Applied Intelligence (2022) 52:17990–18029
https://doi.org/10.1007/s10489-022-03397-4

The water optimization algorithm: a novel metaheuristic for solving optimization problems

Arman Daliri [1] · Ali Asghari [1] · Hossein Azgomi [2] · Mahmoud Alimoradi [1]

Accepted: 15 February 2022 / Published online: 7 April 2022

Abstract
Metaheuristic algorithms (MAs) are used to find the answers to NP-Hard problems. NP-Hard problems basically refer to a set of optimization problems that cannot be solved in a polynomial at a time. MAs try to find the optimal or near-definitive answer in the shortest possible time to solve such problems and a set of optimization algorithms with different origins. These algorithms may be inspired by the natural sciences, physics, mathematics, and political science. However, a particular Metaheuristic algorithm may not provide the best answer to all problems. Each MA may have a better response to specific problems than other similar algorithms. Therefore, researchers will try to introduce and discover new algorithms to find optimal answers to a wide range of problems. In this paper, a new Meta-heuristic algorithm called the Water optimization algorithm (WAO) is presented. WAO is inspired by the chemical and physical properties of water molecules. The main idea of the proposed algorithm is to link water molecules together to find the optimal points. Factors such as particle motion, particle evaporation, and particle bonding have created a mechanism based on swarm intelligence and physical intelligence that inspired this algorithm to solve persistent problems. In this algorithm, answers are defined as a water molecule, a set of them is defined as a local answer. Water bonds provide the right move towards the optimal response. In evaluating the performance of the proposed algorithm, the proposed method is applied to some standard functions and some practical problems. The results obtained from the experiments show that the proposed algorithm has provided appropriate and acceptable answers in terms of execution time and accuracy compared to some similar algorithms.

Keywords Optimization · Metaheuristic · Continuous problems · Hydrogen bonding of water algorithm

1 Introduction

In the theory of computational complexity, there are different types of problems.The computational complexity of problems starts from P-time (P) problems [1] which is the simplest model of the problem and ends with the most challenging model of problems. A problem is non-deterministic polynomial acceptable problems-Hard (NP-Hard), if solved in polynomial time, would make it possible to solve all problems in class NP in polynomial time [2]. NP-Hard Problems include thousands of problems, each of which has many applications in the engineering sciences. To date, no definitive answer has been found for such issues [3]. One of the essential applications of Metaheuristic algorithms is to answer NP-Hard problems reasonably and with a near-definite answer [3]. The proposed Meta-heuristic algorithms are inspired by various sources that operate according to that source.

Metaheuristic algorithms include a set of algorithms that are used in optimization problems. These algorithms are a good solution in cases where there is no definitive solution to the problem or the time to reach a definitive answer is too long. In fact, instead of reaching a definite answer, these algorithms obtain acceptable answers in a more appropriate time. These algorithms often have exploitation and exploration phases, the first of which is to maintain and enhance the achievements of this phase, and the second is to seek better answers in other areas that may not have been explored. The search is done in two models, local and global, in the search space. Some of these algorithms are swarm-based and rely on several initial solutions that are improved in subsequent iterations in proportion to the fitness function of the problem. The population is usually constant until the end of the algorithm.

✉ Ali Asghari
asghari_ali@aut.ac.ir

[1] Department of Computer Engineering, Shafagh Institute of Higher Education, Tonekabon, Iran

[2] Department of Computer Engineering, Rasht Branch, Islamic Azad University, Rasht, Iran

The optimal initial population size and the number of algorithm iterations play an essential role in finding appropriate answers in an acceptable time. Metaheuristic algorithms optimize NP-Hard problems. The optimization performed in these methods depends on searching the sample space of the answers.

Each Metaheuristic algorithm is designed by a source of inspiration, which distinguishes it from other methods in the same category. Algorithms are usually inspired by Nature, mathematics, physics, political and economic sciences, medical and evolutionary sciences [4]. For example, one of the first Meta-heuristic algorithms is the Genetic Algorithm (GA) [5]. Genetic algorithms were initially designed to solve problems with discrete data. The approach of this algorithm to solve problems is optimization [5] Another version of this algorithm was later implemented to solve continuous problems [6]. Another algorithm in this field is Genetic Programming (GP) that was designed by David Goldberg [6–8]. The mentioned algorithms (GA, GP) have some functional similarities in some cases, such as generation and recombination operators.

Although this algorithm is based on the theory of evolution, it differs from the GA algorithm in its answers. For example, different tree-based answers are displayed in this algorithm, which is suitable for modeling [7]. Another algorithm called Ant Colony Optimization (ACO) is known as meta-heuristic algorithms [8]. This algorithm is inspired by the collective behavior of ants [8]. Another widely used algorithm in mechanics is the Spotted Hyena Optimizer (SHO) algorithm [9]. SHO is a convenient algorithm for solving problems in the online domain. The mass hunting behavior of hyenas inspires this algorithm to achieve near-optimal responses. At a higher level of algorithms for solving problems, hyper-heuristic algorithms are presented [10]. Hyper-heuristic algorithms are usually built by combining several heuristics or meta-heuristic algorithms [10]. In some such algorithms, the Machine Learning (ML) method has been used [11]. Hyper-heuristics that have used ML are usually presented in a limited way to solve a practical problem [12].

Although Metaheuristic algorithms seek to find near-optimal answers to NP-Hard problems, according to the NFL theory, not every algorithm can solve all problems [13]. Nor can it be said that a Metaheuristic algorithm solves a problem in the best way. The main challenge in Metaheuristic algorithms is solving problems in the shortest possible time with the utmost accuracy in optimizing NP-Hard problems. Metaheuristic algorithms try to perform well in improving response time, obtaining an optimal or near-optimal response, especially in population-based and congestion methods that can perform well with a minimal population of initial responses. Population size is an essential challenge in this area, so its increase leads to a long time to reach its response, and its decrease leads to a drop in local optimality and early convergence. For example, an algorithm such as ACO has high accuracy in solving the tsp. problem, but its response time is longer than similar algorithms in solving this problem. For this reason, algorithms must converge to a good point both in terms of accuracy and speed of performance. Hyper-heuristic algorithms also face problems by combining several different algorithms. The use of a combination of Metaheuristic algorithms has limited them to solving a specific set of problems.

In this paper, a new Meta-heuristic algorithm is presented that uses the chemical and physical properties of water molecules. Hydrogen bonds of water metaheuristic algorithm (WAO) use the relationship between water molecules and are the optimal answer to problems. The proposed algorithm uses the chemical properties of the water molecule and the hydrogen bonds of this molecule to perform its local search. It also conducts a global search using the laws of physics and the molecular motion of water. In addition, using the application of magnetic forces, the water atom pursues a global search in a sample space. The steps to find the optimal points are obtained with the help of the weight of the bonded molecules inspired by the hydrogen bond of water. In this algorithm, each water molecule is one of the desired answers to search for the best answer.

The source of inspiration for this algorithm is the use of the chemical and physical properties of water. Population-based algorithms are highly dependent on the size of the population and the number of iterations required to achieve the optimal or near-optimal response. A smaller population size will cause a fall in local optimum or early convergence. In contrast, increasing population size will increase response time. Therefore, there will always be a trade-off between the accuracy of the response and the timing. Due to its unique nature, our proposed algorithm requires a smaller population compared to similar methods, which in itself reduces the execution time and increases the accuracy of achieving a more appropriate response. The reason for this is the use of P and HTemp lowering actuators, which are inspired by the physical and chemical properties of the water atom. When applying the P process, new responses are obtained from the bond between members of the population, which, if better fitted, will replace the pre-impact responses.

On the other hand, the P process is applied randomly and in proportion to the quality of the initial responses. This work is done not only to reinforce superior responses but also to enable weaker responses to avoid falling into local optimization. In the P process, the appropriate responses from the collision become a superior response, and fewer and better-improved responses follow the population calculations. The HTemp operator, which is formed by the nature of the boiling point of water, also increases the search speed when needed. In other words, the acceleration of the algorithm is proportional to the progress made, increasing the response speed.The performance of the WAO algorithm is due to the use of chemical and physical properties of water, such as a combination of several Metaheuristics, each of which makes this algorithm more responsive.

The main advantages of the proposed method include the following:

- Design a new metaheuristic algorithm called WAO optimization algorithm, inspired by chemical and physical properties of water molecules.
- Introducing two new P and HTemp factor operators to speed up response time and high accuracy in obtaining the optimal response.
- Utilizing the strengths of several meta-heuristic algorithms.
- Ability to solve continuous NP-Hard problems.
- We are maintaining the quality of responses with a smaller population of initial responses compared to similar methods.
- Comparing the performance of the proposed method and its superiority over a variety of classic and new methods.

The other sections of the article are organized as follows:

In Section 2, the related works are reviewed. Then, in Section 3, the proposed algorithm and its mechanism are introduced. In the fourth part of this research, the performance of the proposed algorithm is evaluated, and finally, in the 5th part, the proposed method is concluded.

2 Related works

Many Metaheuristic algorithms have been developed in recent years. The main task and purpose of their design are to obtain the optimal answer [14]. This section has a detailed look at the process of some classic and new models, how to build them, and a partial history of them. According to the sources of inspiration of each Metaheuristic algorithm, there are categories for them, which generally include Evolutionary Algorithms [15], Physical algorithms [16], Nature algorithms [17], Swarm algorithms [18], and Biological Algorithms [19].

The theory of evolution generally inspires Evolutionary Algorithms. One of the first Evolutionary algorithms is the Genetic Algorithm (GA) [5]. GA is a convenient algorithm for solving problems with discrete data. The GA algorithm is proposed using the generation of chromosomes and the genetic laws of the process. This algorithm was introduced in 1970 by John H. Holland. Another algorithm that can be named in this category is the Differential Evolution Algorithm (DE) [20]. The DE algorithm was introduced to improve the results and problems of the genetic algorithm. GA is an algorithm for solving problems with discrete data, so DE is designed to improve performance for solving problems with GA continuous data. Another algorithm in this category that the CEC2020 test has evaluated is the Improved Unified Differential Evolution (IUDE) algorithm [21]. This algorithm has been formed using a combination of physical and evolutionary algorithms and has improved performance. Other algorithms such as Genetic programming [7], evolutionary strategy [22], and Evolutionary programming [23] are in this category.

The next category of algorithms is Physical algorithms. These algorithms are generally inspired by the science of physics [15]. The simulated annealing (SA) algorithm is one of these that was introduced in 1983 [24]. The SA algorithm is a Meta-heuristic optimization algorithm suitable for large search spaces. Due to its high efficiency and simplicity, it is widely used and popular among various methods, and this algorithm is also suitable for solving discrete problems [24]. Another algorithm that works well in the physical category is the Teaching-learning based optimization (TLBO) algorithm [25]. The rules of learning and teaching inspire this algorithm. Another algorithm called a gravitational search algorithm (GSA) uses physical and gravitational laws in this category [26]. The GSA was introduced in 2009. The Archimedes optimization algorithm (AOA) is a new method that falls into this category. AOA was introduced in 2020 and is inspired by Archimedes' law [16]. This algorithm mimics the principle of Archimedes law of floating force applied upwards on an object, in which it moves partially or completely according to the weight of the fluid immersed in the liquid [16]. Another new algorithm in this category is called the Transient search optimization (TSO) algorithm. This algorithm is inspired by the use of electrical switching circuits such as inductors and capacitors and transient behavior in such circuits.TSO was introduced in 2020 [27]. Another algorithm that has been designed in this field and built using several algorithms is the sCMAgES algorithm. This algorithm is designed and presented in 2020 [28]. (HS) harmony search [29], memetic algorithm (MA) [30], and electromagnetic algorithm an electromagnetism-like mechanism (EM) [31] can also be named from this category, each of which has its characteristics.

Nature-inspired algorithms are another category [17]. In this category, we can mention the Firefly algorithm (FF), which is inspired by the optical connection between fireflies [32]. FF was introduced in 2007. Another algorithm called the Cuckoo search (CS) was introduced in 2013, inspired by cuckoo behavior [33]. The cuckoo algorithm is a suitable algorithm for solving persistent problems. Another widely used algorithm in this category is the Harris Hawks optimization (HHO) algorithm. This algorithm is inspired by the Harris hawk bird hunting process and deals with optimization issues. In this algorithm, Harris hawks can implement various hunting scenarios and models depending on the type of Nature or prey, using collective intelligence when hunting. The HHO algorithm is presented in 2019 [34]. One of the newest algorithms in this field is the Trees Social Relations Optimization

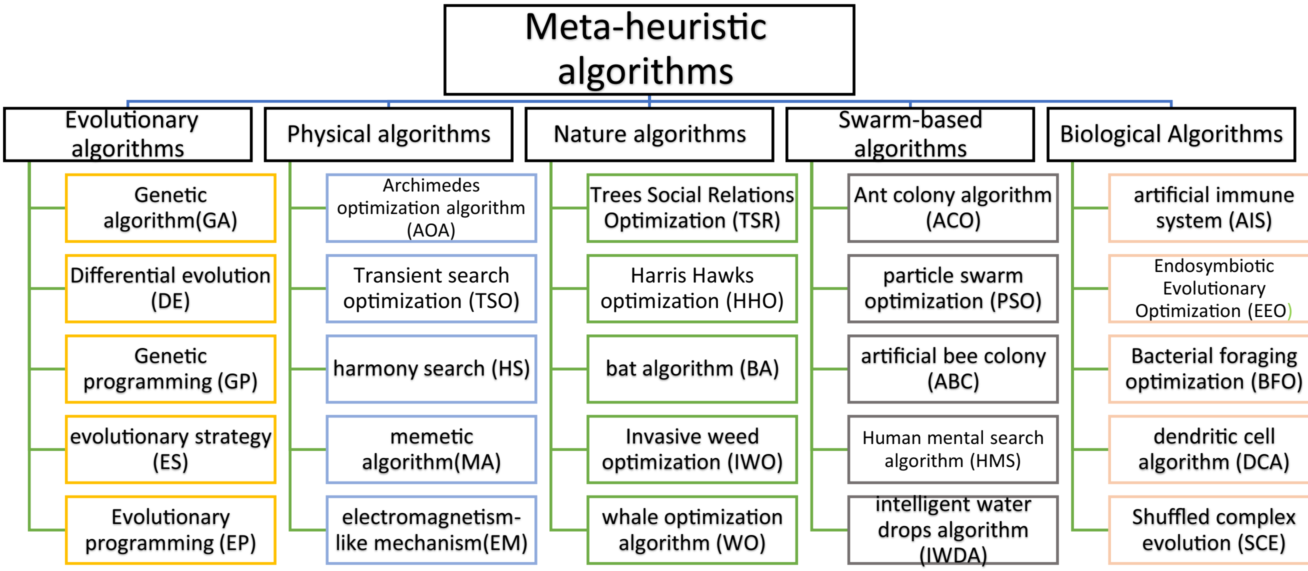


Fig. 1 Classification of metaheuristic algorithms

Algorithm: A new Swarm-Based metaheuristic technique to solve continuous and discrete optimization problems (TSR), which uses the collective intelligence behavior of trees to optimize. This algorithm was published in 2021 [35]. Furthermore, Biogeography-based optimization (BBO) [36], gray wolf optimizer (GWO) algorithms [37], moth flame optimization algorithm (MFO) [38], the ant lion optimizer (ALO) [39], grasshopper optimization algorithm (GOA) [40], (BA) bat-inspired Algorithm [41], Invasive weed optimization algorithm (IWO) [42] and The whale optimization algorithm (WO) [43] are other algorithms in this category.

Swarm algorithms usually use swarm intelligence that can solve many problems [18]. One of the most popular algorithms in this category is called the Ant colony algorithm (ACO). The ACO was introduced in 1992 [8]. The behavior of ants inspires ACO to find food. In this algorithm, agents alone do not have intelligence, but their collective behavior has led to swarm intelligence. Another algorithm called particle swarm optimization (PSO) was introduced in 1995 and fell into this category [44]. PSO is a convenient algorithm for solving persistent problems inspired by birds' social behavior, and many researchers refer to it as the bird algorithm. Other algorithms in the population-based category include the quasi-oppositional chaotic antlion optimizer (QOCALO) algorithm. The source of inspiration for this algorithm is antlion hunting, and Antlions hunt for insects on the surface of the soil using hollows such as ant colonies holes, which is the main source of this algorithm. This algorithm is used to solve persistent problems [45]. The artificial fish swarm optimization algorithm aided by ocean current power (AFSAOCP) is also one of the algorithms inspired by swarm intelligence. This algorithm uses ocean currents to optimize problems in fish hunting behavior, which is individual and collective [46]. Another algorithm in the category of swarm intelligence is the Human mental search algorithm (HMS). This algorithm is inspired by various social behaviors, such as the online auction bidding space. HMS was introduced in 2017 and is suitable for solving persistent problems [47]. Another algorithm in this category is the COLSHADE algorithm, which was designed in 2020 using the combination of the nature of several other super-innovative algorithms [48]. Another successful algorithm in this field is A Self-Adaptive Spherical Search Algorithm (SASS), which was introduced in 2020 [49]. This algorithm is also designed using a combination and improvement of other Metaheuristic algorithms such as Spherical Search (SS). Artificial bee colony algorithm (ABC) [50], artificial fish swarm optimization algorithm (AFSO) [51], intelligent water drops algorithm (IWD) [52], and SCA: A Sine Cosine algorithm (SCA) [53] are other collective search algorithms.

