E+08**	1.925E+11	1.789E+11	1.914E+11	1.821E+11	1.957E+11	1.798E+11	1.947E+11	1.930E+11	1.917E+11	1.906E+11
F_6	Avg	5.052E+08	1.622E+11	1.908E+11	1.775E+11	1.119E+11	1.916E+11	1.237E+11	1.920E+11	1.894E+11	1.772E+11	1.741E+11
	SD	2.904E+07	2.245E+09	6.995E+09	5.392E+09	3.365E+09	2.280E+09	5.151E+09	4.469E+09	3.314E+09	5.040E+09	4.682E+09
	Min	**4.548E+08**	1.584E+11	1.769E+11	1.694E+11	1.056E+11	1.883E+11	1.155E+11	1.853E+11	1.843E+11	1.667E+11	1.609E+11
F_7	Avg	6.181E+08	1.124E+11	2.360E+11	2.510E+11	6.222E+10	1.912E+11	1.292E+11	1.754E+11	3.021E+11	2.954E+11	3.140E+11
	SD	6.298E+07	4.775E+09	1.568E+10	1.950E+10	5.621E+09	3.153E+09	5.991E+09	9.804E+09	2.426E+10	1.498E+10	1.303E+10
	Min	**4.887E+08**	1.044E+11	2.093E+11	2.138E+11	4.953E+10	1.833E+11	1.224E+11	1.623E+11	2.402E+11	2.660E+11	2.868E+11
F_8	Avg	5.767E+08	1.131E+11	2.407E+11	2.462E+11	5.873E+10	1.920E+11	1.281E+11	1.771E+11	3.198E+11	3.009E+11	3.066E+11
	SD	4.465E+07	7.824E+09	1.951E+10	1.662E+10	5.064E+09	3.008E+09	4.703E+09	8.584E+09	1.739E+10	1.792E+10	1.492E+10
	Min	**5.167E+08**	1.047E+11	2.134E+11	2.218E+11	5.014E+10	1.859E+11	1.197E+11	1.620E+11	2.889E+11	2.592E+11	2.789E+11
F_9	Avg	5.912E+08	1.128E+11	2.464E+11	2.432E+11	6.018E+10	1.938E+11	1.279E+11	1.752E+11	3.064E+11	3.006E+11	3.167E+11
	SD	5.433E+07	3.877E+09	2.417E+10	1.683E+10	4.090E+09	1.874E+09	6.135E+09	1.023E+10	1.671E+10	1.407E+10	1.164E+10
	Min	**5.211E+08**	1.072E+11	2.014E+11	2.109E+11	5.392E+10	1.899E+11	1.139E+11	1.536E+11	2.717E+11	2.718E+11	2.930E+11
F_{10}	Avg	4.176E+08	1.934E+11	1.904E+11	1.931E+11	1.827E+11	1.969E+11	1.822E+11	1.956E+11	1.947E+11	1.932E+11	1.926E+11
	SD	4.137E+07	8.662E+08	3.117E+09	7.037E+08	5.223E+08	8.726E+08	7.863E+08	9.685E+08	1.210E+09	6.270E+08	8.025E+08
	Min	**3.631E+08**	1.917E+11	1.854E+11	1.917E+11	1.820E+11	1.950E+11	1.805E+11	1.938E+11	1.927E+11	1.923E+11	1.911E+11
F_{11}	Avg	2.093E+02	1.615E+11	1.908E+11	1.776E+11	1.128E+11	1.931E+11	1.235E+11	1.899E+11	1.873E+11	1.790E+11	1.765E+11
	SD	2.876E+00	3.189E+09	6.545E+09	6.467E+09	3.948E+09	2.744E+09	5.967E+09	4.215E+09	8.921E+09	4.663E+09	5.263E+09
	Min	**2.049E+02**	1.538E+11	1.778E+11	1.649E+11	1.048E+11	1.887E+11	1.158E+11	1.807E+11	1.634E+11	1.658E+11	1.675E+11
F_{12}	Avg	2.104E+02	1.156E+11	2.357E+11	2.454E+11	6.026E+10	1.928E+11	1.294E+11	1.778E+11	3.176E+11	2.994E+11	3.076E+11
	SD	5.192E+00	5.406E+09	2.202E+10	1.541E+10	5.639E+09	3.095E+09	6.332E+09	7.112E+09	1.998E+10	1.487E+10	1.253E+10
	Min	**2.043E+02**	1.061E+11	2.018E+11	2.161E+11	5.135E+10	1.858E+11	1.158E+11	1.643E+11	2.515E+11	2.672E+11	2.857E+11
F_{13}	Avg	2.085E+02	1.146E+11	2.340E+11	2.455E+11	6.187E+10	1.914E+11	1.222E+11	1.775E+11	3.233E+11	3.075E+11	3.112E+11
	SD	5.194E+00	3.707E+09	1.285E+10	1.547E+10	5.771E+09	3.382E+09	6.733E+09	7.389E+09	2.156E+10	1.956E+10	1.416E+10
	Min	**2.020E+02**	1.094E+11	2.135E+11	2.148E+11	5.467E+10	1.859E+11	1.130E+11	1.619E+11	2.822E+11	2.503E+11	2.933E+11
F_{14}	Avg	2.122E+02	1.134E+11	2.389E+11	2.516E+11	6.077E+10	1.922E+11	1.248E+11	1.771E+11	3.244E+11	3.000E+11	3.126E+11
	SD	2.958E+00	5.803E+09	1.983E+10	1.150E+10	5.901E+09	3.796E+09	4.432E+09	6.089E+09	1.799E+10	1.178E+10	1.351E+10
	Min	**2.080E+02**	1.065E+11	2.097E+11	2.315E+11	4.721E+10	1.834E+11	1.176E+11	1.686E+11	3.004E+11	2.821E+11	2.762E+11
F_{15}	Avg	2.202E+02	1.935E+11	1.910E+11	1.935E+11	1.829E+11	1.963E+11	1.815E+11	1.959E+11	1.947E+11	1.931E+11	1.926E+11
	SD	1.053E+00	7.479E+08	3.946E+09	7.550E+08	8.862E+08	1.015E+09	1.178E+09	7.583E+08	1.098E+09	6.566E+08	6.476E+08
	Min	**2.183E+02**	1.924E+11	1.790E+11	1.919E+11	1.811E+11	1.945E+11	1.795E+11	1.943E+11	1.929E+11	1.921E+11	1.918E+11
F_{16}	Avg	2.101E+02	1.627E+11	1.851E+11	1.765E+11	1.115E+11	1.908E+11	1.263E+11	1.911E+11	1.882E+11	1.781E+11	1.744E+11
	SD	3.071E+00	3.074E+09	1.715E+10	5.363E+09	3.428E+09	4.894E+09	5.168E+09	4.346E+09	7.595E+09	4.364E+09	4.927E+09
	Min	**2.033E+02**	1.563E+11	1.380E+11	1.682E+11	1.036E+11	1.775E+11	1.189E+11	1.852E+11	1.760E+11	1.712E+11	1.621E+11
F_{17}	Avg	2.088E+02	1.136E+11	2.363E+11	2.416E+11	6.184E+10	1.930E+11	1.257E+11	1.798E+11	3.199E+11	3.017E+11	3.080E+11
	SD	3.649E+00	3.720E+09	1.120E+10	1.441E+10	5.345E+09	1.900E+09	6.958E+09	7.104E+09	2.350E+10	1.048E+10	1.410E+10
	Min	**2.007E+02**	1.072E+11	2.155E+11	2.189E+11	5.227E+10	1.891E+11	1.140E+11	1.654E+11	2.794E+11	2.861E+11	2.791E+11
F_{18}	Avg	2.100E+02	1.125E+11	2.425E+11	2.509E+11	6.071E+10	1.929E+11	1.256E+11	1.749E+11	3.224E+11	3.050E+11	3.123E+11
	SD	3.683E+00	5.672E+09	1.442E+10	1.227E+10	5.333E+09	2.874E+09	4.290E+09	7.353E+09	1.722E+10	1.315E+10	1.729E+10
	Min	**2.044E+02**	1.042E+11	2.192E+11	2.285E+11	5.008E+10	1.860E+11	1.188E+11	1.623E+11	2.948E+11	2.763E+11	2.807E+11
F_{19}	Avg	2.110E+02	1.128E+11	2.371E+11	2.476E+11	6.182E+10	1.917E+11	1.284E+11	1.808E+11	3.198E+11	2.985E+11	3.114E+11
	SD	4.047E+00	6.283E+09	2.390E+10	1.312E+10	6.175E+09	3.265E+09	7.918E+09	6.796E+09	1.310E+10	1.088E+10	1.170E+10
	Min	**2.042E+02**	1.022E+11	2.050E+11	2.277E+11	5.089E+10	1.839E+11	1.151E+11	1.696E+11	2.910E+11	2.778E+11	2.911E+11
F_{20}	Avg	2.094E+02	1.120E+11	2.316E+11	2.473E+11	6.005E+10	1.935E+11	1.276E+11	1.743E+11	3.107E+11	2.963E+11	3.101E+11
	SD	4.656E+00	4.190E+09	1.452E+10	1.989E+10	6.312E+09	2.875E+09	6.609E+09	8.446E+09	2.179E+10	1.470E+10	1.083E+10
	Min	**2.003E+02**	1.060E+11	2.067E+11	2.036E+11	5.155E+10	1.866E+11	1.130E+11	1.572E+11	2.740E+11	2.607E+11	2.837E+11
Ranking	**W|T|L**	20|0|0	0|0|20	0|0|20	0|0|20	0|0|20	0|0|20	0|0|20	0|0|20	0|0|20	0|0|20	0|0|20