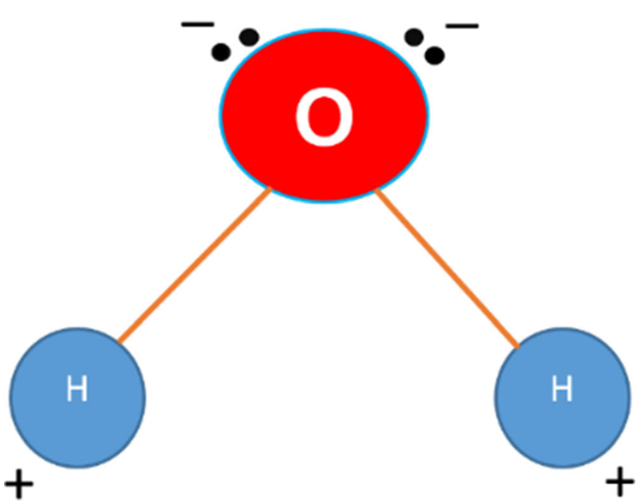


Fig. 2 Water atom and its different poles

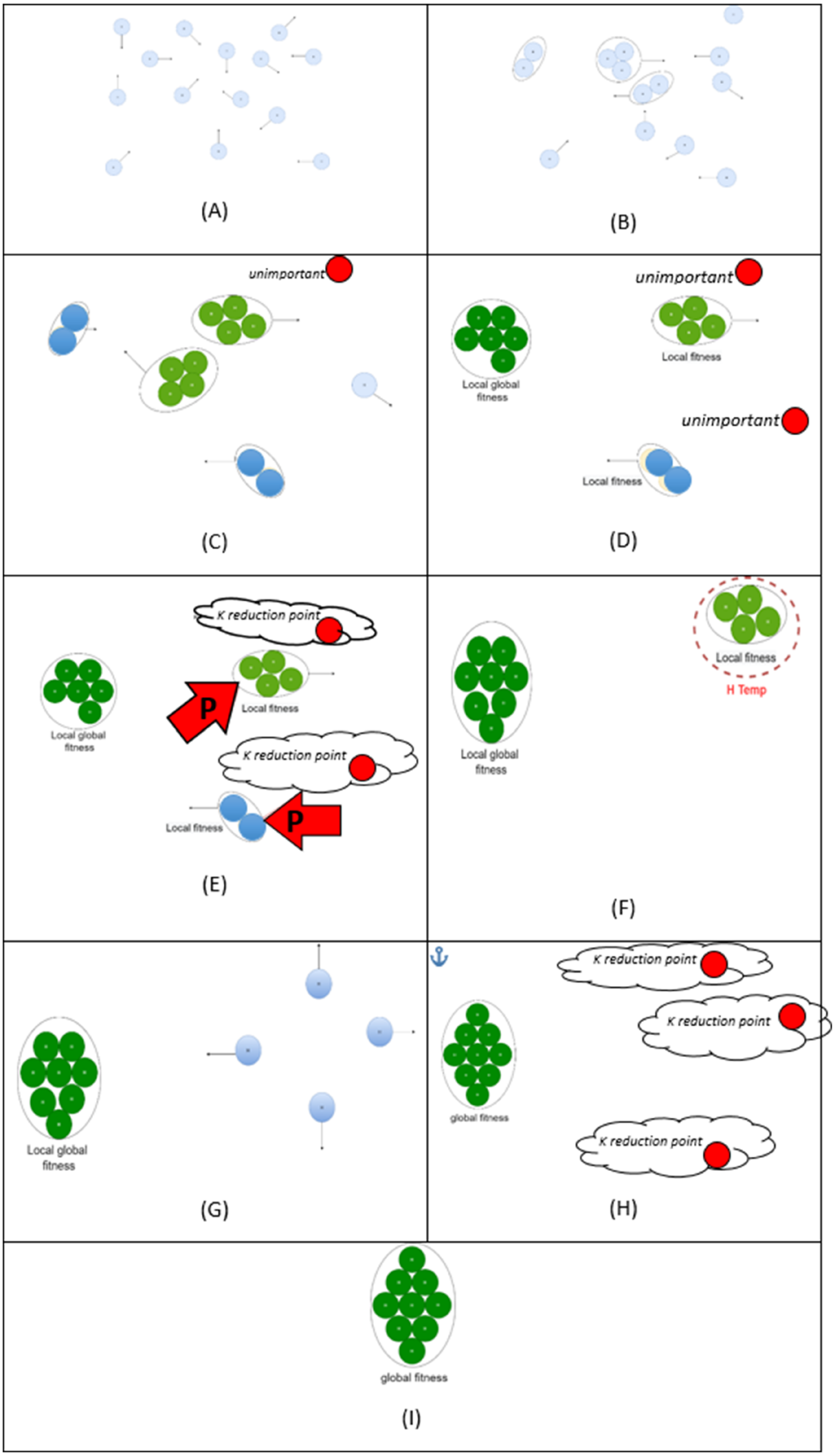


Fig. 3 Steps of WAO algorithm

Table 1 WAO algorithm parameters

Parameter name	Description
Afit	Average optimality
B	link
FitG	Optimal overall
FitL	Local optimization
FitLG	The best local optimizer
H	Particle
HTemp	Boiling point
K	Temperature reduction coefficient
P	strike
PL	Location of each particle
S	Initial speed
Temp	Sample space temperature
Wh	Weight
θh	Initial angle
θNh	The new angle of motion
θp	Impact angle

Biological sources inspire the algorithms presented as Biological Algorithms. The immune system of living things and bacteria growth can be the source of inspiration for these algorithms [19]. For example, one of the most well-known algorithms in this category is called the artificial immune system (AIS). AIS was introduced in 2010 and is inspired by a biological immune system [54]. Another well-known algorithm in this category is the krill herd algorithm (KH) introduced in 2014 [55]. KH has used chaos theory to accelerate the achievement of optimal points. One of the most popular algorithms in this field is the An Endosymbiotic Evolutionary Optimization (EEO) algorithm introduced in 2001. Inspired by endosymbiotic behavior, This algorithm is inspired by the evolution of eukaryotic cells and the combination of prokaryotic and eukaryotic cells for this evolution., this algorithm is optimized for survival and life [56]. Other biological algorithms include the Bacterial foraging optimization (BFO) algorithm [57], The dendritic cell algorithm (DCA) [58], the salp swarm algorithm (SSA) [59], and the Shuffled evolution algorithm. (SCE) [1] also mentioned. Figure 1 shows the classification of Metaheuristic algorithms categorized according to their sources of inspiration and performance.

A review of past work in Metaheuristic methods shows that each of these methods has made reasonable efforts to improve

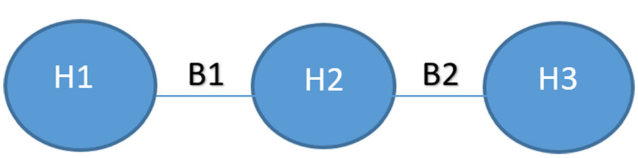


Fig. 4 Hydrogen bonds and weight - H1, H2, H3 water molecules - B1, B2 Hydrogen bonds

Table 2 Initial random parameters

	X_1	X_2	Fitness
$H1_1$	0.5	0.5	0.5
$H2_1$	1	1	1
$H3_1$	1.5	1.5	4.75
$H4_1$	2	2	8
$H5_1$	2.5	1.5	8.5
$H6_1$	0.5	1.5	2.5

the optimal answers. Metaheuristic methods never claim to be the best solution to all problems, so there is still a motivation to find new methods in this area. In most Metaheuristic methods, which are based on population and repetition, choosing the size of the population is a challenge. Less population size leads to early convergence and, in contrast to population, increases time complexity. Therefore, to overcome this challenge, the proposed method is introduced. The use of HTemp and P parameters in the proposed method results in better initial responses and requires a smaller population to reach the optimum point. P factor increases accuracy, and the HTemp factor increases the speed of reaching the target. The details of the algorithm are explained in the relevant sections.

3 Hydrogen bonds of water metaheuristic algorithm (WAO)

In this section, Hydrogen bonds of the water metaheuristic algorithm (WAO) are fully introduced. First, the source of inspiration for this algorithm is pointed out. The working process of the proposed algorithm is presented, and at the end, the parameters of this algorithm are explained. Section 3.1 provides the source of inspiration for this algorithm. Section 3.2 describes the operation of the algorithm. After presenting the proposed algorithm, Section 3.3 defines the parameters and mathematical modeling of this algorithm. In Section 3.4, the real problems are solved using the proposed algorithm. Section 3.5 describes the WAO algorithm and shows its various steps with a flowchart. Finally, the proposed method is described in Section 3.6.

Table 3 Particles and initial values calculation

Particles						
	$H1_1$	$H2_1$	$H3_1$	$H4_1$	$H5_1$	$H6_1$
PL	(0.5,1.5)	(2.5,1.5)	(2,2)	(1.5,1.5)	(1,1)	(0.5,0.5)
S	1	1	1	1	1	1
Wh	18	18	18	18	18	18
θ	270°	270°	270°	0°	180°	90°

3.1 Inspiration from water

The properties of the water molecule inspire the WAO algorithm. One of the essential properties of the water molecule is its chemical polarity. It means that each molecule of water can absorb molecules close to it. Another advantage of this property is that its stability will be higher when each molecule is joined together. Another feature of this molecule is its physical properties. Therefore, the WAO algorithm can answer problems with continuous data. The physical-chemical properties of water molecules inspire the WAO algorithm. Water molecules can bond hydrogen with each other [60]. The water molecule (H2O) is a quadrilateral atom resulting from two covalent bonds of two hydrogens with one oxygen, shown in Fig. 2, given the water molecule [61]. The covalent bond in the water molecule causes it to polarize. When a molecule becomes polarized, it will act like a charged object, absorbing each other if two charged objects with opposite charges come close to each other. Hydrogen bonds form this property in the water molecule [60, 62].

The molecular mass of each molecule is used to measure its mass. This value is measured in moles per kilogram. In materials science, each water molecule's molar mass is equal to 18 mol per gram [63]. Another property of water is that it evaporates at room temperature, which is equal to 24 °C. Its boiling point is equal to 100 degrees Celsius, at which temperature the water begins to change and evaporate. If heat more than 100 degrees Celsius enters the water molecule, it only causes the atoms of this molecule to move faster and more lively [64]. As the level of kinetic energy in this molecule increases, hydrogen bonds are separated, and new bonds may be formed.

3.2 Operation process of WAO algorithm

The WAO algorithm is presented using the chemical-physical properties of the water molecule. Figure 3 provides an overview of the performance of this algorithm; according to Section A of Fig. 3, points are first created randomly in the problem space. According to the source of inspiration, these particles are the same as water molecules. Each particle weights by default and also has a random speed and direction of motion. As long as the water molecules are moving, a hydrogen bond will form as each molecule collides with the other, which in section B, the bonds are separated by lines. Bonding two (or more) particles form a new point, with the total weight of all the bonded particles at that point.

As shown in Section C of Fig. 3, as the bond increases at each point, particles will form that is considered to be the local optimal point. As the algorithm continues in Section D, different points are created, each with different weight and unused and unused points. When this happens, the local search of the algorithm is over. In this part, local optimal points are also formed, which may have the same or different values. To find the best point up to this point in the algorithm, we calculate the weight of each point and display the best point with the maximum weight obtained. In the proposed algorithm, the algorithm process proceeds at each stage with the probability that the best local optimal is the global answer. All points are also given a chance to be optimal.

According to Section E of Fig. 3, the algorithm searches for a better response by tapping the local optimal points. This impact enters the local optimal points with a random direction and random velocity but does not enter the best local optimal point in this iteration. As shown in this section, one of the local optimal points is linked to the best local point of the previous step, but the other local optimal point is far from the best local optimal point. Also, the points that were not connected were removed from the problem space to search for optimality with a temperature reduction coefficient. In the next section of Fig. 3 (Section F), 2 points are formed, which is more likely to be optimal. Therefore, the energy of the molecular movement of water has been used to give a chance to the global search to the optimal local points. Such a way that the points that have not gained a better weight during the process of the previous section are given a temperature of 100 degrees, which causes the local optimal point to overlap, which throws every particle from that point in a random direction and at a random speed. This trend is shown in Section G of Fig. 3.

Eventually, new points may be formed, or the global point may be strengthened, and more low points may be formed. For example, in section H of Fig. 3, by applying molecular motion to the local optimal point, three insignificant particles were formed in the previous section, and one of the particles succeeded in searching for the optimal point. When this happens, the insignificant points are removed from the problem space with the coefficient of temperature decrease, and the

Table 4 The second step of the first iteration of the WAO algorithm

Initial values	H1,H2,H6	H3,H4
PL	1.5	2.5
S	1–1+1=1	1+1=2
Wh	3*18=54	2*18=36
θ	270°	315°

Table 5 The second step of the first iteration of the WAO algorithm

Parameter	H1,H2,H6, H3
PL	0.5
S	1–1+1–1=0
Wh	4*18=72
θ	315°

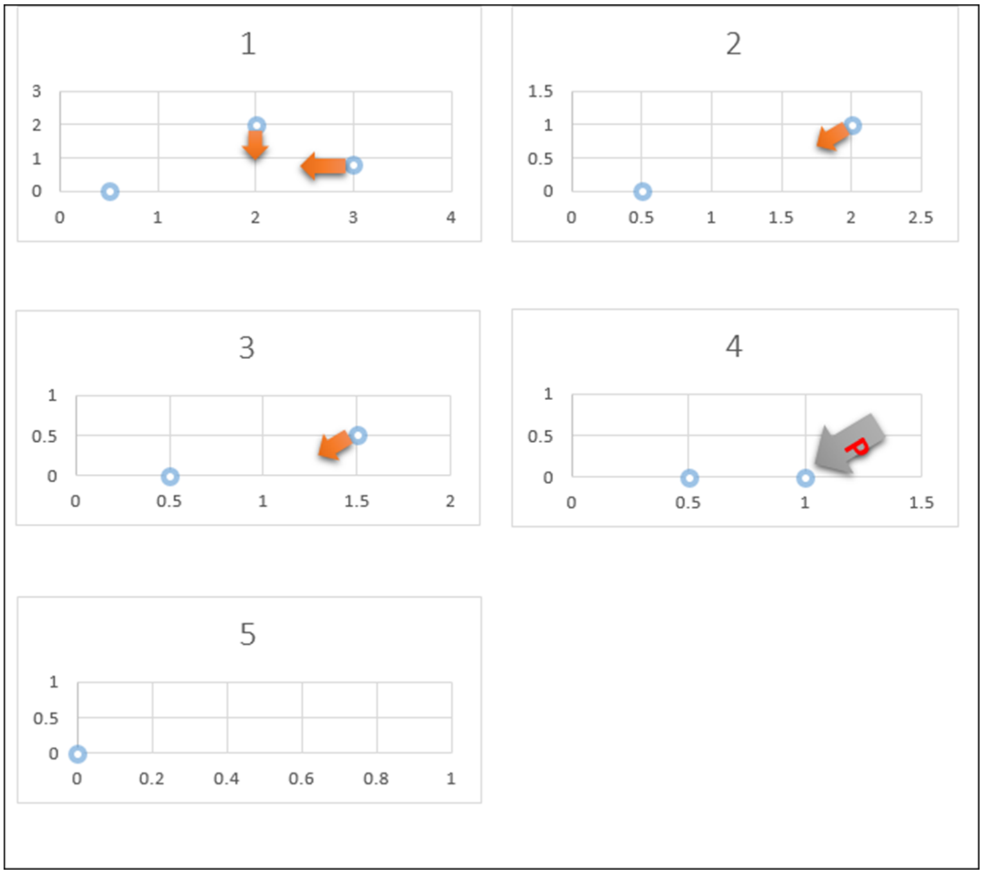


Fig. 5 Steps of WAO algorithm (1.H3 and H4 are linked together, 2. They continue to move, 3. Approaching the best local optimization, 4. The P operator enters them, 5. The best answer and the most links are achieved)

optimal point also appears, which is shown in the final part of Fig. 3 (Section I) of the optimality of this step. After finding the optimality in each step, depending on the condition of the algorithm, the said steps will be repeated.

3.3 Mathematical parameters and formulas

This section introduces the parameters of the WAO algorithm Table 1. According to the general trend of the algorithm, the first parameter that we need to know is the generated responses or the same particle space of the sample problem represented by H. In the proposed method, and the desired points are loaded in a sample space. H particles are moving, which includes the direction of motion and the speed of motion. The direction of motion in this algorithm is θh. This value is set randomly. In two-dimensional space, the value is between 0 and 360 degrees. The velocity parameter in the first step of the algorithm process is constant and equal to 1 for all particles and Sh. The amount of speed and direction of movement at each point varies as the algorithm continues. So by obtaining the velocity fit for the two points that collide with each other, the new value Sh will be determined. The particles of the algorithm have specific coordinates for movement, which will be set depending on the problem, and are represented by the PL parameter.