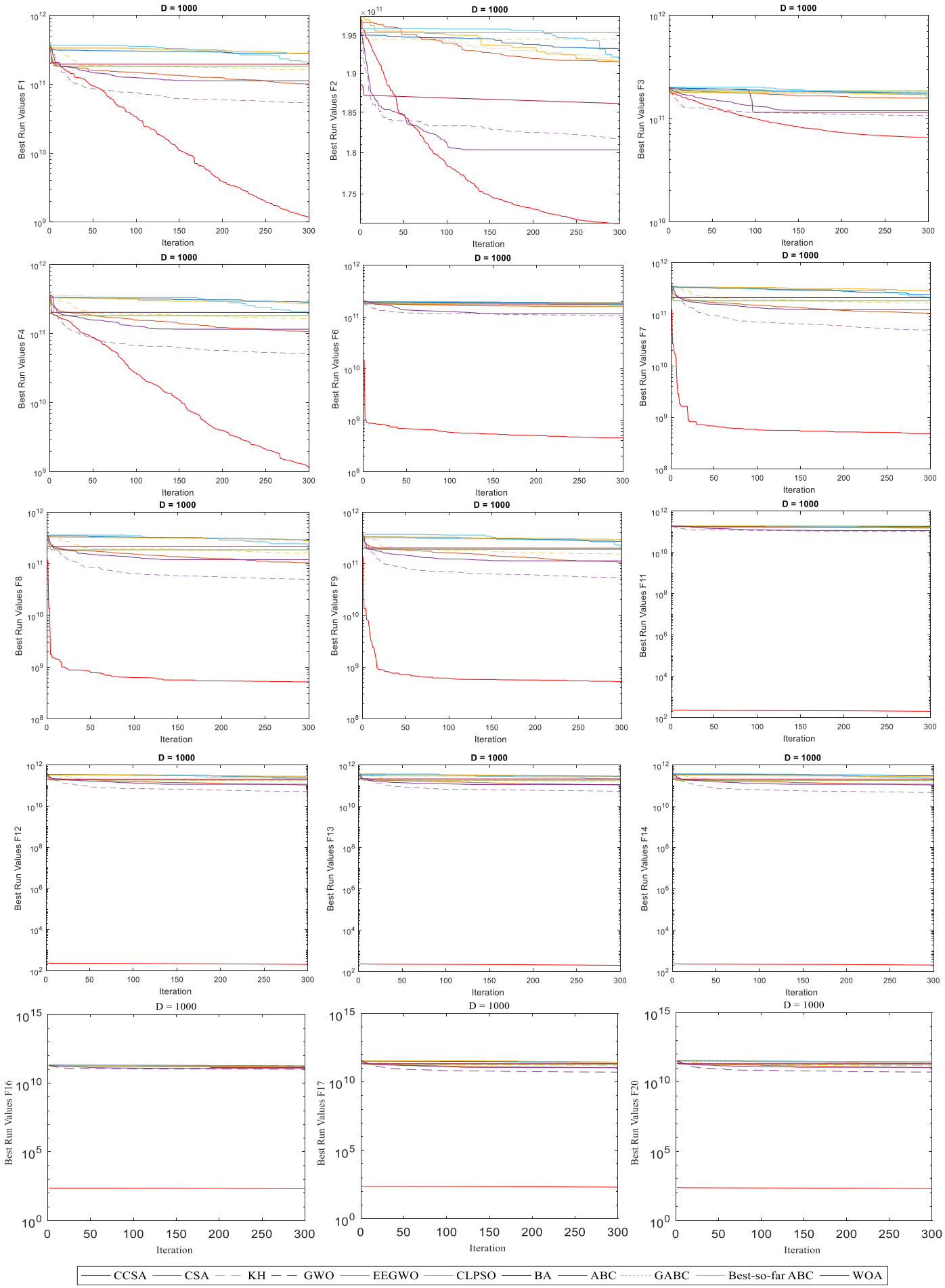


Fig. 16 Convergence analysis on benchmark functions CEC 2010

5.6. Statistical analysis

Besides the above experimental evaluation, CCSA was statistically evaluated to prove its overall performance. Therefore, in this section, the non-parametric Friedman test and Mean Absolute Error (MAE) tests were used by different dimensions 30, 50, 100, and 1000.

5.6.1. Non-parametric Friedman test

The Friedman statistic test (F_f) is used to detect significant differences between the results of two or more algorithms on continuous data. F_f can be used for multiple comparisons among several algorithms by computing the ranking of the observed results for each bio-inspired metaheuristic algorithm. Such that, in each function, the best results and the second best results are considered by rank 1 and 2, and the ranking k is given to the worst results. F_f is computed by Eq. (8) [63].

$$F_f = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \tag{8}$$

Where k is the number of bio-inspired metaheuristic algorithms participated in the test, j represents its associated index, N is the number of test cases or runs, and R_j is standing for the average rank for each algorithm. In addition, the distribution of p-value is according to a chi-squared distribution with k-1degrees of freedom. It is common to declare a result as a significant one if the p-value is less than 0.05 or 0.01.

Based on the results of the Friedman statistical test shown in Tables 7 and 8, there exists a significant difference between CCSA and the compared algorithms for benchmark functions CEC 2017 and CEC 2010. Moreover, Table 9 shows the p-value of some benchmark functions in CEC 2017 for different dimensions 30, 50 and 100, and in CEC 2010 for dimension1000. In this regard, for a majority of functions in both CEC 2017 and CEC 2010, the p-values are below 0.05, which verify the efficiency of the proposed CCSA.