The second parameter of this algorithm is called bonding, where b represents hydrogen bonding. This parameter generates several other parameters that play an essential role in the overall process of the algorithm. This parameter is calculated in the algorithm so that if the distance between H1 and H2 is zero, then a hydrogen bond is formed between two particles. This parameter generates new weight and angle parameters, denoted by (Wh and θNh), respectively. The weight of the bonds formed is specified by the Wh parameter. When two H atoms join together, a B bond is formed between them. We quantify the weight of the particles according to the bond. For example, if we have three atoms, there are two bonds between them, which also weigh 3, shown in Fig. 4.

After gaining new weight, the new angle of motion must be determined. This is because the particles are constantly moving. The parameter θnewh (θNh) is used to check the direction of motion when two points collide. This value is obtained using elementary laws of physics and mathematics. So that the difference in the angle of motion of the heavier particle with the lighter particle is equal to the new value of the direction of motion, more than two particles may collide with each other. Equation 1 is then used to obtain a new angle of motion (Eq. 1).

$$\theta\mathrm{Nh} = \sum_{i=1}^{n} (\theta_i) \qquad (1)$$

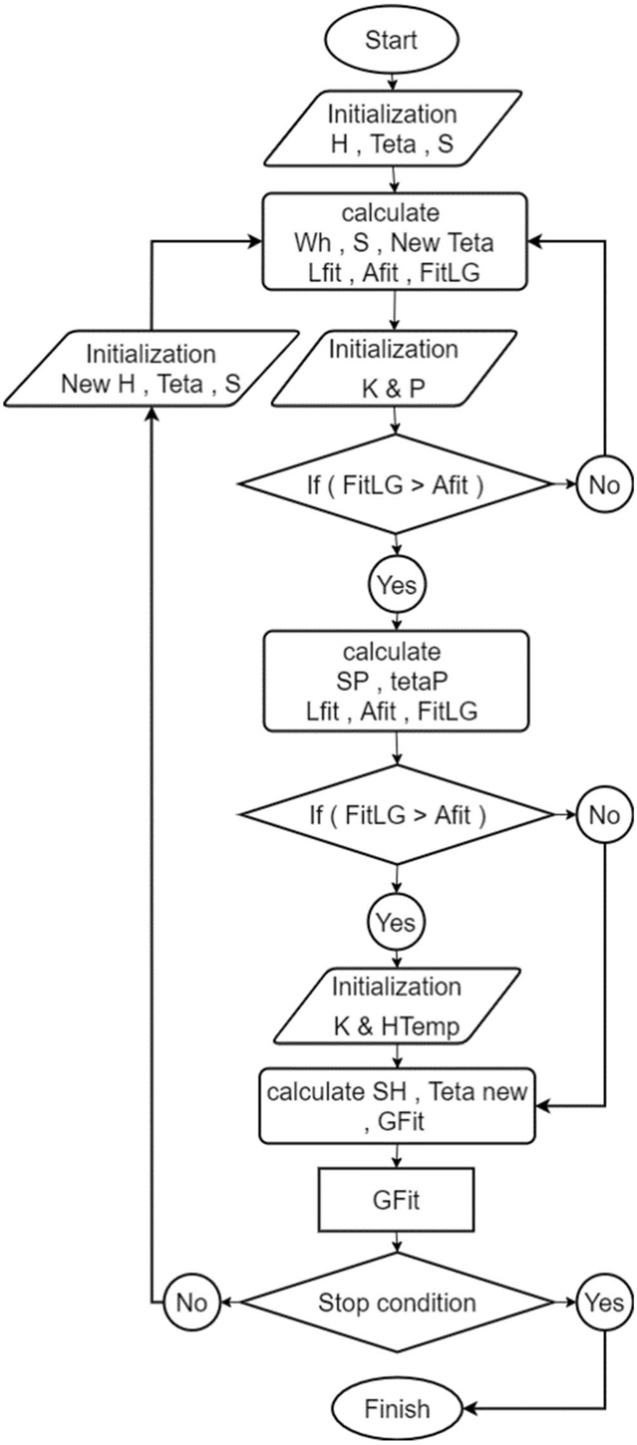


Fig. 6 Flowchart of WAO algorithm

According to the physical relations, after gaining weight and updating the weights, the movement of the points will slow down and eventually become static. The parameters of local optimal weight (FitL) and the average of all FitLs are then calculated to calculate the best local optimal point represented by (AFit). By comparing Afit with all points, we find out which local points are optimal and which points are worthless. Worthless points will be eliminated according to the algorithm process, but FitL points will be hit. It is also called the largest local optimizer (FitLG), which will not be hit at any stage. Using the Impact parameter denoted by (P), the impact points are entered for the global search. P introduces a new force that includes the angle of impact and the speed of impact to any point. The parameters of impact angle and impact speed are equal (θp and Sp), respectively. Their value is entirely random to create fair conditions for the implementation of the steps. Calculating the speed and direction of scattering and scattering of optimal particles is done in the previous steps. Not all points may be searched after applying changes by tapping.

Furthermore, there is a possibility that there will be no change in the answers. The boiling temperature has been used to solve this problem; this is displayed with the parameter (H-temp). This operation occurs when there is no change in the previous optimizations. In this operation, the local optimal particle temperature rises so high that it causes the optimal points to overlap at a random rate for each particle. This process must then be performed to the value of the algorithm stop condition. The stop condition may be the number of repetitions, the time, the convergence, which are bet as desired.

The mathematical relationships of the WAO algorithm are given in the explanation of the parameters. The mass of atoms is calculated using the molar mass, and the mass of one mole of water is 18 g per mole [60] (Eq. 2). As a result, the wh of each particle is 18 mol/g by default. If the molecules are considered Nh and the mass of each molecule is 18 g per mole, then the mass value of each optimal point can be calculated with (Eq. 3).

$$H_2O = 2H + O = \ 2(1) + 16 = 18\ {}^{g}/_{Mol} \tag{2}$$

$$W_h = \sum_{Nh=1}^{n} (18 * Nh) \tag{3}$$

The following parameter that is addressed is the angle of motion. To calculate the angle, we may have an angle number in the motion of N particles. For this reason, in the first step, the difference in angles with each other must be calculated. This difference will be either a positive number in the range of zero degrees to 360 degrees or a negative number below 360 degrees. If this number is in the range of zero to 360 degrees, the result is used as an angle. Otherwise, add the obtained angle with 360 degrees to reach the first positive number in the range of zero degrees to 360 degrees. Equation 4 calculates the angle formula. In this relation, θ1 is the previous angle, and θ2 is the new angle of motion. Equation 4 continues until θ-total.

$$\theta new = \theta_1 - \theta_2 \Rightarrow \ \theta_{total} = 360 + (-\theta new) \tag{4}$$

Table 6 Hyperparameters of algorithms

Algorithms	Parameters	Values
Water Optimization Algorithm (WAO)	Population size	50
	Number of generations	1000
Biogeography-based Optimization (BBO)	Population size	50
	Number of generations	1000
Transient search optimization(TSO)	Number of Elites	2
	Population size	50
	Number of generations	1000
	t	Iteration Counter
Harris hawks optimization (HHO)	Population size	50
	Number of generations	1000
	j	0
Grasshopper Optimization Algorithm (GOA)	Population size	50
	Number of generations	1000
	Search Agent Position	Eq.
	Value of Coefficient	c
Moth-Flame Optimization (MFO)	Convergence constant	[−1,-2]
	Logarithmic spiral	0.75
	Search agents	100
	Number of generations	1000
Salp Swarm Algorithm (SSA)	Search Agent	50
	Boundary No	ub
	Number of generations	1000
Sine Cosine Algorithm (SCA)	Number of elites	2
	Number of generations	1000
	Search agent	100
Whale Optimization Algorithm (WA)	Number of generations	1000
	Parameter b	1
	Initial population	100
The Ant Lion Optimizer (ALO)	Boundary No	ub
	Number of generations	1000
	Search agents	50
Archimedes optimization algorithm(AOA)	Population size	50
	Number of generations	1000
	Update for best answer(L)	L+8

The particle velocity must be checked after obtaining the angle of motion. In classical physics relations, the higher the weight, the lower the speed without acceleration and constant [45]. This is done by combining classical physics (Newton's laws) with velocity formulas. Given that the particles are moving at the theta angle of motion, a new velocity is obtained by taking a simple result between the speeds measured at each point connected to the other point. . For example, if the weight of each point is Wh and the velocity of each point is Sh, then using Newton's laws, Eq. 5 can be formed, which determines the speed of each point. In Eq. 5, two forces are shown, with $F_{1.2}$ being a one-to-two force and $F_{2.1}$ The force is two to one. Finally, by taking the result between the velocities, the velocity of other points will be obtained in Eq. 6.

Table 7 Unimodal benchmark functions

Function	Function name	Dim	Range	Fmin
$f1(x)= \sum_{i=1}^{n} x_i^2$	Sphere	30	[−5.12,5.12]	0
$f2(x)= \sum_{i=1}^{n} x_i^2 + random[0.1]$	Sphere randomic	30	[−5.12,5.12]	0
$f3(x)= \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	Schwefel 2.22	30	[−100,100]	0
$f4(x)= \sum_{i=1}^{n} i x_i^2$	Sum Squares	30	[−10,10]	0
$f5(x)= \sum_{i=1}^{n-1} \left(x_i^2\right)^{x_{i+1}^2+1} + \left(x_{i+1}^2\right)^{x_i^2+1}$	Brown	30	[−1,4]	0
$f6(x)= 1+\sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} cos\left(\frac{x_i}{\sqrt{i}}\right)$	Griewank	30	[−600,600]	0
$f7(x)= - exp\ (-0.5 \sum_{i=1}^{n} x_i^2\ )$	Exponential	30	[−1,1]	0
$f8(x)= exp(-\sum_{i=1}^{n} (x_i/15)^{10} - 2exp\left(-\sum_{i=1}^{n} x_i^2\right) \prod_{i=1}^{n} cos^2(x_i)$	Xin-She Yang N. 3	30	[−2π,2π]	0
$f9(x)= \sum_{i=1}^{n} x_i^2 + \left(\sum_{i=1}^{n} 0.5 i x_i\right)^2 + \left(\sum_{i=1}^{n} 0.5 i x_i\right)^4$	Zakharov	30	[−5,10]	0

Table 8 Multimodal benchmark functions

Function	Function name	Dim	Range	Fmin
$f10(x)=\sum_{i=1}^{n} b\left(x_{i+1}-x_i^2\right)^2+(a-x_i)^2; a=1, b=100$	Rosenbrock	30	[−5,10]	0
$f11(x)=\sum_{i=1}^{n} ix_i^4+random\ [0,1]$	Quartic	30	[−1.28,1.28]	0
$f12(x)=10n+\sum_{i=1}^{n}(x_i^2-10cos(2\pi x_i))$	Rastrigin	30	[−5.12,5.12]	0
$f13(x)=1+\sum_{i=1}^{n} sin^2(x_i)-0.1e^{\left(\sum_{i=1}^{n} x_i^2\right)}$	Periodic	30	[−10,10]	0.9
$f14(x)=1-cos(2\pi\sqrt{\sum_{i=1}^{D} x_i^2})+0.1(\sqrt{\sum_{i=1}^{D} x_i^2})$	Salomon	30	[−100,100]	0
$f15(x)=\sum_{i=1}^{n}\sum_{i=1}^{5} j\sin((j+1)x_i+j)$	Shubert	30	[−10,10]	−29.6733337
$f16(x)=\frac{1}{2}\sum_{1}^{n}\left(x_i^4-16x_i^2+5x_i\right)$	Styblinski-Tank	30	[−5,5]	−39.16599
$f17(x)=\sum_{i=1}^{n}\varepsilon_i\lvert x\rvert_i^i$	Xin-She Yang	30	[−5,5]	0
$f18(x)=(\sum_{i=1}^{n}\lvert x_i\rvert)exp(-\sum_{i=1}^{n} sin\left(x_i^2\right))$	Xin-She Yang N. 2	30	[−2π,2π]	0
$f19(x)=(\sum_{i=1}^{n} sin^2(x_i)-e^{\sum_{i=1}^{n} x_i^2})e^{-\sum_{i=1}^{n} sin^2\sqrt{x_i}}$	Xin-She Yang N. 4	30	[−10,10]	0

$$F_{1.2}=-F_{2.1}, Sh_1=Sh_2, SH=Sh/Wh \tag{5}$$

$$S_{total}=SH_1-SH_2 \tag{6}$$

After considering the angle and weight calculations, the fitness of each repetition is calculated. The suitability parameters of the WAO algorithm consist of four parts: FitL, FitLG, Afit, and Fit-G. For optimal local FitL calculation. The weight of each batch of particles must be calculated at each iteration. There may be several local.

optimizations. The most considerable calculated local optimization is considered to be FitLG. The average of local optimizations is also calculated according to Eq. 7. The optimal global response is the Fit-G point, which has gained the most weight in the response space and will establish a stop condition. This calculation is performed using Eq. 8.

$$AFit=\frac{Number\ of\ H}{Number\ of\ FitL} \tag{7}$$

$$FitG=maximum\ (W_h) \tag{8}$$

A reduction Coefficient must be applied in the sample space to eliminate weak points. The latent heat of the water has been used for this purpose. Latent heat is the amount of energy required to evaporate a substance, and heat of water evaporation is the amount of energy that causes water to evaporate at a temperature of 100 °C, which is calculated in joules and follows Eq. 9 [64]. In this relation, Q is equal to the amount of energy received with the release during a phase change (in joules). M is the mass of the material in question (which in the algorithm is equal to Wh). L The specific latent heat of a substance (this value is measured in joules per mole; the specific latent heat of water at 100 °C is 2400 and the melting point of ice at zero degrees is 336). According to the description, we find that water can reduce worthless particles. The proposed algorithm applies the temperature reduction coefficient after each optimization calculation. This is done according to Eq. 10. In this respect, Hf is equal to the bonded particle, Afit is the average of optimality. If the value obtained is a negative number, it will be removed from the sample space; otherwise, it will remain.

Table 9 Fixed dimension multimodal benchmark functions

Function	Dim	Range	Fmin
$f20(x)=\sum_{i=1}^{11}\left[a_i-\frac{x_1\left(b_i^2+b_ix_2\right)}{b_i^2+b_ix_3+x_4}\right]^2$	4	[−5, 5]	0.00030
$f21(x)=\left(x_2-\frac{5.1}{4\pi^2}x_1^2+\frac{5}{\pi}x_1-6\right)^2+10\left(1-\frac{1}{8\pi}\right)cosx_1+10$	2	[−5,5]	−398
$f22(x)=-\sum_{i=1}^{4} c_iexp\left(-\sum_{j=1}^{3} a_{ij}\left(x_j-p_{ij}\right)^2\right)$	3	[0,1]	−3.86
$f23(x)=-\sum_{i=1}^{4} c_iexp\left(-\sum_{j=1}^{6} a_{ij}\left(x_j-p_{ij}\right)^2\right)$	6	[0,1]	−3.32
$f24(x)=-\sum_{i=1}^{5}[(X-a_i)(X-a_i)^r+c_i]^{-1}$	4	[0,10]	−10.1532
$f25(x)=-\sum_{i=1}^{7}[(X-a_i)(X-a_i)^r+c_i]^{-1}$	4	[0,10]	−10.4028
$f26(x)=-\sum_{i=1}^{10}[(X-a_i)(X-a_i)^r+c_i]^{-1}$	4	[0,10]	−10.5363

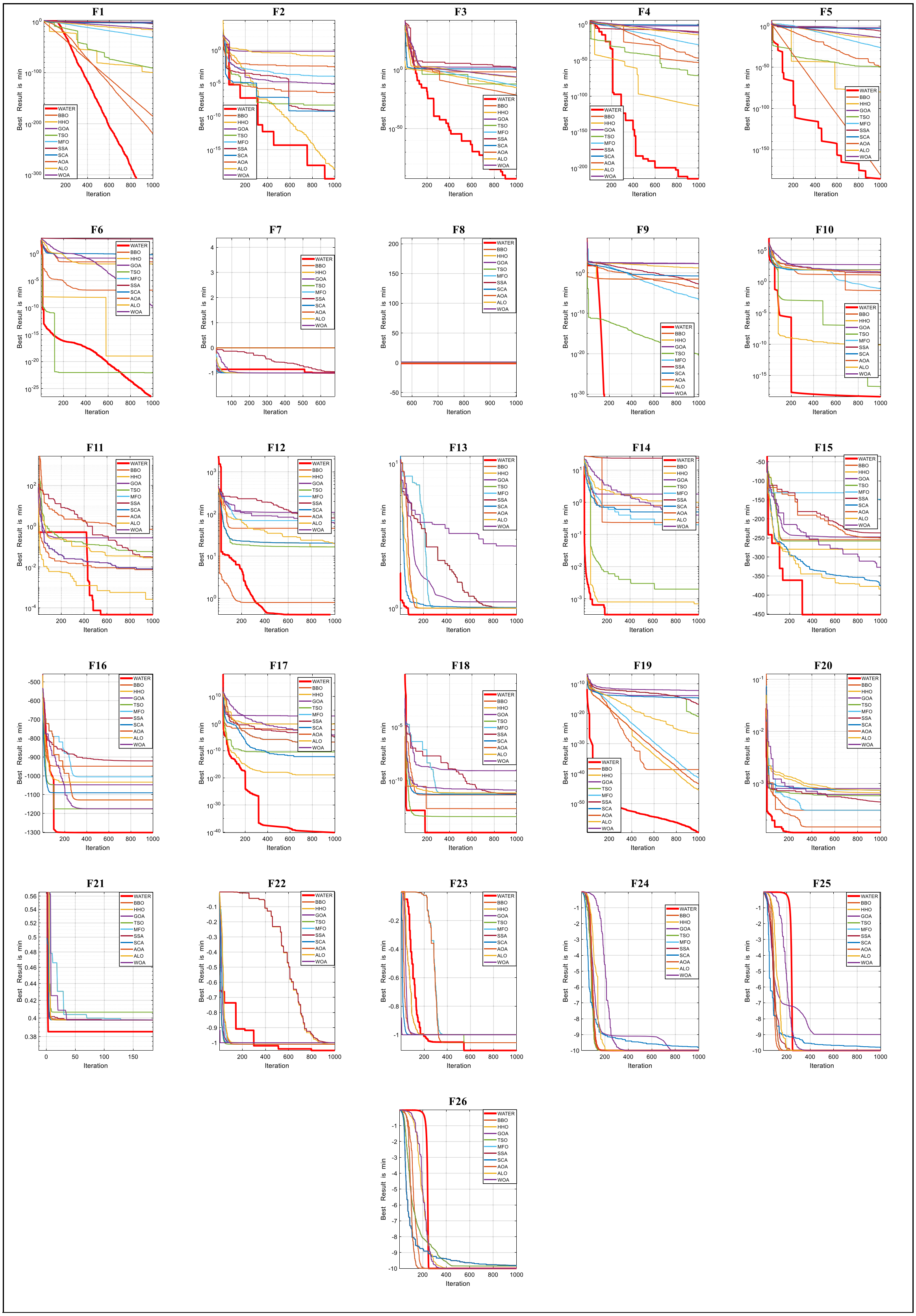


Fig. 7 Graph results of standard functions

$$Q = M.L \tag{9}$$

$$k = Hf - Afit \tag{10}$$

Impact calculation must also be done in this algorithm. Impact includes two parameters of impact angle and impact speed. The values entered in each stroke are selected randomly. The random digits entered are calculated according to the previous velocity and angle relations (Eqs. 5 and 7). After updating the speed and angle, the condition for establishing the welding temperature should be checked. If the condition is met, then the temperature of each remaining optimal local point is considered 100 °C, and the maximum speed in the previous stage recorded is considered the moving speed. To calculate the moving angle, we divide the optimal local particle number by 360 and move the rendered value to the obtained angles at the previous speed. Equation 11 is used To calculate the H_temp angle. We assign the obtained value in different directions of the problems, and each particle starts moving at one of these angles.

$$\theta Htemp = \frac{Nh}{Problem\ directions} \tag{11}$$

3.4 Solving a real problem

In this part of the article, a practical example is solved to understand the proposed algorithm better. To better understand the process of the algorithm, the standard function $f(x) = \sum_{i=1}^{2} x_i^2$ is calculated in three iterations. The dimensions of possible answers are equal to the range (−3, 3). Six initial random answers were used to obtain the minor point of the x_i^2 functions. According to the description of the parameters, the first six particles of the water molecule with the direction of motion and velocity (S, θ) and for each particle, a random location PL is also defined. The initial answers are given in Table 2, which by placing each answer and obtaining the output of the function x_i^2, we represent it with PL in the algorithm. The weight of the particles is also calculated by default according to Eq. 1. The velocity and direction of motion of the particles are also randomly set. Particles and initial values are shown in Table 3, where each particle is represented by H, whose coefficient number indicates the particles and their index.

The amount of displacement of each particle is calculated using the Euclidean distance from the origin of the coordinates, and the amount of its motion is determined by taking the result with the previous location of each particle. By colliding each particle with each other at a point, a bond is created, and with the increase of particles at each point of the device, the coordinates of the optimal points are formed. With the bonding of each particle, the direction of motion also changes, and this change is the result of taking the particles together. According to the previous location of each particle, the amount of displacement obtained from it will be determined, and the particles will be displaced. The amount of displacement is also calculated using the constant velocity displacement formula. According to the steps of the algorithm, in the first step, the particles H1, H2, H6, and the particles H3, H4 are moved together in one place, and bonds are established. Also, the H5 particle is removed from all particles and will not affect optimality; according to the said conditions, this point will be removed from the problem space using the K parameter. To apply K on each particle, first, the parameters Afit, Fitl, FitLG must be calculated, and then concerning the value of Afit, it is determined which particles must be removed from the space. The steps are listed in Table 4.

Calculating parameters:

$$Afit = \frac{number\ of\ H}{number\ of\ bonding\ points} = \frac{6}{2} = 3$$

$$FitLG \geq 3$$

$$FitL < 3$$

$$K_{H5} = Hf - Afit = 0 - 3 = -3$$

Table 10 Average results of unimodal benchmark function

Algorithm Function	WAO	BBO (2012)	HHO (2019)	GOA (2019)	TSO (2020)	MFO (2015)	SSA (2017)	SCA (2016)	AOA (2020)	ALO (2015)	WO (2016)
F1	1.66	0.12	0.46	1.77	0.33	1.84	3.18	0.91	0.23	2.08	4.07
F2	7.87	0.32	0.39	1.88	0.34	1.29	1.23	0.61	0.32	2.42	3.74
F3	1.55e+32	11.28e+34	2.35e+04	1.55e+32	7.06e+34	2.34e+36	8.37e+38	7.06e+34	2.34e+36	2.08e-07	1.95e+38
F4	2.50	11.99	101.49	51.38	57.89	103.90	151.62	43.23	209.20	116.77	209.21
F5	1.24	11.19	9.80e+03	3.24	108.82	65.96	9.80e+03	294.52	0.19	122.20	388.19
F6	0.47	11.99	7.19	3.02	0.68	7.17	578.01	3.63	3.96	7.35	16.18
F7	−0.92	−0.99	0.01	−0.98	−0.00	−0.98	−0.64	−0.99	−0.00	−0.98	−0.95
F8	−0.85	0.99	0.71	9.09e-04	0.01	0.99	0.99	0.99	0.01	0.07	0.99
F9	6.18	11.04	2.86e+03	3.05e+03	2.75e+03	1.01e+04	6.62e+05	105.29	2.81e+03	272.09	4.85e+04

Table 11 Average results of multimodal benchmark functions

Algorithm Function	WAO	BBO	HHO	GOA	TSO	MFO	SSA	SCA	AOA	ALO	WO
F10	9.10e+03	1.52e+03	1.67e+04	4.91e+03	855.42	1.40e+04	1.68e+04	3.11e+03	1.68e+04	1.00e+04	1.86e+04
F11	0.21	1.00	0.07	0.48	2.64	0.73	7.96	0.73	27.97	0.65	0.73
F12	38.52	25.48	27.82	122.32	25.07	88.66	166.22	28.63	4.75	49.91	104.60
F13	0.91	1.46	2.37	1.80	2.24	2.38	2.24	1.20	1.47	1.23	3.65
F14	0.01	0.75	0.30	2.20	0.63	1.09	23.89	0.89	4.18	2.12	2.68
F15	−376.52	−153.31	−263.97	−233.09	−244.58	−136.13	−187.72	−319.78	−197.56	−334.15	−253.11
F16	−1.01e+03	−1.01e+03	−1.05e+03	−1.03e+03	−1.15e+03	−950.91	−892.85	−1.08e+03	−1.05e+03	−1.01e+03	−1.11e+03
F17	1.98e+15	1.85e+08	8.06e+08	2.18e+07	7.52e+08	8.42e+08	3.10e+10	6.70e+08	8.42e+08	7.94e+08	2.70e+09
F18	0.12e-08	1.42e-09	1.19e-06	9.63e-07	9.79e-07	1.19e-06	1.12e-06	9.81e-07	1.39e-09	1.12e-07	5.27e-06
F19	1.82e-15	3.07e-12	3.52e-11	2.82e-11	3.21e-10	4.46e-11	3.21e-10	2.84e-11	4.45e-11	4.37e-11	5.99e-10
F20	9.10e+03	1.52e+03	0.00	4.91e+03	9.15e-04	1.40e+04	1.68e+04	3.11e+03	3.39e-04	1.00e+04	1.86e+04

Continuing the process of the algorithm and obtaining the local optimization, it is observed that the junction point H3, H4 have not changed and have moved away from better answers. In this case, using the hit operator (P) on this link, a global search is created to join a better point. After applying this operator, the two optimally obtained answers are close, but no connection occurs. To give local optimization a second chance, use the HTEMP operator to remove the links of the desired points and force it to search the world again. In this step, by entering HTEMP to the local optimal point H3, H4 separates the link and gives each one a random angle of motion and velocity. The H3 particle then binds to the best local FitLG optimizer, and the H4 particle moves away from the response space. According to the algorithm steps, the optimization values are updated, and the K operator is applied to the value points. At this stage, it is observed that the global optimal point is formed by the bond of 4 particles located in the coordinates (0 and 0.5). Which also weighs 4. At the end of each iteration, the weight of each response is calculated and compared with the previous iteration. Then the fit of the obtained point to the function will be calculated and compared with the previous fit. The steps described in Table 5 are also calculated and given.

Calculating parameters:

$$P : \{S = 2, \theta = 180°\}, P + (H3, H4) = Pl\,(\,0, 0.5\,)$$

$$HTEMP : \{\, make\ H3, H4\ to\ 2\ points\}$$
$$-H3\{s : 1, \theta : 315°\}{-}H4\{s : 1, \theta : 45°\}$$

$$Afit = \frac{number\ of\ H}{number\ of\ bonding\ points} = \frac{6}{1} = 6$$

$$FitLG \geq 6$$

$$FitL < 6$$

$$K_{H4} = Hf - Afit = 0{-}3 = -6$$

$$Wh = 4\ with\ fit\ 1$$

Table 12 Average results of fixed dimension multimodal benchmark functions

Algorithm Function	WAO	BBO	HHO	GOA	TSO	MFO	SSA	SCA	AOA	ALO	WO
F21	1.78e-04	4.79e-04	0.40	0.00	0.41	6.59e-04	7.23e-04	0.00	0.40	0.00	0.00
F22	0.38	0.39	−0.41	0.39	−0.99	0.40	0.39	0.39	−0.97	0.40	0.39
F23	−0.65	−0.99	−0.93	−0.94	−0.74	−0.97	−0.40	−0.98	−0.76	−0.96	−0.99
F24	−2.77	−0.98	−9.10	−0.89	−9.19	−0.72	−0.32	−0.94	−9.10	−0.93	−0.99
F25	−7.62	−9.11	−8.79	−7.54	−9.06	−9.10	−3.60	−8.83	−9.06	−8.80	−7.97
F26	−7.47	−9.11	−8.82	−7.82	−8.55	−9.06	−3.56	−8.92	−9.02	−8.79	−8.06

Table 13 Best results of unimodal benchmark functions

Algorithm Function	WAO	BBO	HHO	GOA	TSO	MFO	SSA	SCA	AOA	ALO	WO
F1	0	4.20e-183	1.64e-18	0.05	7.34e-45	1.08e-32	4.46e-05	0.00	3.24e-181	2.79e-17	1.11e-14
F2	2.82e-24	0.02	8.52e-19	0.64	9.04e-05	1.14e-04	6.96e-10	6.20e-10	0.00	0.12	7.26e-06
F3	3.09e-93	1.16e-9	4.54e-16	100.87	5.57e-13	1.63e-14	7.11	2.78	2.32e-22	6.78e+34	4.95e-07
F4	7.99e-216	6.28e-49	1.70e-11	0.00	3.04e-72	5.30e-29	3.76e-11	0.08	5.78e-54	2.68e-15	1.02e-12
F5	7.79e-186	1.78e-52	1.16e-80	0.00	1.79e-50	6.31e-27	1.02e-06	9.72e-04	1.99e-181	4.46e-15	7.87e-15
F6	0	4.3e-164	1.06e-19	0.14	8.52e-23	0	530.84	0.62	1.74e-07	0.01	1.69e-10
F7	−1	−1	0	−0.99	−0.25	−1.00	−1.00	−1.00	−0.25	−1.00	−1.00
F8	-1	0.99	5.03e−19	0	1.41e-04	0.99	0.99	0.99	0.00	0	0.99
F9	0	3.57e-13	0.02	199.06	4.30e-21	2.15e-07	0.00	0.16	0.02	17.34	205.11

After completing the first iteration, the second iteration begins, and the algorithm tries to get a better answer. The optimal point obtained in the previous iteration remains in place. It contains four answers in 1 place. Two points, H4 and H5, are loaded randomly with the default and random values of their parameters in the problem. As shown in Fig. 5, points H4 and H5 are interconnected and continue to move close to the best response. According to the process of the algorithm, the calculated local optimal point is hit by the operator and causes its connection with the global optimal point. After the result between angles and velocity, it is observed that the best local optimization is equal to the global optimization, and there is no need to apply the HTEMP operator. In this case, the second iteration ends, and in fact, the algorithm achieves the response convergence. After performing the steps of the algorithm, the optimality values are calculated and compared with the previous steps. The suitability of this step is equal to the weight of 6, and at points (0 and 0), the suitability of its function is equal to the seam, and the minimum function is achieved.

3.5 Pseudo-code and flowchart

The steps of the WAO algorithm are as follows.

1. The initial population is loaded randomly with the default speed value and random motion angle.
2. The weight of the optimum points, the new angle of motion, and the new speed are calculated for each point.
3. Local optimization, overall optimization, and average optimization for global search and elimination of weak points are calculated.
4. If the local optimizations are less than the average optimization, then a shock jump occurs, creating new points, new speed, the new angle of motion, and the temperature reduction coefficient is calculated and applied to eliminate weak points.

Table 14 Best results of multimodal benchmark functions

Algorithm Function	WAO	BBO	HHO	GOA	TSO	MFO	SSA	SCA	AOA	ALO	WO
F10	4.00e-19	20.89	7.50e-11	527.33	1.71e-17	0.07	26.50	80.26	0.03	72.22	37.46
F11	4.61e-05	3.60e-02	2.60e-04	0.26	0.05	0.00	0.03	0.00	0.70	0.02	0.00
F12	0	7.55	16.50	109.33	16.50	69.64	80.60	20.27	0.79	19.89	60.12
F13	0.90	1.06	0.99	1.10	0.99	1.00	1.00	1.00	0.99	1.00	2.68
F14	3.20e-04	1.05	6.95e-04	1.79	0.00	0.19	23.49	0.49	0.23	0.99	0.40
F15	−376.84	−245.44	−279.83	−247.53	−258.09	−148.50	−236.81	−374.98	−249.28	−384.46	−327.16
F16	−1.03e+03	−1.03e+03	−1.12e+03	−1.04e+03	−1.17e+03	−1.00e+03	−920.52	−1.09e+03	−1.12e+03	−1.0e+03	−1.17e+03
F17	6.85e-41	2.40e-36	1.29e-19	672.76	3.65e-11	1.82e-07	2.08e-05	5.03e-13	1.62e-07	0.72	6.86e-05
F18	6.19e-13	3.03e-11	7.09e-12	8.75e-10	4.93e-14	7.09e-12	6.34e-12	5.53e-12	2.70e-13	7.74e-12	1.42e-11
F19	2.13e-60	4.54e-32	3.14e-46	1.10e-14	1.09e-21	3.95e-42	1.37e-17	2.05e-15	3.52e-44	2.45e-27	5.59e-13
F20	4.00e-19	20.89	6.62e-04	527.33	5.82e-04	0.07	26.50	80.26	1.46e-04	72.22	37.46

5. If there is no significant change in the impact jump, the temperature jump will enter, and new points will appear.
6. The average optimality (AFit) and overall optimality (FitLG) are checked. If the overall optimality (FitLG) is better than the average optimality (AFit), the stage is completed; otherwise, it returns to stage 3.
7. The above steps are performed until the stop condition is met.

After reviewing all the steps, the pseudo-code is given in Algorithm 1. Then at the end of this section is an overview and flowchart of the algorithm in Fig. 6.