Table 7 Overall rank by Friedman test in dimensions D=30, 50 and 100

Algorithm	D	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
CCSA	30	2.97	**1**	2	**1.30**	**1**	3	**1.03**	**1**	**1**	**1**	**1**	**1**	**1**	**1.10**	**1**
	50	**1**	**1**	**1.97**	**1.1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1.30**	**1.07**
	100	**1**	**1**	**1.47**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1.20**	**1**
CSA	30	3.17	3.87	**1**	5.60	6.63	7	5.43	5.23	3.60	6.33	2.50	3.83	2.80	1.90	2.57
	50	4.067	3.53	**1.03**	5.73	6.03	7.03	5.33	5.87	3.37	5.33	2.07	4.7	3.37	1.70	2.63
	100	**4**	3.87	1.53	4.27	4	7.10	5.23	4.37	2.47	4.60	2	4.57	2.47	1.80	3.13
BA	30	9.97	10.03	8.40	10	9.07	9.17	9.77	9.2	9.23	8.60	10.07	9.97	7.27	6.70	8
	50	10	10.07	9.27	10	9.13	8.97	9.93	9.2	8.73	8.23	9.97	9.97	9.87	8.60	6.53
	100	10	9.77	8.53	10	7.97	8.87	10.33	9.13	4.40	6.90	9.90	10	10	8.53	10
CLPSO	30	7	6.33	5.33	7.67	5.53	5	7.17	6.43	4.63	7.60	4	6.43	8.03	4.47	3.37
	50	7.67	6.87	6.8	8.03	7.57	5	7.17	7.43	5.4	8.63	4.93	7.57	7.87	5.9	8.30
	100	7.033	7.03	7.20	7.60	9.03	5.07	6.07	8.13	8.63	9.47	7.87	8.30	7.20	8.40	7.40
GWO	30	8.97	7.57	3.13	7.47	4.13	6	5.43	3.43	3.37	4.80	3.73	7.90	6.97	5.17	7.80
	50	9	6.17	3.43	7.93	3.03	6	4.83	2.83	3.17	5.33	4.43	7.93	8.27	3.97	8.30
	100	9	5.60	3.13	8.30	2.07	5.93	4.27	2.27	3.47	3.27	3.63	8.20	8.53	3.63	7.60
EEGWO	30	11	10.9	6.60	11	10.97	11	10.9	11	11	11	10. 9	11	11	11	11
	50	11	10.83	9.33	11	11	11	10.9	11	11	11	10.67	11	10.97	11	10.97
	100	11	10.13	4.80	11	11	11	10.67	11	10.97	11	10.73	11	11	11	11
WOA	30	7.93	7.37	9.03	8.10	9.77	9.83	9.33	9.53	9.77	9	6.10	8.40	7.40	8.83	8.40
	50	7.30	8	3.83	7.77	9.77	9.97	9.17	9.73	9.93	9.3	3.20	8.3	6.13	5.83	6.83
	100	5.53	7.97	9.63	6.07	9.30	10	9	9.67	7.70	9.27	3.43	6.20	5.87	4.17	5.33
KH	30	6.1	8.80	4.13	5.63	7.63	8	5.9	6.43	7.57	7.73	6.87	7.03	4.50	7.37	5.80
	50	5	9.03	4.97	4.63	6.17	8.03	7.3	6.73	6.73	6.70	8	4.57	3.93	7.97	4.33
	100	7.97	10.10	4.70	8	4.43	8.03	7.97	5.13	4.37	6.43	7.87	4.93	4.8	4.97	2.80
ABC	30	**1.4**	2.53	8.07	2.07	4.53	2	4.23	5.33	6.07	3.60	7.53	3.33	3.93	6.73	4.20
	50	2.47	3.2	7.5	2.53	4.37	3	3.33	4.33	7.17	3.30	6.87	2.77	4.53	7.07	4.67
	100	2.63	2.70	7.23	2.80	6.20	2.97	3.03	5.50	9.43	3.33	6.40	2.60	4.17	7.17	5.47
GABC	30	2.53	2.60	8.83	3.17	2.03	**1**	2.17	2.43	2.57	2.47	5.33	2.43	3.77	5.10	4.83
	50	2.53	2.27	8.80	2.43	2.33	2	2.1	2.33	2.73	2.63	5.93	2.47	2.43	4.83	3.03
	100	2.37	2.43	8.67	2.20	3.37	2.03	2.03	3.23	5.50	4.1	5.77	2.40	2.73	6	3.77
Best-so-far ABC	30	4.97	5	9.47	4	4.7	4	4.63	5.97	7.20	3.87	7.97	4.67	9.33	7.63	9.03
	50	5.97	5.03	9.07	4.83	5.60	4	4.93	5.53	6.77	4.53	8.93	5.73	7.63	7.83	9.33
	100	5.47	5.40	9.10	4.77	7.20	4	6.40	6.57	8.07	6.63	7.40	2.80	8.23	9.13	8.5

Table 7 Continued

Fun / Alg.	D	F_{16}	F_{17}	F_{18}	F_{19}	F_{20}	F_{21}	F_{22}	F_{23}	F_{24}	F_{25}	F_{26}	F_{27}	F_{28}	F_{29}	F_{30}
CCSA	30	**1.33**	**1.10**	**1.40**	**1.07**	**1.20**	**2.50**	**1.17**	**1**	3.07	**1.07**	**1.63**	1.70	**1**	**1.03**	**1.07**
	50	**1.37**	**1.07**	**1.03**	**1**	**1.03**	**1.03**	**1.40**	**1.10**	**1.33**	**1.07**	**1.57**	2.03	**1**	**1**	**1.13**
	100	**1.03**	**1.03**	**1**	**1**	**1.20**	**1**	**1.1**	**1**	**1**	**1**	**1.30**	2	2	**1**	**1**
CSA	30	5.70	5.73	1.60	2.97	6.13	6.40	2.47	7.10	5.37	6.30	3.20	8.90	3.93	7.03	5.40
	50	4.83	5.57	2	4.03	4.40	4.37	2.57	7.17	4.23	5.43	2.90	8.40	5.77	7.03	6.07
	100	5.17	4.30	2	4.03	3.50	4.63	5.80	7.13	7.13	4.37	6.23	8.53	5.07	6.33	4.93
BA	30	9.37	9.63	8.57	8.60	9.87	9.23	9.23	9.83	9.73	10	9.53	**1.57**	7.80	9.83	9.17
	50	9.43	9.87	9.40	9.17	9.27	9.13	8.03	9.70	9.80	10	9.70	**1**	2.7	9.93	9.77
	100	9.5	**9.97**	8.77	9.93	9.37	9.43	6.90	9.70	9.87	10	10	**1**	**1**	9.90	10
CLPSO	30	4.90	3.23	5.23	2.73	3.60	5.43	7.37	5.73	5.90	5.87	6.37	6.67	9	4.73	5.30
	50	6.33	5.47	7.13	4.20	4.17	7.60	8.03	6	4.73	8.27	6.93	6.97	9.83	5.83	5.1
	100	7.87	7.97	8.8	6.23	6.40	6.97	9.53	6	5.97	8.50	6.20	7.17	9.60	7.50	5.73
GWO	30	5.13	5.20	6.97	6.70	5.87	4.47	7.97	4.17	3.73	8.13	6.60	6.23	8.7	4.67	7.90
	50	3.8	3.77	4.67	8.23	4.73	2.60	5.97	2.57	2.07	8.47	5.37	5.87	8.7	4.53	7.93
	100	3.20	2.37	4.10	8.07	3.03	2.03	3.40	4.47	2.90	8.50	2.93	6	7.67	3.33	8.10
EEGWO	30	11	11	11	11	10.9	10.97	11	10.97	11	11	11	11	11	11	11
	50	10.97	11	10.97	11	10.9	11	11	11	10.97	11	11	11	11	11	11
	100	11	11	11	11	11	11	11	11	11	11	10.93	10.97	11	11	11
WOA	30	9.23	8.93	8.53	9.40	8.63	9.67	8.93	8.67	8.67	7.80	9.10	8.87	7.87	9	8.80
	50	9.33	8.97	7.97	8.90	8.73	9.83	9.37	8.70	8.20	7.2	9.27	9.03	7.87	9	8.47
	100	9.43	8.70	4.5	7.47	8.90	9.30	9.10	8.10	8.13	5.70	8.93	8.30	5.93	9.03	8.53
KH	30	8.23	7.90	5.80	7.90	7.87	7.20	2.97	8.30	8.60	6.63	7.70	9.20	4.73	8.03	8.03
	50	7.07	8.03	7.10	7.13	7.73	6.17	7.37	8.43	8.93	4.83	7.77	9.57	5.5	7.9	7.67
	100	6.50	6.50	4.23	5.10	6	8.20	6.97	9.07	8.87	5.37	7.97	10.03	7.40	7.13	6.03
ABC	30	4.03	4.93	6.33	4.60	4.47	3.03	5	3.30	**2.6**	2.57	2.60	4.30	3.07	3.80	2.87
	50	3.7	3.83	5.13	3.90	5.87	4.90	3.83	3.97	6.87	3.1	2.97	4.27	4.07	3.90	2.63
	100	2.93	3.70	7.20	3.17	5.67	4.67	3.57	3.07	3.70	2.97	3.67	4.20	4	3.63	2.93
GABC	30	2.83	3.07	3.90	4.23	3.33	2.60	3.80	3.07	3.63	2.43	3.30	3.23	2.67	2.20	2.07
	50	3.83	3.67	3.80	2.30	4.20	2.97	3.50	3.03	4.30	2	3.8	3.07	2.4	2.13	2.23
	100	3.80	3.73	5.53	2.1	4.97	3.07	3.10	2.03	2.50	2.03	2.77	3.13	3	2.10	2.07
Best-so-far ABC	30	4.23	5.27	6.67	6.80	4.10	4.50	6.10	3.87	3.70	4.20	4.97	4.33	6.23	4.63	4.40
	50	5.33	4.77	6.80	6.13	4.97	6.40	4.93	4.33	4.57	4.63	4.73	4.80	7.17	3.73	4
	100	5.57	6.73	8.87	7.90	5.97	5.70	5.53	4.43	4.93	6.57	5.07	4.67	9.33	5.03	5.67

Table 8 Overall rank by Friedman test for large-scale problems with D=1000

Alg. / Fun	CCSA	CSA	BA	CLPSO	GWO	EEGWO	WOA	KH	ABC	GABC	Best-so-far ABC
F_1	**1**	3.13	7.27	7.73	2	5.87	3.87	5.13	10.40	9.60	10
F_2	**1**	6.67	5.20	7.07	2.87	10.80	2.13	9.93	9	6.07	5.27
F_3	**1**	4.13	8.80	6.13	2.07	9.87	3.07	9.67	8.73	6.33	6.20
F_4	**1**	3	7.47	7.60	2	6	4	5	10.60	9.60	9.73
F_5	**1**	7.20	4.13	6.73	2.87	10.73	2.27	10.2	8.60	6.80	5.47
F_6	**1**	4.07	9.33	6.40	2	9.60	3	9.73	9	6.40	5.47
F_7	**1**	3	7.27	7.80	2	5.93	4	5.07	10	9.53	10.40
F_8	**1**	3.13	7.47	7.53	2	6	3.87	5	10.4	9.67	9.93
F_9	**1**	3.07	7.67	7.33	2	5.93	3.93	5.07	9.93	9.67	10.40
F_{10}	**1**	7.13	4.93	6.47	2.60	10.80	2.40	9.73	8.87	6.60	5.47
F_{11}	**1**	4.07	9.47	6.47	2.07	10.20	2.93	9	8	6.27	5.93
F_{12}	**1**	3	7.13	7.87	2	5.93	4	5.07	10.60	9.47	9.93
F_{13}	**1**	3.20	7.27	7.73	2	5.93	3.80	5.7	10.47	9.80	9.73
F_{14}	**1**	3.13	7.20	7.80	2	5.93	3.87	5.07	10.53	9.33	10.13
F_{15}	**1**	6.87	5.20	6.80	2.87	10.47	2.27	10	8.93	6.27	5.33
F_{16}	**1**	4.20	8.73	6.33	2	9.80	3	9.60	8.93	6.60	5.80
F_{17}	**1**	3	7.47	7.53	2	5.93	4	5.07	10.53	9.67	9.80
F_{18}	**1**	3.07	7.27	7.73	2	6	3.93	5	10.40	9.60	10
F_{19}	**1**	3	7.33	7.67	2	5.93	4	5.07	10.47	9.40	10.13
F_{20}	**1**	3	7.13	7.87	2	6	4	5	10.27	9.33	10.40
Avg. Rank	**1**	4.054	7.19	7.230	2.168	7.6825	3.417	6.96	9.733	8.301	8.276
Overall Rank	**1**	4	6	7	2	8	3	5	11	10	9

Table 9 P-value of Friedman test in CEC 2017 and CEC 2010

CEC 2017								
D	F_1	F_7	F_{10}	F_{15}	F_{20}	F_{25}	F_{28}	F_{30}
30	4.326E-56	2.726E-50	2.734E-50	3.148E-50	5.807E-48	1.419E-53	5.245E-52	7.084E-55
50	8.628E-58	2.112E-54	6.938E-50	1.10E-50	2.837E-42	3.309E-56	1.708E-55	4.043E-56
100	6.074E-58	1.215E-56	1.813E-49	2.549E-53	1.057E-43	9.316E-57	8.912E-58	9.344E-57
CEC 2010								
D	F_1	F_6	F_8	F_{10}	F_{13}	F_{15}	F_{18}	F_{20}
1000	1.9127E-26	9.822E-25	1.817E-26	8.453E-24	1.880E-26	3.338E-23	1.538E-26	1.018E-26

5.6.2. Mean Absolute Error (MAE) test

As a statistical measure of a difference between two continuous variables, mean absolute error (MAE) computed by Eq. (9) shows how far estimates or forecasts are from the actual values.

$$\mathrm{MAE} = \frac{1}{\mathrm{NF}}\sum_{i=1}^{\mathrm{NF}} |O_i - y_i| \tag{9}$$

Where O_i is the global optimum of the function i, NF is the number of functions, and y_i is the best result of the function i obtained by the algorithms. Table 10 shows the results of performance of MAE criteria on the results of tests for different dimensions 30, 50 and 100 in CEC 2017 functions, and dimension 1000 in CEC2010 functions.

Table 10 Mean absolute error in different dimensions

Algorithms	MAE D=30	Rank D=30	MAE D=50	Rank D=50	MAE D=100	Rank D=100	MAE D=1000	Rank D=1000
CCSA	**502.5183**	**1**	**2.2877E+04**	**1**	**1.1356E+08**	**1**	**1.2098E+10**	**1**
CSA	1.9508E+09	4	6.3620E+21	3	4.3961E+84	4	1.3306E+11	3
BA	1.3798E+35	10	7.1697E+66	10	4.7138E+154	9	1.9154E+11	6
CLPSO	3.4927E+16	8	1.9842E+44	7	3.7458E+121	7	2.0192E+11	7
GWO	6.9483E+15	6	1.6994E+37	6	8.1058E+97	5	8.8258E+10	2
EEGWO	2.9539E+41	11	1.6275E+70	11	2.1144E+163	11	1.8734E+11	5
WOA	2.2478E+16	7	6.2187E+44	8	8.6773E+129	8	1.2906E+11	3
KH	8.7571E+19	9	2.2963E+53	9	3.3333E+150	10	1.7328E+11	4
ABC	8.9938E+05	3	3.5308E+22	4	1.0660E+70	3	2.4058E+11	9
GABC	2.2533E+05	2	2.1723E+16	2	6.3377E+61	2	2.3462E+11	8
Best-so-far ABC	1.6096E+11	5	1.7524E+32	5	5.9946E+104	6	2.4161E+11	10

5.7. Impact analysis of the modifications

Thus far, the experimental and statistical results prove that CCSA is superior to other compared algorithms. This section is to analyse the impact of the introduced conscious neighborhood and the strategies NLS, NGS and WAS. As the pseudo code of CCSA shown, firstly, the neighborhood of each crow c_i is generated by Definition 2. Then, contrary to CSA which selects a search strategy for all different problems unconsciously, CCSA uses either NLS or NGS by comparing the quality of the neighborhood with the non-neighborhood. If $fbest_{local} < fbest_{global}$, then the quality of the neighborhood is considered good enough to be exploited using NLS because c_{local} is a random neighbor. Otherwise, CCSA explores the non-neighborhood towards the c_{global} by selecting k dimensions randomly and changing their values using Eq. (6). Eventually, WAS is used to correct the position of the crow c_i if it could not acquire a better fitness after using either NLS or NGS. Therefore, to analyse the impact of our modification, four algorithms CSA, NLS+WAS, NGS+WAS, and CCSA are considered and their behavior is compared for solving different problems.