Algorithm 1. WAO Algorithm

```
WAO Algorithm
1. Input: H, Pl, S, θ, P, Htemp;
2. Output: Best Solution ;
3. Procedure WAO
4. Generate n primary molecules of H
5. Set Pl , S=1 , θ=( 0°to 360° )
6. set I for itration counter ;
7.   While ( I  in stopping condition )
8.      Compute Wh for each H bonding (Wh=((∑ 18 ∗ Nh));
9.      Compute New (Pl , S ,  θ ) for each bonding
10.       S_f = s/Wh ;
11.       Sk = Sf_1 − Sf_2 ;
12.       θ(New) = θ1 − θ2 ;
13.       θ(total) = (360° + (−θnew)) ;
14.       update Pl
15.       Compute (FitL , FitLG , AFit ,)
16.       FitL = ∑Wh ;
17.       FitLG=Maximum FitL;
18.       AFit = ∑_{fit1}^{n} FitL/n ;
19.           If ( FitL < FitLG ) do
20.               Importe  P to FitL (Enter P with the values of θP and Sp)
22.               Sp=∑wh  of FitL ;
23.               Snew=Sp – Sk ;
24.               θp = (0° to 360°) ;
25.               θnew =  θp − θ(total)
26.               θ(NEW) = (360° + (−θnew))
27.               update Pl
28.               Importe  k = Hf − Afit ;
29.           End If;
30.           If ( FitL had no change ) do
31.                Importe HTemp to FitL ;
33.                Increase Temp to 100°C  ;
34.                θHtemp = Nh/360 ;
35.               Importe  k = Hf − Afit ;
36.            Else
37.                Comparison the AFit & FitG;
38.            End If ;
39.            If (fitG > Afit)
40.                Then end this iteration ;
41.            Else
42.                Go to line 6 ;
43.            End If ;
44.  End  while;
45. End Procedure;
```

3.6 The time complexity of WAO

The time complexity of the proposed WAO is analyzed as follows. The While loop of WAO takes the most. It repeats iteration counter times. In each iteration, and according to the value of (FitL), we have three if conditions that contain *p, q*, and *m* statements, respectively. So the complexity of the algorithm is *O(iteration counter + max(p,q,m)*, which is equal to *O(n).*

4 Analysis of experiments and results

In this section, the work done to evaluate the WAO algorithm is introduced. This assessment presents two artificial areas of continuity and practical and real continuous issues. In the field of artificial problems, 26 standard functions have been used to evaluate the proposed algorithm, and its results have been presented in comparison with similar algorithms. In applied problems, four applied problems have been used, and each problem has been defined and studied. First, in Section 4.1, continuous functions are introduced, and the results are shown with other continuous algorithms. Section 4.2 introduces practical problems and then compares the algorithms in numbers and shapes for better understanding. These issues include speed reducer, welded beam, robot path planning, and Combinatorial-Optimization-Based Threat Evaluation and Jamming Allocation (cotja). Also, to prove the algorithm's efficiency in the last part of this section, the proposed algorithm, along with four well-known algorithms, has participated in the CEC2020 test. This test includes 57 real optimization problems designed by the CEC Congress to validate Metaheuristic algorithms [65].

Table 6 shows the hyperparameters of similar continuous algorithms; This table uses each algorithm's default values that compare the results with their implementation. In the compared algorithms, the parameters Population size and number of generations are observed in some algorithms. The values of these parameters in all the studied algorithms are given with 50 and 1000, respectively. These values were chosen because the execution conditions are the same. The rest of the values given are valued according to the main algorithms of each article. The given parameters have default values of each algorithm. These values are given due to equal and fair conditions and for comparison of other algorithms with the WAO algorithm.

4.1 Review and analysis of experiments performed on standard functions

The Water metaheuristic algorithm uses 26 standard functions to compare with competing algorithms. The same execution conditions, the values of the repetition of the functions are

Table 15 Best results of fixed dimension multimodal benchmark functions

Algorithm Function	WAO	BBO	HHO	GOA	TSO	MFO	SSA	SCA	AOA	ALO	WO
F21	1.14e-04	3.07e-04	0.39	8.03e-04	0.40	3.07e-04	4.43e-04	7.48e-04	0.39	7.43e-04	6.20e-04
F22	0.38	0.39	−1.00	0.39	−1.01	0.39	0.39	0.39	−1.01	0.39	0.39
F23	−0.65	−1.00	−1.00	−1.00	−1.05	−1.00	−0.99	−1.00	−1.05	−1.00	−1.00
F24	−2.77	−1.00	−10.00	−0.99	−10.00	−1.00	−0.99	−1.00	−10.00	−1.00	−1.00
F25	−10	−10.00	−10.00	−9.97	−10.00	−10.0	−9.99	−9.80	−10.00	−10.00	−10.00
F26	−10	−10.00	−9.82	−8.99	−9.85	−10.00	−9.99	−9.79	−10.00	−10.00	−10.00

considered the same in all problems. To better understand the evaluation, problems were tested in three classes: 1- Unimodal benchmark functions [37], 2- Multimodal benchmark functions [66] and 3- Fixed dimension multimodal benchmark functions [37, 66]. To compare the algorithm, BBO [32], HHO [31], GOA [13], TSO [26], MFO [34], SSA [53], SCA [47],AOA [15], ALO [35], and WO [39] have been used; Table 7 lists the Unimodal functions, which give the optimal values, range, and function name. Multimodal benchmark functions are also listed in Table 8, which also shows average values. Finally, Table 9 also includes the previous information in Fixed dimension multimodal benchmark functions, the given computational information.

The results of the WAO algorithm in responding to standard functions are better than competing algorithms in most functions. The basic parameters and initial solutions

Table 16 Time results of all algorithms

Algorithm Function	WAO	BBO	HHO	GOA	TSO	MFO	SSA	SCA	AOA	ALO	WO
F1	**3.6866**	4.3654	6.022	5.1144	5.121	5.9945	4.0114	4.5118	4.123	5.4555	6.3311
F2	**3.5945**	4.1148	5.101	5.1147	5.092	5.1466	3.9874	3.5554	4.011	5.3126	5.6544
F3	**3.7914**	4.3687	4.992	5.0012	5.032	5.3654	4.1121	4.3664	5.011	5.1154	5.1114
F4	**3.4547**	3.9874	5.899	5.4578	5.366	5.9987	3.9654	3.5654	4.110	5.4487	5.2314
F5	**3.4874**	4.1687	5.336	4.9546	5.066	6.3256	4.3365	4.7745	4.544	6.3346	5.3244
F6	**3.1779**	4.1369	5.112	4.6354	5.664	6.4541	3.6541	3.6571	4.411	6.7541	6.1645
F7	**3.4777**	3.9874	6.121	5.6541	5.855	6.7844	4.1121	4.6645	5.011	6.2114	6.0344
F8	**4.0144**	4.7741	5.411	5.2136	5.011	5.1554	4.3214	4.9975	4.154	5.6654	5.341
F9	**3.7875**	4.1389	5.844	5.9874	5.789	5.9415	4.3365	4.3322	4.314	5.3115	5.2214
F10	**3.1478**	3.8741	5.632	4.5689	5.103	6.3321	3.9654	3.9344	4.003	6.3111	6.6541
F11	**4.1196**	4.6974	5.139	4.9124	5.911	4.9456	4.2254	4.2854	4.977	4.1256	6.1442
F12	**3.4147**	3.8414	5.782	4.8524	5.312	4.9214	4.3114	4.3554	4.744	4.1144	6.3001
F13	**3.9745**	4.2159	5.144	5.4974	5.101	6.4895	4.5689	4.8789	4.441	6.1895	5.8014
F14	**3.1789**	3.5641	5.311	4.9844	5.021	6.3189	3.9114	3.1164	4.005	6.9389	5.6551
F15	**3.4574**	4.1036	5.082	4.8741	5.021	5.3298	3.8974	3.7474	4.413	5.2198	5.2741
F16	**3.5446**	3.8741	6.092	4.5644	5.701	5.1147	3.9774	3.9747	5.113	5.2217	5.2461
F17	**3.7784**	3.7841	6.121	5.1001	5.421	6.1020	3.8874	3.8774	4.033	6.1990	5.9674
F18	**4.0224**	4.3256	5.787	6.1014	5.331	6.2354	4.3332	4.5432	4.145	6.2354	5.0011
F19	**4.1298**	4.8621	6.410	6.3101	5.031	7.0121	4.5687	4.5587	4.978	7.0121	5.1017
F20	**3.4479**	3.5214	5.012	5.0114	4.127	6.1498	4.3558	4.2558	4.443	6.1958	5.1473
F21	**3.8974**	4.3001	4.042	4.1796	4.121	6.3321	4.1124	4.1774	4.143	6.5121	6.1741
F22	**3.1799**	3.9478	5.422	4.6932	5.114	5.8521	3.9114	3.7714	4.044	5.4621	6.4424
F23	**3.6541**	4.1877	5.141	4.3124	5.021	6.2305	3.8224	3.6924	4.023	6.6805	5.1477
F24	**3.4799**	4.3121	4.987	5.0210	4.157	6.1140	3.8874	3.8774	4.112	6.1220	5.7241
F25	**3.7894**	4.2379	5.101	5.3598	5.443	6.4474	4.2162	4.5762	4.011	6.1474	5.5314
F26	**3.4547**	3.9874	5.877	4.9593	5.123	6.3219	4.3354	4.2554	5.003	6.3229	5.101

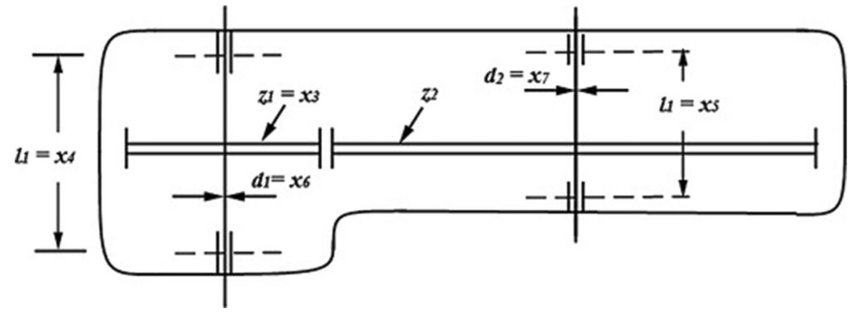


Fig. 8 Schematic view of speed reducer problem [25]

of the functions are the same to make a fair comparison, and the dimensions and amplitude in each function are the same for each computed algorithm. According to Tables 7, 8, and 9, F1 to F9 are Unimodal benchmark functions, F10 to F19 are Multimodal benchmark functions, and F20 to F26 is Fixed dimension multimodal benchmark functions. Figure 7 shows all available answers of 26 standard functions in a graph. According to these diagrams, the strength of the proposed algorithm compared to competing algorithms can be recognized. According to the diagrams, it can be said that the proposed algorithm has a complete advantage in responding to the functions F1, F2, F3, F4, F5, F6, F9, F11, F12, F14, F17, F18, F19, F20, and with more repetition. Therefore, it has a better response than competing algorithms. Although the proposed algorithm is superior in the said functions, according to the numerical results of the response of the algorithms, it can be seen that the performance of the proposed algorithm is entirely superior.

The average answers of each algorithm are given in Tables 10, 11, and 12 to evaluate the WAO algorithm in the field of standard functions. According to the previous classification, Table 10 is the average response of Unimodal benchmark functions, Table 13 is the average response of Multimodal benchmark functions, and Table 12 is the average response of Fixed dimension multimodal benchmark functions. Each algorithm shows the closest answer to the final answer about their general answers in 1000 repetitions in these tables. The proposed algorithm results are competitive compared to the competing algorithms, and it has obtained better answers in some functions. According to the previous classification, Table 10 is the average response of Unimodal benchmark functions; Table 11 is the average response of Multimodal benchmark functions, and Table 12 is the average response of Fixed dimension multimodal benchmark functions. Each algorithm shows the closest answer to the final answer concerning their general answers in 1000 repetitions in these tables. The proposed algorithm results are competitive compared to the competing algorithms, and it has obtained a better response in some functions.

In this section, The best answer obtained in each algorithm is discussed. Moreover, to show the capability of the proposed algorithm in the field of standard functions. According to the previous order in Tables 13, 14, and 15, the best answers of each algorithm are given in Unimodal, Multimodal, and Fixed dimension multimodal functions, respectively. According to the results obtained in the said tables, it can be said that the proposed algorithm is competitive with the compared

Table 17 Speed reducer problem results

Algorithm	Worst	Best	Mean	Std.Dev.	Time (Sec)
WAO	1.3270e+04	1.4022e+03	1.6046e+03	880.8960	3.9861
BBO	1.8138e+04	3.1356e+03	3.3081e+03	801.3358	4.9564
HHO	2.0742+04	2.9781e+03	3.0073e+03	1.6282e+03	5.921
GOA	1.2181e+04	2.6225e+03	2.7890e+03	967.5614	5.0112
TSO	2.1637e+04	2.5581e+04	1.5291e+03	3.3067e+03	5.181
MFO	2.1419e+04	2.3509e+03	2.5594e+03	884.9666	6.0123
SSA	2.5437e+04	2.5533e+03	2.9630e+03	1.0682e+03	4.1564
SCA	3.6867e+03	2.3672e+03	2.6440e+03	225.7505	4.1314
AOA	4.6030e+03	2.7031e+03	4.2413e+03	2.9732e+04	4.741
ALO	9.8795e+03	2.2513e+03	2.3266e+03	639.8625	5.4547
WO	1.6263e+04	1.6098e+03	1.8155e+03	766.9684	6.1144

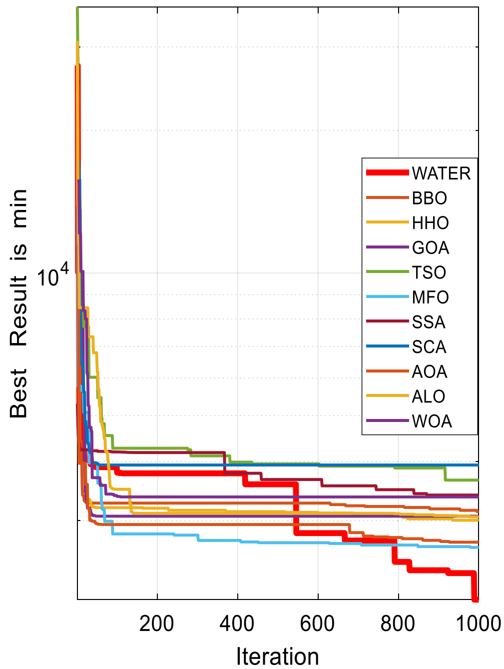


Fig. 9 Graph results of speed reducer problem

algorithms and, in some functions, has achieved a better response than the other algorithms.

To prove the superiority of the proposed algorithm, when performing the performance test, the execution time of each algorithm in 1000 repetitions is also recorded. The system used with the Core i5 processor and 8 GB of RAM has recorded this time in the same execution conditions. The results are shown in Table 16. As the time calculations show, the proposed algorithm has the lowest execution time among competing algorithms.

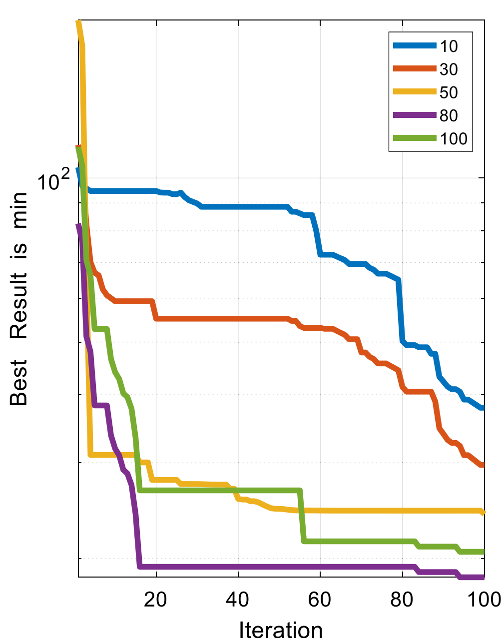


Fig. 10 The difference of population of speed reducer problem results

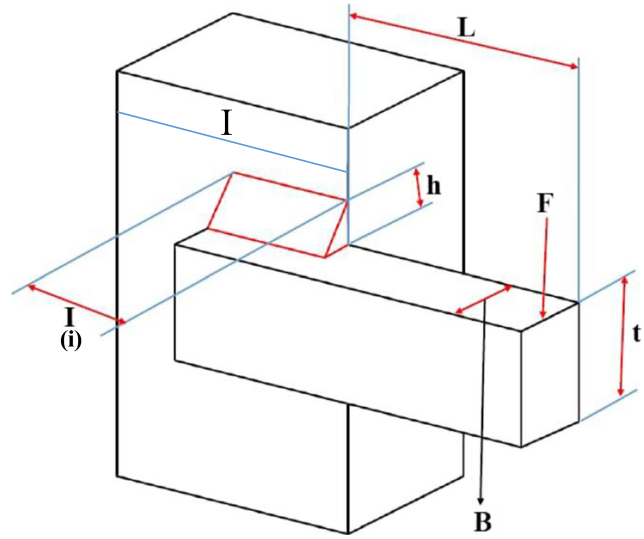


Fig. 11 Schematic view of Welded beam design

4.2 Review and analysis of experiments performed on real functions

In this section, the WAO algorithm is examined and tested on four practical issues. These issues include the speed reducer problem [67], welded beam problem [68], robot path planning [69], and Combinatorial-Optimization-Based Threat Evaluation and Jamming Allocation (COTJA) [70]. To evaluate the performance of the proposed algorithm, the test results of this section are presented in numerical form. First, the problem statement is explained for each problem, and then the results of the WAO algorithm and competing algorithms are evaluated. Also, to prove the capability of the proposed algorithm, experiments that depend on the initial population of each algorithm are presented. In such experiments, we have determined what the WAO algorithm has shown to different populations. Finally, to complete the evaluation and prove the applicability of the proposed algorithm, using the CEC 2020

Table 18 Welded beam design results

Algorithm	Worst	Best	Mean	Std.Dev.	Time (Sec)
WAO	14.7495	**1.1764**	1.4546	0.7483	**3.7541**
BBO	8.2782	1.4918	1.5212	0.3644	5.4564
HHO	14.9829	2.0110	1.9384	0.9815	5.584
GOA	14.1028	1.5163	1.6039	0.7677	5.4412
TSO	15.3078	16.9154	0.8726	2.0504	5.474
MFO	8.0565	1.5243	1.8215	0.6473	6.1244
SSA	21.3410	2.4488	2.6011	0.8726	4.6977
SCA	17.3706	1.7489	1.9426	0.8183	4.0156
AOA	11.3730	1.9475	2.6011	18.0292	5.041
ALO	10.7459	2.3946	2.5118	0.6987	5.1977
WO	3.1024	2.4488	2.4525	0.0271	6.1011

test, the results are compared with the best algorithms presented in this congress [65].