Fig. 17 shows the average of the best fitness of all crows in each iteration (the average best-so-far) on some functions of CEC 2017 with different dimensions. As the curves showed for unimodal functions F_1 and F_2, the average distance between solutions found by CCSA and

NLS+WAS is less than others which is evident the impact of using NLS in the exploitation. Meanwhile, for multi-modal functions F_5 and F_9 which have a large number of local optima, the average distance between solutions found by CCSA and NGS+WAS is less than others which is evident the impact of using NGS in the exploration. The average distance between solutions found by CCSA and other algorithms for hybrid benchmark functions F_{18} and F_{20}, and for composition function F_{22} show that CCSA is better than NLS+WAS and NGS+WAS in term of the balance between local and global search and premature convergence. In fact, applying the introduced conscious neighborhood for selecting search strategy and using strategies NLS, NGS and WAS enhance CCSA for solving different problems.

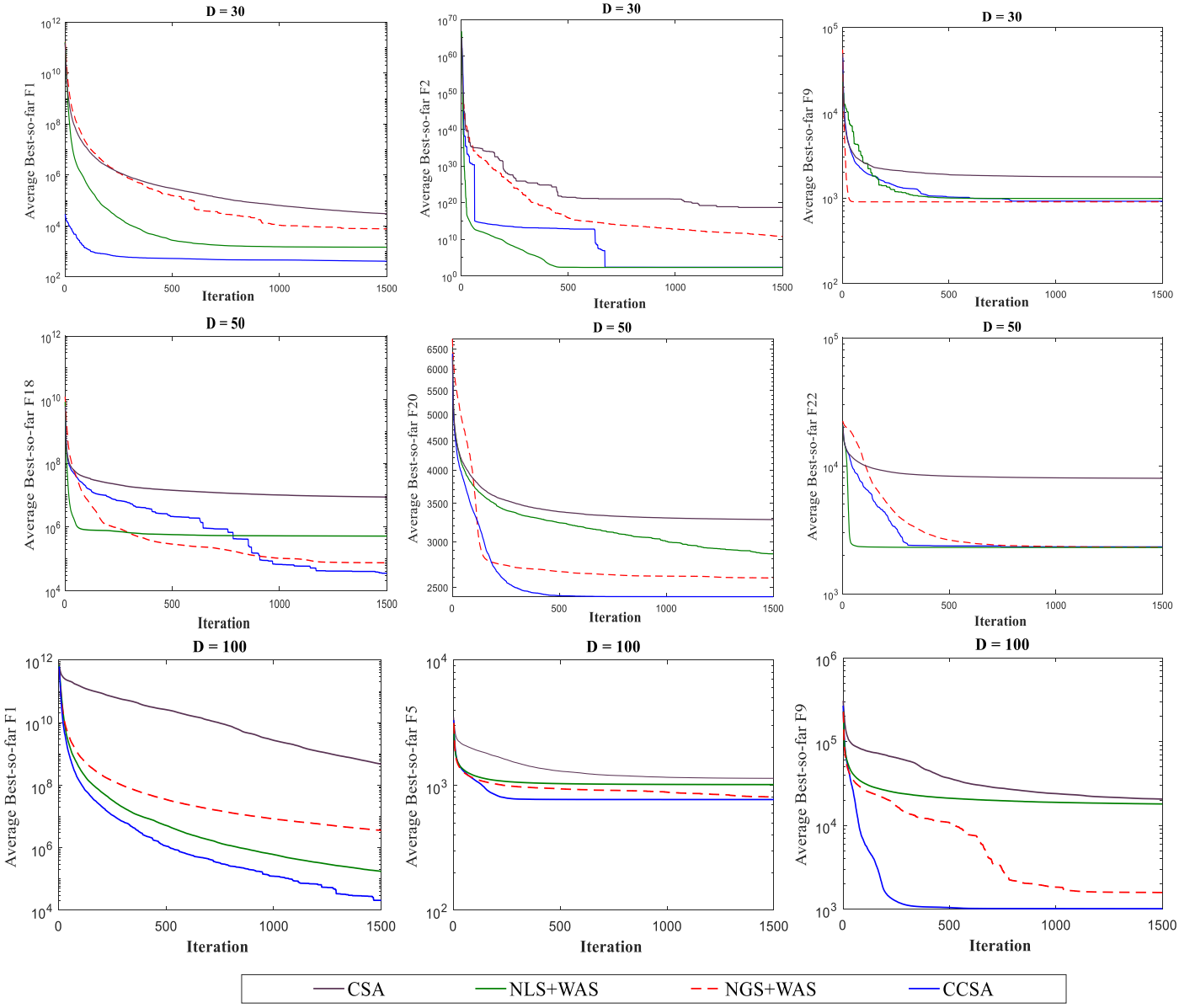


Fig. 17 The average distance between solutions found by CSA, NLS+WAS, NGS+WAS and CCSA

5.8. Solving applied problems of engineering design using CCSA

Applicability of the optimization algorithms for real-world applications is usually evaluated by real-world problems such as engineering design problems [11,64,65]. Therefore, in this section, the applicability of CCSA is evaluated and compared with other state-of-the-art swarm intelligence algorithms using four engineering design problems (described in the Appendix) as follows:

F_1: Parameter estimation for frequency modulated (FM) sound waves

F_2: Pressure vessel design problem

F_3: Three-bar truss problem

F_4: Welded beam design problem

In this experiment, the population size (N), the maximum number of iterations (MaxIt) and the number of the run were set to 200, 1500 and 30 respectively. Tables 11-14 show that CCSA finds the best optimal values for variables of these four engineering problems compared to other algorithms.

Table 11 Parameter estimation for frequency modulated (FM) sound wave

Algorithm	Optimal values for variables						Optimum cost
	a_1	ω_1	a_2	ω_2	a_3	ω_3	
CCSA	1.0000	5.0311	1.5000	-4.8000	-2.0000	4.9000	**2.7889E-11**
CSA	0.0195	-2.4574	-1.5678	1.6408	-5.7668	4.5536	14.1112
BA	0.3357	5.0981	0.4667	5.4750	4.7526	-5.0051	23.1104
CLPSO	1.0485	4.9866	-1.4922	4.8086	-2.0223	-4.9030	0.2039
GWO	1.0038	4.9995	-1.4934	4.8002	-1.9993	-4.8998	0.0013
EEGWO	0.5160	5.0089	-0.7298	0.6853	-2.7159	4.6224	24.9337
WOA	0.5715	-5.0077	3.2597	-4.8165	0.0050	-6.3046	13.3999
KH	-0.6615	0.0381	3.8776	0.5511	-1.8855	-3.2441	16.2655
ABC	-1.0208	-4.9949	-1.4431	-4.8001	2.0372	-4.8994	0.0633
GABC	-0.9992	-5.0011	-1.5005	-4.7992	1.9994	-4.9003	5.1475E-04
Best-so-far ABC	1.0048	4.9944	-1.4800	4.8029	2.0029	4.8997	0.0088

Table 12 Results for pressure vessel design problem

Algorithms	Optimal values for variables				Optimum cost
	T_s	T_h	R	L	
CCSA	0.7782	0.3847	40.3197	199.9991	**5.8853E+03**
CSA	4.5792	80.599	81.430	118.9119	7.7613E+03
BA	0.9284	0.4589	48.1000	114.1757	6.1950E+03
CLPSO	0.7903	0.3905	40.8171	194.0763	5.9425E+03
GWO	0.7782	0.3848	40.3203	200.0000	5.8860E+03
EEGWO	1.9370	0.9962	79.0153	10.0000	1.8012E+04
WOA	0.7833	0.3872	40.5855	196.3311	5.8943E+03
KH	1.0397	0.5247	53.2363	74.6464	6.6129E+03
ABC	0.7786	0.4082	40.3266	200.0000	5.8897E+03
GABC	0.7782	0.3848	40.3231	199.9534	5.8858E+03
Best-so-far ABC	0.7786	0.3850	40.3251	199.9855	5.8898E+03

Table 13 Results for the three-bar truss problem

Algorithms	Optimal values for variables		Optimal weight
	x_1	x_2	
CCSA	0.78865625	0.40830170	**2.63895844E+02**
CSA	0.69660225	0.88149073	2.63897198E+02
BA	0.78871035	0.40814873	2.63895848E+02
CLPSO	0.78872952	0.40809450	2.63895848E+02
GWO	0.78868348	0.40822490	2.63895864E+02
EEGWO	0.78898946	0.40736223	2.64283957E+02
WOA	0.78667553	0.41393360	2.63898802E+02
KH	0.90058694	0.06744589	2.64062751E+02
ABC	0.78858225	0.40861455	2.63896340E+02
GABC	0.78876674	0.40798940	2.63895865E+02
Best-so-far ABC	0.78836577	0.40912463	2.63895976E+02

Table 14 Results of the welded beam design problem

Algorithms	Optimal values for variables				Optimum cost
	h	l	t	b	
CCSA	0.2057	3.4702	9.0362	0.2057	**1.7249**
CSA	0.1628	8.1441	1.2523	1.3875	1.8162
BA	0.1971	3.7889	8.7538	0.2193	1.8052
CLPSO	0.2058	3.4736	9.0260	0.2062	1.7275
GWO	0.2057	3.4721	9.0366	0.2057	1.7251
EEGWO	0.1778	6.4368	9.0584	0.2314	2.2859
WOA	0.2070	3.5250	8.9756	0.2085	1.7450
KH	0.1573	4.8676	9.3590	0.2050	1.8747
ABC	0.2077	3.4707	8.9381	0.2110	1.7506
GABC	0.2011	3.5423	9.1140	0.2058	1.7317
Best-so-far ABC	0.1983	3.6412	9.0351	0.2058	1.7364

6. Discussion

This section aims to analyze the experimental results of CCSA and relate them to its structure and operators. The results and convergence curves shown in Table 2 and Fig. 12 prove the superiority of the proposed algorithm on the majority of unimodal benchmark functions. The main reason for this merit of CCSA in the exploitation and convergence rate is using the local search based on the introduced conscious neighborhood. Moreover, CCSA uses the introduced NLS for local search when for a random neighbor c_{local} $fbest_{local} < fbest_{global}$ which means its neighbors can be considered as candidate solutions and be exploited.

The results and curves in Table 3 and Fig. 13 show that CCSA benefits from high exploration for multi-modal functions which have a large number of local optima. This is mostly because when $fbest_{local} \geq fbest_{global}$ which means the chance of having a candidate solution among the neighbor is very low then CCSA explores the search space to find a promising zone. Meanwhile, the CCSA algorithm increases the exploration by changing only k dimensions toward c_{global} which makes it far from the greedy fashion.

As the results in Tables 4 and 5 and the convergence curves of Figs. 14 and 15 shown, the proposed algorithm is superior to the compared algorithms for hybrid and composite benchmark functions. Because of having the characteristic of maintaining the continuity around the local and global optima in these functions, these results prove that CCSA increases the balance between global and local search and reduces the premature convergence. The main reason is that CCSA uses introduced conscious neighborhood which increases its focus around the candidate solutions. Moreover, it uses the introduced WAS to correct the position of those crows that could not acquire a better fitness by using either NLS or NGS in order to escape from local optima. The convergence curves presented in Figs. 12-15 guarantee convergence of CCSA because they show the fitness of crows is decreased over the course of iterations.

Besides the experimental results, the statistical results tabulated in Tables 7- 10 reveal that the proposed algorithm is statistically significant as compared to the compared algorithms. The results in Tables 11-14 show that CCSA outperforms the compared algorithms for solving applied problems of engineering design. As a summary, the overall performance comparison is shown in Table 15.

Table 15 Overall comparison of the proposed CCSA with other state-of-the-art swarm intelligence algorithms

Categories of swarm intelligence algorithms	**Insects behavior**			**Terrestrial animal behavior**		**Marine animal behavior**		**Bird behavior**			
Algorithms / Properties	ABC [22]	GABC [14]	Best-so-far ABC [23]	GWO [20]	EEGWO [21]	KH [2]	WOA [11]	BA [18]	CLPSO [13]	CSA [17]	CCSA
Local searchability	Low	High	High	High	High	High	High	High	High	High	High
Global searchability	High	High	High	Low	High	High	High	Low	High	High	High
Balance between local & global search	Low	High	High	Low	Low	High	Low	Low	High	Low	High
Premature convergence	High	High	High	High	High	High	High	High	High	High	Low
High dimension ability	Low	Low	Low	Low	High	High	Low	Low	Low	Low	High

7. Conclusions and future works

The intelligent behavior of crows inspires the crow search algorithm (CSA) in order to solve optimization problems. However, it selects a search strategy unconsciously by comparing a constant value of awareness probability (AP) with a random number. Consequently, it mostly suffers from lacking a proper balance between local and global search and having premature convergence, especially in large-scale problems. Moreover, its efficiency for solving some multi-modal, hybrid and composition problems are not sufficient enough. To tackle these weaknesses, in this paper, an improved version of CSA named conscious neighborhood-based crow search algorithm (CCSA) was proposed.

In the proposed algorithm, a new neighborhood concept is defined to perceive the search space and select local and global search strategies consciously. Furthermore, the movement of crows in the search space is improved using three new strategies NLS, NGS, and WAS. In addition, WAS recognizes the crows located in the flat or local optima and provides another opportunity for them by changing their position during the different jump-flies. The experiment results and relevant discussions support the following conclusions:

- Using the introduced conscious neighborhood enhances the exploitation and the balance between local and global search.
- Using introduced NLS and NGS strategies improve the ability of exploitation and exploration.
- The introduced WAS strategy increases the balance between global and local search and reduces premature convergence.
- CCSA is more efficient than the compared algorithm for different unimodal, multi-modal, hybrid and composition problems in several dimensions.
- CCSA is also superior to the compared algorithms for solving large-scale global optimization problems.
- The proposed CCSA is applicable for solving engineering design problems.

CCSA is developed for single-objective and continuous problems. Therefore, multi-objective and binary versions of this algorithm may be developed as future works for solving multi-objective and discrete problems. Moreover, using CCSA for solving optimization problems in different applications and domains can be another valuable future work.

Appendix

Four engineering problems used in Section 5.7 are described as follows.

F_1: Parameter estimation for frequency modulated (FM) sound wave [66]

FM is a highly complex multimodal problem with strong epistasis that generates a sound similar to the target. This problem has a parameter vector X= $[a_1, \omega_1, a_2, \omega_2, a_3, \omega_3]$ with six dimension and the minimum fitness value is $f(\vec{x}_i) = 0$. To estimate the sound wave, it uses the following Eqs. (A.1) and (A.2) where $\theta = \frac{2\pi}{100}$ and the parameters are set between [-6.4, 6.35]. In addition, the fitness function is computed by Eq. (A.3).

$$y(t) = a_1 \times \sin(\omega_1 \times t \times \theta + a_2 \times \sin(\omega_2 \times t \times \theta + a_3 \times \sin(\omega_3 \times t \times \theta))) \quad (A.1)$$

$$y_0(t) = (1.0) \times \sin((5) \times t \times \theta - (1.5) \times \sin((4.8) \times t \times \theta + 2 \times \sin((4.9) \times t \times \theta))) \quad \text{(A.2)}$$

$$F(\vec{x}) = \sum_{t=0}^{100} (y(t) - y_0(t))^2 \quad \text{(A.3)}$$

F_2: Pressure vessel design (PVD) problem [67]

In PVD problem, the objective function is to minimize the total cost, including the cost of the material, forming and welding shown in Fig. A.1. In this problem, there are four decision following variables: x_1 is the thickness of the shell (T_s), x_2 is the thickness of the head (T_h), x_3 is the inner radius (R), and x_4 is the length of the cylindrical section of the vessel, not including the head (L). This problem with four optimization constraints is formulated by Eq. (A.4).

$$\begin{aligned}
&\text{Consider} && \vec{x} = [x_1 x_2 x_3 x_4] = [T_s\ T_h\ R\ \ L], \\
&\text{Min} && f(\vec{x}) = 0.6224 x_1 x_3 x_4 + 1.7781 x_2 x_3^2 + 3.1661 x_1^2 x_4 + 19.84\, x_1^2 x_3, \\
&\text{Subject to} && g_1(\vec{x}) = -x_1 + 0.0193 x_3 \le 0, \\
& && g_2(\vec{x}) = -x_2 + 0.00954 x_3 \le 0, \\
& && g_3(\vec{x}) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1{,}296{,}000 \le 0, \\
& && g_4(\vec{x}) = x_4 - 240 \le 0, \\
&\text{Variable range} && 0 \le x_i \le 100, \quad i = 1,2 \\
& && 10 \le x_i \le 200 \quad i = 3,4
\end{aligned} \quad \text{(A.4)}$$