4.2.1 Speed reducer problem

In the case of the speed reducer problem, the goal is to minimize the speed reducer optimally. As shown in Fig. 8, there are several optimization parameters in this problem; in this case, Z represents the number of teeth in the pin. L is the length of the first shaft between the bearings, and L2 is the length of the second shaft between the bearings. d1 is the diameter of the first shaft, and d2 is the diameter of the second shaft. In this issue, the two parameters of optimizing the face width and module of teeth are also variables. In any case, the problem of these variables can exist under different names. In general, this problem is introduced as a real problem with seven optimization variables. This is suitable for continuous algorithms.

As shown in Table 17, the compared algorithms are compared in 5 columns. The first column, called worst, represents the first answer obtained by the algorithm. The second column, called best, is equal to the optimal and final response of each algorithm. The third column is called mean, which represents the mean response. The fourth column is std—dev, which shows the standard deviation of the algorithms. The fifth column is time, which is the execution time of the algorithms. In solving the speed reducer problem, the proposed algorithm has complete superiority over the compared algorithms. The WAO algorithm has obtained the highest values in the top response and the mean responses.

To better understand the performance of the proposed algorithm on the Speed reducer Problem, the performance of competing algorithms and the WAO algorithm shows in Fig. 9. The WAO algorithm is the first algorithm to achieve the optimal response among the proposed algorithms. The WAO algorithm achieves the optimal response from 800 iterations onwards. A graph is prepared and shown in Fig. 10. In this chart, different populations of 10, 30, 50, 80, and 100 have been tested. Also, the number of repetitions in its maximum value is 1000. Algorithm A first optimized with a population of 10 and achieved convergence in its response. With the addition of the population, the response has gradually improved and is very close to the global optimum.

4.2.2 Welded beam design problem

The real continuous problem of Welded beam design is shown in Fig. 11. It is a representation of the welded beam on the desired substrate. In this figure, the beam F is welded to the substrate with a distance of L with welds of thickness h and a length of (i). The desired beam is a rectangular section with length t and width B. The beam material, in this case, is steel. This problem aims to obtain the cost of beam welding and deflection of the end of the beam under load pressure F [51, 68]. The results obtained from this issue are discussed below.

According to the previous problem in Table 18, the comparable algorithms in the five columns have been compared with each other before. The proposed algorithm in solving the problem of Welded beam design has achieved the best optimal answer. The WAO algorithm also has the highest average performance. The WAO algorithm in the mean column has also reached the best answer, which shows the superiority of this algorithm over other algorithms. The WAO algorithm has the best answer. The worst performance also belongs to SSA and WAO algorithms. Also, for a better understanding of the

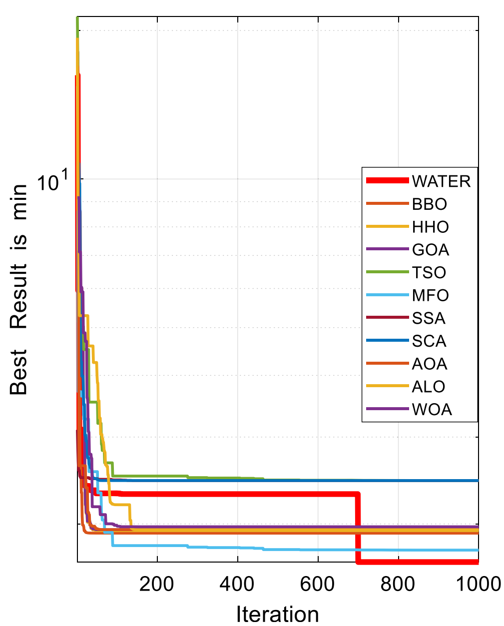


Fig. 12 Graph results of Welded beam design problem

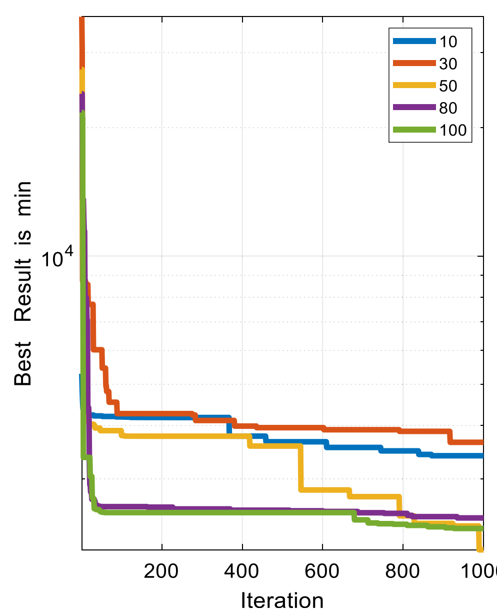


Fig. 13 The difference of population of welded beam design results

performance of each algorithm, Fig. 12 shows a graphical representation of the answers. Also, according to studies on the number of population in this issue, the WAO algorithm has performed well and has reached full optimality even with a population of 50. Figure 13 shows the results of population change.

4.2.3 Robot path planning problem

The performance of existing algorithms deals with the problem of robot routing. In this case, the robot starts moving from a specific direction and moves towards the target, including various obstacles and paths. The goal is to find the easiest route, the closest route. In this case, the obstacles and the target point are identified randomly. This experiment is performed with an initial population of 50 and a repetition rate of 100. Table 19 shows the numerical results of this experiment. The WAO algorithm has had the best performance in obtaining the best response and the average response. Also, the average time of obtaining these calculations is 4.83 s, and the WAO algorithm has achieved the optimal answer 1.42 s earlier. After the WAO algorithm, the GOA algorithm has achieved the second optimal response. The TSO algorithm also has the worst response of all the algorithms.

To better understand the performance of each algorithm, Fig. 14 shows the performance results in this diagram. The WAO algorithm has achieved the best response and convergence after 60 iterations. Also, the WAO algorithm has achieved results in iterations below 10, which most algorithms have achieved from iterations 20 onwards. This performance of the proposed algorithm makes the calculations faster. Therefore, the proposed algorithm has the best performance among the compared algorithms. Also, in Fig. 15, the performance of the WAO algorithm with different populations is investigated, and it has a good performance.

Table 19 Robot path planning results

Algorithm	Worst	Best	Mean	Std.Dev.	Time(Sec)
WAO	194.9845	**24.1539**	29.8039	22.8057	**3.4114**
BBO	104.6929	37.8750	76.8314	19.4883	5.1554
HHO	**118.3172**	38.5503	30.3835	13.0157	5.122
GOA	117.7863	24.9787	31.9381	16.4096	4.9844
TSO	142.4730	141.4545	19.6985	53.1741	4.321
MFO	160.3998	29.4894	37.5654	20.5887	6.1944
SSA	144.6542	26.5946	33.8778	18.5676	4.1244
SCA	164.4003	28.2950	36.6915	21.4057	4.2344
AOA	101.7501	31.6901	33.7651	117.5317	3.977
ALO	171.1605	29.4585	38.2002	22.2859	5.3147
WO	149.4548	25.7227	33.3559	19.4597	5.8977

4.2.4 Combinatorial-optimization-based threat evaluation and jamming allocation (COTJA)

In this issue, advanced radar network systems have been studied. So that in advanced radar network systems, some measures must be taken in the face of threats. The goal is to allocate resources and optimize resources under the management of an aircraft detection radar. The allocation of distributed resources in the environment is an NP-Harddiscrete problem. The goal is to allocate these resources at the lowest possible cost. The results are given in Table 20 numerical calculations To investigate this issue. According to the studies performed, it can be said that the proposed algorithm has the average response and the average superiority time in the best answer obtained. The WAO algorithm performed better than the compared algorithms with a time difference of less than 1 s and a considerable difference due to the best response.

To determine the performance of each algorithm, Fig. 16 shows the results of this problem in the form of diagrams. The proposed algorithm achieves the optimal answer after 450 iterations. None of the algorithms compared in the 500 replications performed is optimal and has a more flawed result than the WAO algorithm. After the WAO algorithm, the WO algorithm has reached a superior result. Figure 17 also shows the demographic of the WAO algorithm. The WAO algorithm with a population of 50 has achieved the optimal response and has gradually improved with the increasing population of responses.

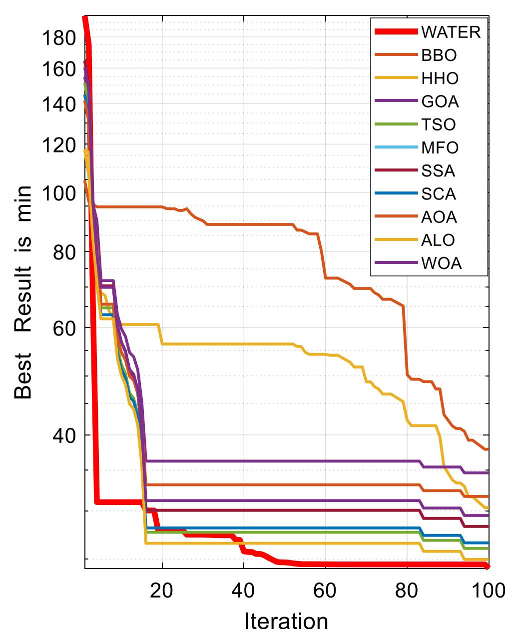


Fig. 14 Graph results of robot path planning

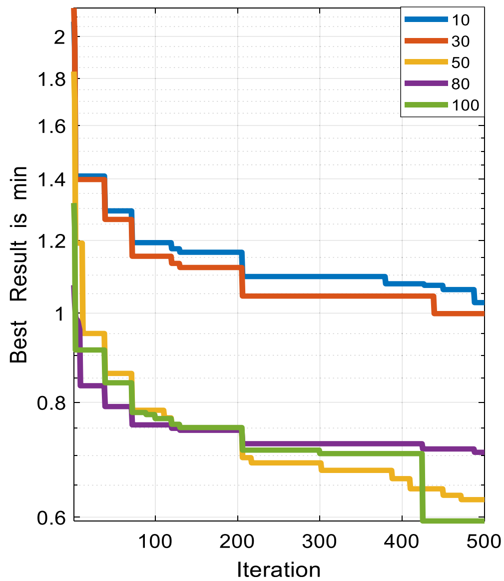


Fig. 15 The difference of population of robot path planning results

4.2.5 CEC 2020 test evaluation results

This section examines the WAO algorithm to answer the CEC2020 test. This test includes 57 real-world optimization problems that Metaheuristic algorithms can answer. Metaheuristic algorithms can evaluate their performance to improve their efficiency using this test [65]. In this section, a complete and real set of such problems is solved using the proposed algorithm and compared with the top 6 algorithms in this field. The algorithms compared in this section include COLSHADE algorithms for Real-World Single-Objective Constrained Optimization Problems [48], A Self-Adaptive Spherical Search Algorithm for Real-World Constrained Optimization Problems (SASS) [49], A Modified Covariance Matrix Adaptation Evolution Strategy for Real-World Constrained Optimization Problems [28], An improved unified differential evolution algorithm for constrained optimization problems (IUDE) [21]. Another algorithm that is one of the best algorithms in the CEC 2107 test is examined in this article. The Improved Multi-operator Differential Evolution Algorithm for Solving Unconstrained Problems (IMODE), which had the best results in the CEC 2017 test, is a combination of several meta-heuristic algorithms [71]. Also, the Algorithm Improving the local search capability of Effective Butterfly Optimizer using Covariance Matrix Adapted Retreat phase (EBOwithCMAR), which is another superior algorithm in the CEC 2017 test, is included in this evaluation. This algorithm consists of several algorithms, the most famous of which is the butterfly algorithm [72].

According to research, these six algorithms have obtained the best results in this test. The extracted results are set and implemented according to the article A test-suite of non-convex constrained optimization problems from the real world and some baseline results in the reference [65]. Table 27 shows Industrial Chemical Processes problems results (RC01 – RC07), Table 28 shows Process Synthesis and Design Problems results (RC08 – RC14), Table 29 shows Mechanical Engineering Problems results (RC15 – RC33), Table 30 shows Power System Problems results (RC34 – RC044), Table 31 shows Power Electronic Problems results (RC45 – RC50), Table 32 shows Livestock Feed Ration Optimization Problems results (RC51 – RC57), which can be seen in the Appendix. In these tables, 57 issues of each section are separated in order. The results in this table include the responses of each algorithm, with Best being the best response of the algorithms, Median the average response of each algorithm, Mean is the mean response of each algorithm, Worst is the worst performance of the algorithm, and SD is the

Table 20 Combinatorial-optimization-based threat evaluation and jamming allocation results

Algorithm	Worst	Best	Mean	Std.Dev.	Time(Sec)
WAO	1.8315	**0.6269**	0.7335	0.1222	**3.8945**
BBO	1.6677	0.7522	0.9233	0.1113	4.8764
HHO	3.0782	0.7430	0.7003	0.0861	6.122
GOA	3.0762	1.3620	1.6110	0.2338	4.8767
TSO	1.5371	1.0727	0.1177	0.7758	5.031
MFO	1.3962	0.6494	0.7193	0.0799	5.3611
SSA	1.8218	0.8217	1.0086	0.1216	4.1978
SCA	2.0274	0.9145	1.1224	0.1353	4.2397
AOA	1.8734	0.7058	0.9383	1.5059	4.113
ALO	1.8629	0.8083	0.8683	0.1089	5.1243
WO	1.9704	0.8550	0.9184	0.1152	5.5647

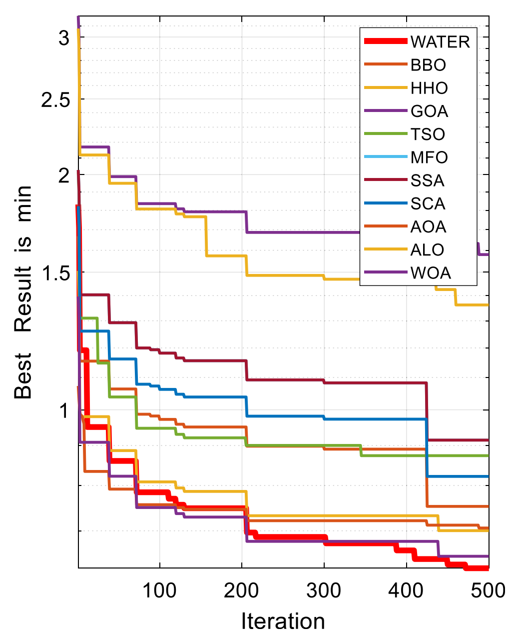


Fig. 16 Graph results of COTJA

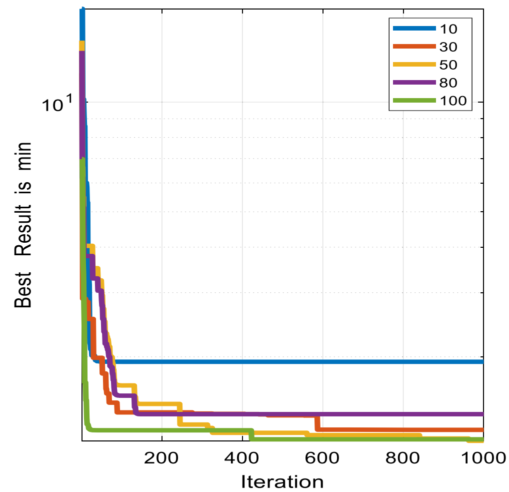


Fig. 17 The difference in the population of COTJA

standard deviation of the algorithms. One measure of the superiority of one method is that it has a better Best value compared to other methods. The proposed algorithm excelled in most issues, and its Best value was better than other algorithms. The proposed algorithm achieved the best answer in 40 tests out of 57 tests and in the other tests had values close to other algorithms. These results indicate the superiority of the WAO algorithm in most evaluation scenarios.