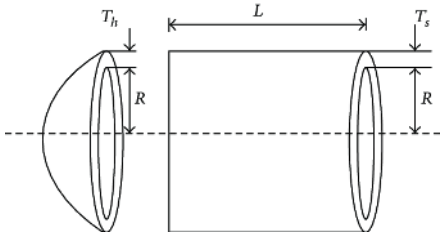


Fig. A.1 Design of pressure vessel problem

F_3: Three-bar truss problem

In this problem, the volume of a statistically loaded three-bar truss is to be minimized. The schematic of the three-bar truss problem is shown in Fig. A.2. The formulation of this optimization problem is computed by Eq. (A.5).

$$\begin{aligned}
&\text{Min} && f(\vec{x}) = \left(2\sqrt{2x_1} + x_2\right) \times l, \\
&\text{Subject to} && g_1(\vec{x}) = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}\,x_1^2 + 2x_1 x_2} P - \sigma \le 0, \\
& && g_2(\vec{x}) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1 x_2} P - \sigma \le 0,
\end{aligned} \quad \text{(A.5)}$$

$$g_3(\vec{x}) = \frac{1}{\sqrt{2}x_2 + x_1} P - \sigma \leq 0,$$

Variable range $0 \leq x_i \leq 1, \qquad i = 1,2$

$\mathrm{l} = 100\mathrm{cm}, \mathrm{P} = 2\ \mathrm{kN/cm^2}, \text{and } \sigma = 2\ \mathrm{kN/cm^2}$

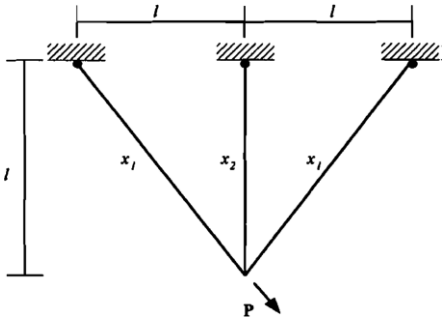


Fig. A.2 Three-bar truss problem

F_4: Welded beam design (WBD) problem [67]

In this problem, there are four decision variables including h(x_1) is the thickness of weld, l(x_2) is the length of the clamped bar, t(x_3) is the height of the bar, and b(x_4) is the thickness of the bar. Fig. A.3 represents the schematic of the welded beam. The objective function is designed for the minimum the fabrication cost subject to constraints on shear stress (t), bending stress in the beam (u), buckling load on the bar (P_c), end deflection of the beam (d). The formulation of the WBD problem is computed by Eq. (A.6).

Consider $\vec{x} = [x_1 x_2 x_3 x_4] = [h \quad l \quad t \quad b],$ (A.6)

Min $f(\vec{x}) = 1.10471 x_1^2 x_2 + 0.04811 x_3 x_4 (14.0 + x_2),$

Subject to $g_1(\vec{x}) = \tau(\vec{x}) - \tau_{max} \leq 0,$

$g_2(\vec{x}) = \sigma(\vec{x}) - \sigma_{max} \leq 0,$

$g_3(\vec{x}) = \delta(\vec{x}) - \delta_{max} \leq 0,$

$g_4(\vec{x}) = x_1 - x_4 \leq 0,$

$g_5(\vec{x}) = P - P_c(\vec{x}) \leq 0,$

$g_6(\vec{x}) = 0.125 - x_1 \leq 0,$

$g_7(\vec{x}) = 1.10471 x_1^2 + 0.04811 x_3 x_4 (14.0 + x_2) - 5.0 \leq 0,$

Variable range $0.1 \leq x_i \leq 2, \qquad i = 1,4$

$0.1 \leq x_i \leq 10 \qquad i = 2,3$

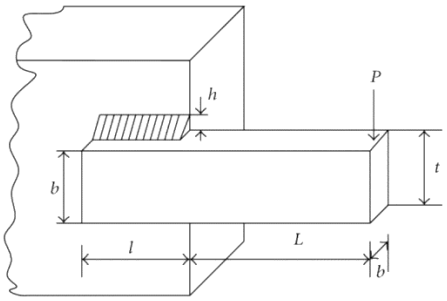


Fig. A.3 Welded beam design (WBD) problem

Acknowledgment

The authors would like to thank anonymous reviewers for their comments and suggestions.

Declarations of interest: none.

Ethical approval: This article does not contain any studies with animals performed by any of the authors.

8. References

[1] A.K. Kar, Bio inspired computing – A review of algorithms and scope of applications, Expert Syst. Appl. 59 (2016) 20–32.
[2] A.H. Gandomi, A.H. Alavi, Krill herd: A new bio-inspired optimization algorithm, Commun. Nonlinear Sci. Numer. Simul. 17 (2012) 4831–4845.
[3] E.-G. Talbi, Metaheuristics: from design to implementation, John Wiley & Sons, New Jersey, USA, 2009.
[4] C. Blum, J. Puchinger, G.R. Raidl, A. Roli, Hybrid metaheuristics in combinatorial optimization: A survey, Appl. Soft Comput. 11 (2011) 4135–4151.
[5] M. Dorigo, G. Di Caro, Ant colony optimization: a new meta-heuristic, Proc. 1999 Congr. Evol. Comput. (Cat. No. 99TH8406). 2 (1999) 1470–1477.
[6] S. Taghian, M.H. Nadimi-Shahraki, H. Zamani, Comparative Analysis of Transfer Function-based Binary Metaheuristic Algorithms for Feature Selection, in: 2018 Int. Conf. Artif. Intell. Data Process., 2018: pp. 1–6.
[7] A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
[8] L. dos Santos Coelho, P. Alotto, Gaussian artificial bee colony algorithm approach applied to Loney's solenoid benchmark problem, IEEE Trans. Magn. 47 (2011) 1326–1329.
[9] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, MHS'95. Proc. Sixth Int. Symp. Micro Mach. Hum. Sci. (1995) 39–43.
[10] D. Karaboga, C. Ozturk, A novel clustering approach: Artificial Bee Colony (ABC) algorithm, Appl. Soft Comput. 11 (2011) 652–657.
[11] S. Mirjalili, A. Lewis, The Whale Optimization Algorithm, Adv. Eng. Softw. 95 (2016) 51–67.
[12] J.C. Bansal, H. Sharma, S.S. Jadon, M. Clerc, Spider monkey optimization algorithm for numerical optimization, Memetic Comput. 6 (2014) 31–47.

[13] J.J. Liang, A.K. Qin, P.N. Suganthan, S. Baskar, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions, IEEE Trans. Evol. Comput. 10 (2006) 281–295.
[14] G. Zhu, S. Kwong, Gbest-guided artificial bee colony algorithm for numerical function optimization, Appl. Math. Comput. 217 (2010) 3166–3173.
[15] Y. Shi, C.-M. Pun, H. Hu, H. Gao, An improved artificial bee colony and its application, Knowledge-Based Syst. 107 (2016) 14–31.
[16] D. Karaboga, B. Gorkemli, A quick artificial bee colony (qABC) algorithm and its performance on optimization problems, Appl. Soft Comput. 23 (2014) 227–238.
[17] A. Askarzadeh, A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm, Comput. Struct. 169 (2016) 1–12.
[18] X.-S. Yang, A New Metaheuristic Bat-Inspired Algorithm, Nat. Inspired Coop. Strateg. Optim. (NICSO 2010). 284 (2010) 65–74.
[19] X.-S. Yang, A. H. Gandomi, Bat algorithm: a novel approach for global engineering optimization, Eng. Comput. 29 (2012) 464–483.
[20] S. Mirjalili, S. Mirjalili, A.L.-A. in engineering Software, U. 2014, Grey wolf optimizer, Elsevier. 69 (2014) 46–61.
[21] W. Long, J. Jiao, X. Liang, M. Tang, An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization, Eng. Appl. Artif. Intell. 68 (2018) 63–80.
[22] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Glob. Optim. 39 (2007) 459–471.
[23] A. Banharnsakun, T. Achalakul, B. Sirinaovakul, The best-so-far selection in Artificial Bee Colony algorithm, Appl. Soft Comput. 11 (2011) 2888–2901.
[24] N.H. Awad, M.Z. Ali, J.J. Liang, B.Y. Qu, P.N. Suganthan, Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective bound constrained real-parameter numerical optimization, Nanyang Technological University Singapore, 2016.
[25] K. Tang, X. Li, P.N. Suganthan, Z. Yang, T. Weise, Benchmark functions for the CEC 2010 special session and competition on large-scale global optimization: Nature Inspired Computation and Applications Laboratory, University of Science and Technology of China, 2009.
[26] L. Guo, G.-G. Wang, A. H. Gandomi, A. H. Alavi, H. Duan, A new improved krill herd algorithm for global numerical optimization, Neurocomputing. 138 (2014) 392–402.
[27] S. Mirjalili, Introduction to Evolutionary Single-Objective Optimisation, in: Evol. Algorithms Neural Networks Theory Appl., Springer International Publishing, Cham, 2019: pp. 3–14.
[28] M. Tian, X. Gao, C. Dai, Differential evolution with improved individual-based parameter setting and selection strategy, Appl. Soft Comput. 56 (2017) 286–297.
[29] D.E. Goldberg, J.H. Holland, Genetic algorithms and machine learning, Mach. Learn. 3 (1988) 95–99.
[30] J.R. Koza, Genetic programming as a means for programming computers by natural selection, Stat. Comput. 4 (1994) 87–112.
[31] H.-G. Beyer, H.-P. Schwefel, Evolution strategies-A comprehensive introduction, Nat. Comput. 1 (2002) 3–52.
[32] R. Storn, K. Price, Differential Evolution-A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, J. Glob. Optim. 11 (1997) 341–359.
[33] X. He, Y. Zhou, Enhancing the performance of differential evolution with covariance matrix self-adaptation, Appl. Soft Comput. 64 (2018) 227–243.
[34] M. Tian, X. Gao, Differential evolution with neighborhood-based adaptive evolution

mechanism for numerical optimization, Inf. Sci. (Ny). 478 (2019) 422–448.
[35] Z. Meng, J.-S. Pan, QUasi-Affine TRansformation Evolution with External ARchive (QUATRE-EAR): An enhanced structure for Differential Evolution, Knowledge-Based Syst. 155 (2018) 35–53.
[36] Z. Meng, J.-S. Pan, K.-K. Tseng, PaDE: An enhanced Differential Evolution algorithm with novel control parameter adaptation schemes for numerical optimization, Knowledge-Based Syst. 168 (2019) 80–99.
[37] I. Boussaïd, J. Lepagnot, P. Siarry, A survey on optimization metaheuristics, Inf. Sci. (Ny). 237 (2013) 82–117.
[38] A.H. Gandomi, X.S. Yang, S. Talatahari, A.H. Alavi, Metaheuristic Algorithms in Modeling and Optimization, Elsevier, 2013.
[39] P.-W. Tsai, J.-S. Pan, B.-Y. Liao, S.-C. Chu, Interactive artificial bee colony (iabc) optimization, in: ISI2008, 2008.
[40] S. Aslan, H. Badem, D. Karaboga, Improved quick artificial bee colony (iqABC) algorithm for global optimization, Soft Comput. (2019) 1–22.
[41] S. Saremi, S. Mirjalili, A. Lewis, Grasshopper Optimisation Algorithm: Theory and application, Adv. Eng. Softw. 105 (2017) 30–47.
[42] S. Mirjalili, The Ant Lion Optimizer, Adv. Eng. Softw. 83 (2015) 80–98.
[43] S. Mirjalili, Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems, Neural Comput. Appl. 27 (2016) 1053–1073.
[44] S. Mirjalili, Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm, Knowledge-Based Syst. 89 (2015) 228–249.
[45] B. Wang, X. Jin, B. Cheng, Lion pride optimizer: An optimization algorithm inspired by lion pride behavior, Sci. China Inf. Sci. 55 (2012) 2369–2389.
[46] G. Dhiman, V. Kumar, Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications, Adv. Eng. Softw. 114 (2017) 48–70.
[47] M. Jain, V. Singh, A. Rani, A novel nature-inspired algorithm for optimization: Squirrel search algorithm, Swarm Evol. Comput. 44 (2019) 148–175.
[48] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (2017) 163–191.
[49] A. Kaveh, N. Farhoudi, A new optimization method: Dolphin echolocation, Adv. Eng. Softw. 59 (2013) 53–70.
[50] H. Zamani, M.-H. Nadimi-Shahraki, Feature Selection Based on Whale Optimization Algorithm for Diseases Diagnosis, Int. J. Comput. Sci. Inf. Secur. 14 (2016) 1243.
[51] M. Abd Elaziz, D. Oliva, Parameter estimation of solar cells diode models by an improved opposition-based whale optimization algorithm, Energy Convers. Manag. 171 (2018) 1843–1859.
[52] M. Abdel-Basset, L. Abdle-Fatah, A.K. Sangaiah, An improved Lévy based whale optimization algorithm for bandwidth-efficient virtual machine placement in cloud computing environment, Cluster Comput. 21 (2018) 1–16.
[53] Y. Ling, Y. Zhou, Q. Luo, Lévy flight trajectory-based whale optimization algorithm for global optimization, IEEE Access. 5 (2017) 6168–6186.
[54] S. Shadravan, H.R. Naji, V.K. Bardsiri, The Sailfish Optimizer: A novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems, Eng. Appl. Artif. Intell. 80 (2019) 20–34.
[55] Z. Beheshti, S.M. Hj. Shamsuddin, CAPSO: Centripetal accelerated particle swarm optimization, Inf. Sci. (Ny). 258 (2014) 54–79.
[56] Y. Chen, L. Li, H. Peng, J. Xiao, Y. Yang, Y. Shi, Particle swarm optimizer with two

differential mutation, Appl. Soft Comput. 61 (2017) 314–330.
[57] H. Shi, S. Liu, H. Wu, R. Li, S. Liu, N. Kwok, Y. Peng, Oscillatory Particle Swarm Optimizer, Appl. Soft Comput. 73 (2018) 316–327.
[58] K. Zhang, Q. Huang, Y. Zhang, Enhancing comprehensive learning particle swarm optimization with local optima topology, Inf. Sci. (Ny). 471 (2019) 1–18.
[59] Q. Liu, L. Wu, W. Xiao, F. Wang, L. Zhang, A novel hybrid bat algorithm for solving continuous optimization problems, Appl. Soft Comput. 73 (2018) 67–82.
[60] X.-S. Yang, Suash Deb, Cuckoo Search via lévy flights, 2009 World Congr. Nat. Biol. Inspired Comput. (2009) 210–214.
[61] A.H. Gandomi, X.-S. Yang, A.H. Alavi, Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems, Eng. Comput. 29 (2013) 17–35.
[62] X.-B. Meng, X.Z. Gao, L. Lu, Y. Liu, H. Zhang, A new bio-inspired optimisation algorithm: Bird Swarm Algorithm, J. Exp. Theor. Artif. Intell. 28 (2016) 673–687.
[63] J. Derrac, S. García, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, Swarm Evol. Comput. 1 (2011) 3–18.
[64] X. Xia, L. Gui, Z.-H. Zhan, A multi-swarm particle swarm optimization algorithm based on dynamical topology and purposeful detecting, Appl. Soft Comput. 67 (2018) 126–140.
[65] H. Liu, Z. Cai, Y. Wang, Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization, Appl. Soft Comput. 10 (2010) 629–640.
[66] S. Das, P.N. Suganthan, Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems, 2010.
[67] C.A. Coello Coello, Use of a self-adaptive penalty approach for engineering optimization problems, Comput. Ind. 41 (2000) 113–127.