After implementing the CEC 2020 test for the proposed algorithm, Wilcoxon statistical analysis was performed. Table 21 shows the results of the Wilcoxon statistical analysis. In this table, the column values R + and R- represent the value of the signed-rank sum of the WAO algorithm and the comparable algorithms. The P value column shows the difference between the results of each algorithm and the problems solved in the CEC 2020 test because the WAO algorithm has been compared with nine Metaheuristic algorithms that did not use the CEC2020 test, statistical analysis of the results obtained from those algorithms has also been added in Table 21. In the continuation of this test, the algorithm in different dimensions of 10, 15, 20, and 50 is examined. According to the results obtained from this evaluation, the WAO algorithm has had better results in most dimensions. Due to a large number of results, the table of results is placed in the Appendix, which includes Tables 33, 34, 35, 36. In these results, the values of the best result (best), standard deviation (std) and median response (mean) are also included. In addition, according to the CEC2020 test instructions to evaluate the performance measure of each algorithm are repeated 25 times independently. The algorithms are compared in different dimensions, and finally, the value of the performance measure is determined. The values obtained from the performance measurement results are shown in Table 22.

The proposed algorithm is a new algorithm with unique features that had acceptable results in evaluations. The WAO algorithm is suitable for solving continuous optimization problems and is designed for such problems. However, one limitation of the proposed method is that it is limited in discrete problems and may not have optimal answers in some problems. For example, in the CEC 2020 test, the WAO algorithm did not achieve superior results in several problems with a discrete nature, so the proposed algorithm must be updated for discrete problems to be used for both discrete and continuous domains.

Table 21 Results of wilcoxon signed-rank test

WAO vs	Best			Mean			Median		
	R+	R-	p value	R+	R-	p value	R+	R-	p value
IUDE	321	135	0.0626	418	312	0.4231	313	135	0.0641
SASS	325.5	106	0.0096	325	82	0.0698	326	99	0.0078
ϵMAgES	215	165	0.5643	322	121	0.0541	342	112	0.0098
COLSHADE	315	120	0.0618	412	90	0.0021	356	97	0.0081
IMODE	311	122	0.0626	356	112	0.0598	368	111	0.0598
EBOwithCMAR	308	124	0.0649	362	129	0.0578	352	115	0.0647
BBO	313	121	0.0791	365	156	0.0534	321	127	0.0789
HHO	306	98	0.0123	303	109	0.0978	298	111	0.0197
GOA	256	123	0.3654	307	126	0.0687	292	136	0.0269
TSO	286	103	0.0569	369	156	0.0234	332	156	0.0569
MFO	356	156	0.1244	323	108	0.0878	302	148	0.0978
SSA	298	147	0.2784	312	113	0.0787	268	102	0.1650
SCA	306	126	0.2403	295	98	0.0523	288	131	0.2104
AOA	326	98	0.0978	315	103	0.0487	316	126	0.0063
ALO	326	156	0.1265	288	102	0.0689	310	116	0.0089

Table 22 Performance measure

Algorithm	Performance measure
WAO	**0.2202**
IUDE	0.4025
SASS	0.2923
ϵMAgES	0.3109
iLSHADEϵ	0.3086
IMODE	0.2899
EBOwithCMAR	0.2654
BBO	0.4297
HHO	0.4096
GOA	0.3945
TSO	0.3614
MFO	0.3321
SSA	0.3615
SCA	0.3121
AOA	0.3256
ALO	0.3012
WO	0.2622

4.2.6 Evaluation based on Taguchi parameter setting

In the final part of the implementation of the WAO algorithm, to prove the performance criteria of the proposed algorithm, the algorithm is implemented using the Taguchi parameter adjustment method [73]. Parameters automatically selected by the Taguchi method are selected in 25 different modes that are output by this method [74]. The WAO algorithm has achieved good results using this method. In this algorithm, among all the parameters, there are five parameters that the Taguchi method adjusts in 5 different steps with different values and are sent to the evaluation stage of the algorithm. The parameters of each algorithm are different, so the algorithm may have combinations with smaller dimensions. Table 37 shows these values, which are in the Appendix section. The parameters are also given for ten other algorithms, and their different steps and combinations can be seen. The value numbers of initial particles and the dimensions of the problems are selected using the same in the Taguchi method.

After taking the outputs of the Taguchi method to evaluate the WAO algorithm, four real problems were solved using the available figures, including Speed reducer problem, Welded beam, robot path planning, and COTJA were tested and evaluated. The results of these tests include 25 different modes of Taguchi parameter adjustment on each problem. The values Worst, Best, mean, Std, and Time for each test are prepared as output for each problem. The values obtained did not differ much compared to setting the same parameters in the test of each problem and are very close to the previous results. However, the new results are entirely improved and better than the experiments in Section 4. These results are presented in Tables 23, 24, 25, and 26 for the Speed reducer problem, Welded beam design, robot path planning, and COTJA, respectively. The columns of all tables contain the same values as the previous tables, and the Best column shows the best response in the current iterations. The results obtained using the Taguchi parameter adjustment method show the proper performance of the proposed algorithm.

5 Discussion

The WAO algorithm is inspired by the physical and chemical properties of water molecules. Properties such as P and HTemp are influential in forming and changing the state of water atoms. This algorithm has good acceleration due to its special and unique feature while its effective performance in reaching the optimal point. The P operator causes the initial responses to collide with better results and, unlike most similar methods, converges with a smaller population to the

Table 23 Speed reducer problrm evaluation with taguchi method

Algorithm	Worst	Best	Mean	Std.Dev.	Time (Sec)
WAO	**1.1104e+04**	**1.4002e+03**	1.6001e+03	879.0154	**3.9322**
BBO	1.8038e+04	3.1256e+03	3.2751e+03	801.1048	4.9294
HHO	2.0542+04	2.8781e+03	3.0020e+03	1.6056e+03	5.851
GOA	1.1581e+04	2.5269e+03	2.7810e+03	967.4523	5.0012
TSO	2.1567e+04	2.5485e+04	**1.5211e+03**	3.3055e+03	5.164
MFO	2.1312e+04	2.3418e+03	2.5509e+03	884.9528	6.0013
SSA	2.5233e+04	2.5482e+03	2.9601e+03	**1.0637e+03**	4.1492
SCA	3.5867e+03	2.2982e+03	2.6429e+03	225.7482	4.1104
AOA	4.6000e+03	2.6831e+03	4.2402e+03	2.9623e+04	4.701
ALO	9.8795e+03	2.2493e+03	2.3248e+03	639.7309	5.4243
WO	1.6119e+04	1.5093e+03	1.8136e+03	765.0084	6.0104

Table 24 Wlded beam problrm evaluation with taguchi method

Algorithm	Worst	Best	Mean	Std.Dev.	Time (Sec)
WAO	12.0212	**1.0111**	**1.2554**	0.9554	**3.1070**
BBO	8.2000	1.3095	1.3016	**0.3403**	5.3514
HHO	13.0005	2.0110	1.8267	0.9631	5.531
GOA	13.0008	1.4079	1.5104	0.7307	5.2402
TSO	14.3002	16.8163	0.6705	2.0124	5.349
MFO	8.0008	1.4009	1.7904	0.5403	6.1004
SSA	20.3109	2.3249	2.5009	0.6799	4.52707
SCA	16.0509	1.6573	1.8397	0.6982	4.0036
AOA	10.2709	1.8109	2.4602	18.0102	5.023
ALO	10.2004	2.2005	2.4034	0.5628	5.0958
WO	**3.0009**	2.1007	2.3276	0.0209	6.0103

global optimum. Also, the HTemp parameter can increase the execution speed of the algorithm when the answers are improving. These two unique features distinguish the proposed method from other existing methods. The results of evaluations on unimodal and multimodal benchmarks show the superiority of the proposed method compared to ten similar new and classic methods. In these issues, the effect of the proposed method for different populations and replications were carefully studied and evaluated, and due to the new features of the proposed method, namely the effect of P and HTemp, good results were obtained. Also, the classic and essential quarterly in various engineering fields were used as applied scenarios, in which the superiority of the proposed method is shown. Various evaluations for multiple parameters show that the proposed method as an effective method can be used in optimization problems and produce effective results. Various evaluations for multiple parameters show that the proposed method can be used effectively in optimization issues and produce effective results.

Standard Taguchi methods for setting optimal parameters on the proposed method and comparable algorithms have been compared for further and complementary validation of the proposed method. The CEC 2020 Test is also used to comprehensively evaluate the proposed method, in addition to the existing evaluations, and the results are presented in full detail in the Appendix section. The experiments and validation methods have shown that the proposed method has a suitable and acceptable performance in both areas of algorithm speed and obtaining better results.

The proposed method is optimized to solve continuous problems. However, as future work in this field, we intend to update an optimized version of the proposed algorithm to solve more discrete problems. Also, increase its ability to solve multi-objective problems.

6 Conclusion

Metaheuristic algorithms are used to solve optimization problems as well as NP-Hard problems. Where there is no definitive answer to solve problems, and optimization methods are used to solve them. The goal of these algorithms is to obtain an optimal or near-optimal response in good time. For this reason, algorithms have been introduced, which have been suitable for solving a specific category of problems. However, due to the wide variety of optimization issues, researchers are still looking for new ways to overcome the limitations of previous methods. For this reason, the WAO algorithm was introduced and formed inspired by the molecular bonds of water and its properties. The proposed algorithm consists of particles of water molecules that try to find the optimal point by establishing hydrogen bonding under the physical and chemical properties of the water molecule.

Table 25 Robot path planning problrm evaluation with taguchi method

Algorithm	Worst	Best	Mean	Std.Dev.	Time(Sec)
WAO	183.0045	**24.1148**	**27.0032**	**18.3009**	**3.3234**
BBO	100.5203	37.4693	75.8008	19.0008	5.1502
HHO	**110.4743**	38.3309	29.0703	19.0005	5.022
GOA	113.5639	24.2236	30.7824	18.0028	4.8903
TSO	138.2869	141.3006	27.0921	53.0073	4.208
MFO	155.4899	29.1148	35.7839	20.0009	6.0931
SSA	144.6542	26.2793	31.8999	18.0043	4.1019
SCA	158.0907	28.1803	35.7983	21.0072	4.1109
AOA	**97.5237**	31.3571	31.8904	117.5003	3.103
ALO	167.0089	29.2143	36.9972	22.1413	5.2539
WO	137.56887	25.3114	31.0099	19.0003	5.6304

Table 26 COTJA problrm evaluation with taguchi method

Algorithm	Worst	Best	Mean	Std.Dev.	Time(Sec)
WAO	1.6124	**0.6139**	0.7233	0.1101	**3.1036**
BBO	1.6243	0.6979	0.9111	0.1103	4.1597
HHO	3.0209	0.6879	0.6009	**0.0089**	6.009
GOA	3.0347	1.3137	1.6009	0.1215	4.4327
TSO	1.4168	1.0138	**0.0186**	0.5784	4.987
MFO	**1.2943**	0.5387	0.7079	0.0398	5.1182
SSA	1.6103	0.7087	1.0009	0.0205	4.1005
SCA	2.0094	0.8067	1.1112	0.0317	4.2003
AOA	**1.5572**	0.7001	0.9186	1.3008	4.007
ALO	1.5078	0.7099	0.8529	0.0109	5.1009
WO	1.8201	0.7732	0.9079	0.1009	5.4934

The proposed algorithm works in both global and local search domains. The two most essential operators of this method are P and HTemp, which create unique properties. Using the P operator causes the current responses to change position by changing their position due to the force applied to them, and the responses can collide to achieve better responses and connect them. This feature causes the initial population to be reduced due to the impact of P and particle bonding, resulting in more appropriate responses that affect the algorithm's faster convergence. The HTemp actuator also accelerates, achieving a better response and increasing or decreasing its intensity when necessary. The results of evaluations on real problems, standard benchmarks, and various validation techniques show that the proposed method has a significant advantage over several essential and new algorithms in continuous problems. As future work in this field, we intend to update better an optimized version of the proposed method to solve discrete problems. Also, increase its ability to solve multi-objective problems.

Appendix: Validation tests

Table 27 Results of industrial chemical processes problems (RC01 -RC07)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
RC01	WAO	**1.87E+02**	1.89E+02	1.89E+02	1.92E+02	2.17E-14	100	0.00E+00	33
	IUDE	1.91E+02	3.05E+02	2.85E+02	2.00E+00	1.99E+02	0	1.48E+05	0
	SASS	1.89E+02	3.08E+02	2.84E+02	2.06E+00	1.95E+02	2	1.45E+05	5
	ϵMAgES	1.89E+02	1.90E+02	1.90E+02	1.92E+02	7.96E-01	100	0.00E+00	16
	COLSHADE	1.89E+02	2.07E+02	1.97E+02	2.03E+02	6.58E+00	32	4.46E-04	4
	IMODE	1.46E+02	2.25E+02	1.91E+02	2.23E+02	7.24E+00	51	1.23E-04	12
	EBOwithCMAR	1.06E+02	3.12E+02	1.90E+02	2.17E+02	6.09E+00	12	9.02E-04	15
RC02	WAO	7.03E+03	7.04E+03	7.04E+03	7.05E+03	0.00E+00	100	0.00E+00	100
	IUDE	7.05E+03	7.05E+03	6.93E+03	5.94E+03	3.70E+02	92	9.99E+03	92
	SASS	7.04E+03	7.05E+03	6.98E+03	5.95E+03	3.72E+02	90	9.92E+03	88
	ϵMAgES	7.05E+03	7.05E+03	1.05E+04	2.66E+04	6.54E+03	68	8.29E+01	68
	COLSHADE	7.05E+03	7.05E+03	7.05E+03	7.05E+03	5.57E-13	100	0.04E+00	100
	IMODE	7.19E+03	7.06E+03	1.26E+04	2.26E+04	4.27E+03	94	8.29E+01	59
	EBOwithCMAR	7.06E+03	7.05E+03	5.97E+03	6.75E+03	1.25E-17	88	0.07E+00	84
RC03	WAO	−4.51E+03	−4.51E+03	−4.35E+03	−1.42E+02	8.77E+02	100	0.00E+00	35
	IUDE	−4.53E+03	−1.43E+02	−6.08E+03	−1.57E+04	5.88E+03	68	3.86E+00	12
	SASS	−4.55E+03	−1.43E+02	−6.09E+03	−1.52E+04	5.88E+03	62	0.00E+00	22
	ϵMAgES	−1.43E+02	7.63E+01	3.21E+01 -	2.49E+02	1.43E+02	100	0.00E+00	0
	COLSHADE	**−4.59E+03**	−1.43E+02	9.57E+02	4.95E+02	2.04E+03	100	0.00E+00	36
	IMODE	−1.49E+02	5.22E+01	2.14E+01	1.41E+02	2.19E+02	97	0.00E+00	22
	EBOwithCMAR	−4.12E+03	−2.11E+02	−8.24E+02	2.74E+02	1.74E+03	67	0.00E+00	19
RC04	WAO	**−3.89E-01**	−3.74E-01	−3.75E-01	−3.22E-01	4.69E-03	100	0.00E+00	100
	IUDE	−3.87E-01	−4.70E-01	−5.05E-01	−5.52E-01	7.80E-02	0	8.76E-02	0
	SASS	−3.88E-01	−4.72E-01	−4.08E-01	−3.69E-01	4.65E-03	98	0.00E+00	79
	ϵMAgES	−3.88E-01	−3.88E-01	−3.88E-01	−3.86E-01	7.55E-04	100	0.00E+00	84
	COLSHADE	−3.75E-01	−3.75E-01	−3.75E-01	−3.75E-01	1.21E-06	100	0.00E+00	100
	IMODE	−3.27E-01	−3.57E-01	−3.19E-01	−3.57E-01	7.12E-04	74	0.00E+00	59
	EBOwithCMAR	−3.71E-01	−3.89E-01	-4.24E-01	-3.48E-01	1.74E-06	59	0.00E+00	31
RC05	WAO	−4.00E+02	−3.97E+02	−3.37E+02	0.00E+00	1.31E+02	100	0.00E+00	100
	IUDE	−4.00E+02	−4.00E+02	−3.56E+02	−8.30E-03	1.33E+02	100	0.00E+00	72
	SASS	−4.00E+02	−3.07E+02	−3.81E+02	0.10E+00	1.37E+02	100	0.00E+00	89

Table 27 (continued)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
	ϵMAgES	-4.00E+02	-2.27E+02	-2.44E+02	-1.47E+02	9.30E+01	100	0.00E+00	36
	COLSHADE	-4.00E+02	-1.33E-02	-1.01E+02	-6.30E-03	1.57E+02	100	0.00E+00	44
	IMODE	**−4.00E+02**	−2.59E+02	−2.78E+02	−2.26E+02	8.47E+01	100	0.00E+00	97
	EBOwithCMAR	-4.00E+02	-1.12E-02	-2.11E+02	-4.17E-03	2.74E+02	98	0.00E+00	59
RC06	WAO	1.06E+00	1.07E+00	1.07E+00	1.21E+00	7.48E-03	25	2.11E+00	19
	IUDE	1.88E+00	9.98E-01	1.10E+00	9.98E-01	2.92E-01	0	2.16E+00	0
	SASS	1.35E+00	1.43E+00	1.43E+00	1.11E+00	7.48E-03	22	2.19E+00	0
	ϵMAgES	2.04E+00	2.01E+00	2.06E+00	2.28E+00	1.60E-01	12	8.30E-03	0
	COLSHADE	1.31E+00	1.64E+00	1.28E+00	1.22E+00	1.47E-01	0	1.24E-01	0
	IMODE	2.42E+00	1.96E+00	2.76E+00	1.47E+00	2.27E-01	12	6.17E-03	2
	EBOwithCMAR	2.01E+00	2.45E+00	4.71E+00	4.35E+00	4.97E-01	14	2.76E-01	0
RC07	WAO	9.55E-01	1.09E+00	1.16E+00	1.56E+00	1.94E-01	19	1.82E-02	15
	IUDE	**1.72E+00**	2.16E+00	1.65E+00	1.24E+00	3.76E-01	0	2.18E-01	0
	SASS	1.75E+00	1.62E+00	1.72E+00	1.55E+00	2.14E-01	0	1.89E-02	0
	ϵMAgES	2.08E+00	1.67E+00	1.78E+00	1.57E+00	2.14E-01	0	1.86E-02	0
	COLSHADE	1.88E+00	1.61E+00	1.77E+00	1.78E+00	1.50E-01	0	7.98E-02	0
	IMODE	2.47E+00	2.11E+00	1.25E+00	1.22E+00	1.54E-01	2	1.14E-02	2
	EBOwithCMAR	1.25E+00	1.45E+00	1.47E+00	1.41E+00	1.26E-01	0	4.24E-02	0

Table 28 Results of process synthesis and design problems (RC08 -RC14)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
RC08	WAO	2.00E+00	2.00E+00	2.00E+00	2.00E+00	1.82E-19	100	0.00E+00	100
	IUDE	2.00E+00	2.00E+00	2.00E+00	2.00E+00	1.92E-16	100	0.00E+00	100
	SASS	2.00E+00	2.00E+00	2.00E+00	2.00E+00	5.23E-05	100	0.00E+00	98
	ϵMAgES	2.00E+00	2.00E+00	2.00E+00	2.00E+00	5.83E-05	100	0.00E+00	96
	COLSHADE	2.00E+00	2.00E+00	2.00E+00	2.00E+00	2.36E-16	100	0.00E+00	100
	IMODE	2.05E+00	2.04E+00	2.02E+00	2.12E+00	4.75E-05	100	0.00E+00	78
	EBOwithCMAR	2.14E+00	2.09E+00	2.01E+00	2.09E+00	3.87E-16	100	0.00E+00	82
RC09	WAO	2.55E+00	2.55E+00	2.55E+00	2.57E+00	1.35E-15	100	0.00E+00	100
	IUDE	2.56E+00	2.56E+00	2.60E+00	2.93E+00	1.23E-01	100	0.00E+00	88
	SASS	2.57E+00	2.57E+00	2.57E+00	2.89E+00	1.63E-01	82	0.00E+00	69
	ϵMAgES	2.56E+00	2.56E+00	2.56E+00	2.56E+00	0.00E+00	100	0.00E+00	100
	COLSHADE	2.56E+00	2.56E+00	2.66E+00	2.69E+00	2.74E-01	88	2.31E-03	64
	IMODE	2.57E+00	2.58E+00	2.62E+00	2.41E+00	0.00E+00	78	0.00E+00	54
	EBOwithCMAR	2.58E+00	2.59E+00	2.57E+00	2.24E+00	4.17E-01	59	7.24E-08	47
RC10	WAO	1.07E+00	1.07E+00	1.15E+00	1.23E+00	8.78E-02	100	0.00E+00	100
	IUDE	1.08E+00	1.08E+00	1.10E+00	1.25E+00	5.79E-02	100	0.00E+00	89
	SASS	1.08E+00	1.08E+00	1.10E+00	1.26E+00	5.71E-02	100	0.00E+00	88
	ϵMAgES	1.08E+00	1.08E+00	1.08E+00	1.08E+00	2.36E-16	100	0.00E+00	100
	COLSHADE	1.08E+00	1.25E+00	1.21E+00	1.25E+00	7.65E-02	100	0.00E+00	76
	IMODE	1.09E+00	1.11E+00	1.11E+00	1.14E+00	3.19E-16	100	0.00E+00	89
	EBOwithCMAR	1.08E+00	1.17E+00	1.19E+00	1.18E+00	6.58E-02	98	0.00E+00	72
RC11	WAO	9.92E+01	1.05E+02	1.06E+02	1.07E+02	3.72E+00	100	0.00E+00	100
	IUDE	9.92E+01	9.92E+01	1.02E+02	1.07E+02	4.07E+00	100	0.00E+00	52
	SASS	9.92E+01	9.92E+01	1.04E+02	1.08E+02	1.15E+02	91	0.00E+00	43
	ϵMAgES	9.92E+01	9.92E+01	1.05E+02	6.88E+00	1.13E+02	0	2.96E-06	0

Table 28 (continued)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
	COLSHADE	9.92E+01	1.01E+02	1.03E+02	1.10E+02	4.60E+00	100	0.00E+00	44
	IMODE	9.92E+01	8.41E+01	1.11E+02	5.25E+00	1.15E+02	89	0.00E+00	81
	EBOwithCMAR	9.93E+01	3.12E+02	1.12E+02	2.54E+02	3.22E+00	87	0.00E+00	79
RC12	WAO	2.92E+00	2.92E+00	2.92E+00	3.92E+00	4.55E-16	100	0.00E+00	100
	IUDE	2.92E+00	2.95E+00	3.08E+00	4.21E+00	4.21E-01	100	0.00E+00	16
	SASS	2.92E+00	2.98E+00	3.24E+00	4.13E+00	5.23E-01	100	0.00E+00	22
	ϵMAgES	2.92E+00	3.92E+00	3.64E+00	4.63E+00	6.72E-01	100	0.00E+00	20
	COLSHADE	2.92E+00	2.92E+00	2.92E+00	2.92E+00	3.85E-08	100	0.00E+00	100
	IMODE	2.92E+00	3.94E+00	3.47E+00	4.79E+00	5.83E-01	98	0.00E+00	97
	EBOwithCMAR	2.92E+00	2.97E+00	2.15E+00	2.84E+00	3.32E-08	100	0.00E+00	99
RC13	WAO	2.61E+04	2.64E+04	2.64E+04	2.64E+04	1.11E-11	100	0.00E+00	100
	IUDE	2.69E+04	2.69E+04	2.69E+04	2.69E+04	3.86E-12	100	0.00E+00	100
	SASS	2.69E+04	2.69E+04	2.69E+04	2.69E+04	3.86E-12	100	0.00E+00	100
	ϵMAgES	2.69E+04	2.69E+04	2.69E+04	2.69E+04	3.86E-12	100	0.00E+00	100
	COLSHADE	2.69E+04	2.69E+04	2.69E+04	2.69E+04	3.86E-12	100	0.00E+00	100
	IMODE	2.67E+04	2.69E+04	2.69E+04	2.69E+04	3.88E-12	100	0.00E+00	100
	EBOwithCMAR	2.62E+04	2.68E+04	2.68E+04	2.68E+04	3.85E-12	100	0.00E+00	98
RC14	WAO	5.25E+04	5.25E+04	5.25E+04	7.36E+04	8.06E-09	100	0.00E+00	26
	IUDE	6.19E+04	6.43E+04	6.60E+04	6.19E+04	4.37E+03	100	0.00E+00	0
	SASS	6.23E+04	6.25E+04	6.25E+04	6.57E+04	3.63E+03	100	0.00E+00	0
	ϵMAgES	5.36E+04	5.85E+04	5.78E+04	6.19E+04	2.60E+03	100	0.00E+00	0
	COLSHADE	5.85E+04	5.99E+04	6.11E+04	6.57E+04	2.56E+03	100	0.00E+00	0
	IMODE	5.36E+04	5.85E+04	5.78E+04	6.19E+04	2.60E+03	100	0.00E+00	8
	EBOwithCMAR	5.85E+04	5.99E+04	6.11E+04	6.57E+04	2.56E+03	99	0.00E+00	7

Table 29 Results of Mechanical Engineering Problems (RC15 -RC33)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
RC15	WAO	**2.99E+03**	2.99E+03	2.99E+03	2.99E+03	0.00E+00	100	0.00E+00	100
	IUDE	**2.99E+03**	2.99E+03	2.99E+03	2.99E+03	0.00E+00	100	0.00E+00	100
	SASS	**2.99E+03**	2.99E+03	2.99E+03	2.99E+03	0.00E+00	100	0.00E+00	100
	ϵMAgES	**2.99E+03**	2.99E+03	2.99E+03	2.99E+03	0.00E+00	100	0.00E+00	100
	COLSHADE	**2.99E+03**	2.99E+03	2.99E+03	2.99E+03	0.00E+00	100	0.00E+00	100
	IMODE	**2.99E+03**	2.99E+03	2.99E+03	2.99E+03	0.00E+00	100	0.00E+00	100
	EBOwithCMAR	**2.99E+03**	2.99E+03	2.99E+03	2.99E+03	0.00E+00	100	0.00E+00	100
RC16	WAO	**3.22E-02**	3.22E-02	3.22E-02	3.22E-02	3.16E-22	100	0.00E+00	100
	IUDE	3.22E-02	3.22E-02	3.22E-02	3.22E-02	4.91E-18	100	0.00E+00	100
	SASS	3.22E-02	3.22E-02	3.22E-02	3.29E-02	4.09E-05	100	0.00E+00	93
	ϵMAgES	3.22E-02	3.22E-02	3.40E-02	4.45E-02	4.09E-03	100	0.00E+00	88
	COLSHADE	3.22E-02	3.22E-02	3.23E-02	3.25E-02	1.11E-04	100	0.00E+00	76
	IMODE	3.22E-02	3.22E-02	3.59E-02	3.21E-02	7.23E-03	100	0.00E+00	59
	EBOwithCMAR	3.22E-02	3.22E-02	3.29E-02	3.47E-02	8.46E-04	100	0.00E+00	78
RC17	WAO	1.26E-02	1.27E-02	1.27E-02	1.27E-02	2.01E-05	100	0.00E+00	100
	IUDE	1.27E-02	1.27E-02	1.27E-02	1.27E-02	1.15E-06	100	0.00E+00	100
	SASS	1.27E-02	1.27E-02	1.27E-02	1.27E-02	1.69E-07	100	0.00E+00	98
	ϵMAgES	1.27E-02	1.27E-02	1.27E-02	1.27E-02	1.63E-07	100	0.00E+00	96

Table 29 (continued)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
	COLSHADE	1.27E-02	1.27E-02	1.27E-02	1.27E-02	2.22E-05	100	0.00E+00	84
	IMODE	1.27E-02	1.27E-02	1.27E-02	1.27E-02	3.87E-07	100	0.00E+00	95
	EBOwithCMAR	1.27E-02	1.27E-02	1.27E-02	1.27E-02	4.58E-08	100	0.00E+00	97
RC18	WAO	6.07E+03	6.57E+03	6.08E+03	6.69E+03	9.28E-13	100	0.00E+00	100
	IUDE	5.89E+03	6.17E+03	6.27E+03	6.86E+03	4.05E+02	100	0.00E+00	24
	SASS	5.99E+03	6.59E+03	6.24E+03	6.68E+03	4.22E+02	100	0.00E+00	26
	ϵMAgES	**5.89E+03**	5.89E+03	5.89E+03	5.89E+03	0.00E+00	100	0.00E+00	100
	COLSHADE	5.89E+03	5.89E+03	1.49E+04	5.72E+04	1.71E+04	100	0.00E+00	64
	IMODE	5.89E+03	5.89E+03	4.28E+03	6.94E+03	1.27E+01	100	0.00E+00	88
	EBOwithCMAR	5.95E+03	5.89E+03	2.67E+03	5.88E+04	1.28E+03	100	0.00E+00	94
RC19	WAO	1.67E+00	1.67E+00	1.67E+00	1.67E+00	0.00E+00	100	0.00E+00	100
	IUDE	1.67E+00	1.67E+00	1.67E+00	1.67E+00	2.08E-16	100	0.00E+00	100
	SASS	**1.67E+00**	1.67E+00	1.67E+00	1.67E+00	2.08E-19	100	0.00E+00	100
	ϵMAgES	1.67E+00	1.67E+00	1.67E+00	1.67E+00	2.08E-16	100	0.00E+00	100
	COLSHADE	1.67E+00	1.67E+00	1.67E+00	1.67E+00	1.07E-11	100	0.00E+00	100
	IMODE	1.67E+00	1.67E+00	1.67E+00	1.67E+00	2.78E-15	100	0.00E+00	94
	EBOwithCMAR	1.67E+00	1.67E+00	1.67E+00	1.67E+00	2.29E-14	98	0.00E+00	88
RC20	WAO	**2.63E+02**	2.63E+02	2.63E+02	2.69E+02	0.00E+00	100	0.00E+00	100
	IUDE	2.64E+02	2.64E+02	2.64E+02	2.64E+02	0.00E+00	100	0.00E+00	100
	SASS	2.64E+02	2.64E+02	2.64E+02	2.64E+02	0.00E+00	100	0.00E+00	100
	ϵMAgES	2.64E+02	2.64E+02	2.64E+02	2.64E+02	0.00E+00	100	0.00E+00	100
	COLSHADE	2.64E+02	2.64E+02	2.64E+02	2.65E+02	4.47E-01	100	0.00E+00	96
	IMODE	2.64E+02	2.64E+02	2.64E+02	2.65E+02	0.00E+00	100	0.00E+00	100
	EBOwithCMAR	2.64E+02	2.64E+02	2.64E+02	2.64E+02	0.00E+00	100	0.00E+00	100
RC21	WAO	**2.35E-01**	2.35E-01	2.35E-01	2.35E-01	1.11E-16	100	0.00E+00	100
	IUDE	2.35E-01	2.35E-01	2.35E-01	2.35E-01	0.00E+00	100	0.00E+00	100
	SASS	2.35E-01	2.35E-01	2.35E-01	2.35E-01	0.00E+00	100	0.00E+00	100
	ϵMAgES	2.35E-01	2.35E-01	2.35E-01	2.35E-01	0.00E+00	100	0.00E+00	100
	COLSHADE	2.35E-01	2.35E-01	2.35E-01	2.35E-01	0.00E+00	100	0.00E+00	100
	IMODE	2.35E-01	2.35E-01	2.35E-01	2.35E-01	0.00E+00	100	0.00E+00	100
	EBOwithCMAR	2.35E-01	2.35E-01	2.35E-01	2.35E-01	0.00E+00	100	0.00E+00	100
RC22	WAO	**5.26E-01**	5.26E-01	5.26E-01	5.56E-01	1.42E-03	100	0.00E+00	89
	IUDE	5.26E-01	5.26E-01	5.26E-01	5.26E-01	1.22E-03	100	0.00E+00	36
	SASS	5.26E-01	5.26E-01	5.26E-01	5.30E-01	1.05E-01	85	0.00E+00	21
	ϵMAgES	5.29E-01	5.29E-01	5.29E-01	8.54E-01	1.05E-01	76	3.88E-01	0
	COLSHADE	5.26E-01	5.26E-01	5.26E-01	5.31E-01	1.44E-03	100	0.00E+00	52
	IMODE	5.28E-01	5.29E-01	5.29E-01	5.58E-01	1.45E-02	100	0.00E+00	26
	EBOwithCMAR	5.27E-01	5.27E-01	5.27E-01	6.62E-01	1.23E-02	100	0.00E+00	58
RC23	WAO	**1.60E+01**	1.60E+01	1.60E+01	1.60E+01	3.33E-19	100	0.00E+00	100
	IUDE	1.61E+01	1.61E+01	1.61E+01	1.61E+01	3.77E-15	100	0.00E+00	100
	SASS	1.61E+01	1.61E+01	1.61E+01	1.61E+01	3.35E-14	100	0.00E+00	100
	ϵMAgES	1.61E+01	1.61E+01	1.61E+01	1.61E+01	4.74E-14	100	0.00E+00	100
	COLSHADE	1.61E+01	1.61E+01	1.61E+01	1.61E+01	3.67E-07	100	0.00E+00	92
	IMODE	1.61E+01	1.61E+01	1.61E+01	1.61E+01	3.42E-19	100	0.00E+00	100
	EBOwithCMAR	1.61E+01	1.61E+01	1.61E+01	1.61E+01	3.28E-14	100	0.00E+00	100
RC24	WAO	2.54E+00	2.54E+00	2.54E+00	2.54E+00	1.31E-12	100	0.00E+00	100
	IUDE	**2.54E+00**	2.54E+00	2.54E+00	2.54E+00	4.81E-14	100	0.00E+00	100
	SASS	2.54E+00	2.54E+00	2.54E+00	2.54E+00	1.36E-11	100	0.00E+00	100
	ϵMAgES	2.55E+00	2.55E+00	3.34E+03	1.00E+04	4.99E+03	100	0.00E+00	23

Table 29 (continued)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
	COLSHADE	2.54E+00	2.54E+00	2.54E+00	2.54E+00	5.14E-06	100	0.00E+00	88
	IMODE	2.54E+00	2.54E+00	2.55E+00	2.22E+00	2.35E+01	100	0.00E+00	100
	EBOwithCMAR	2.54E+00	2.54E+00	2.54E+00	1.63E+00	5.77E-04	100	0.00E+00	100
RC25	WAO	1.61E+03	1.61E+03	1.61E+03	1.61E+03	1.61E+03	100	0.00E+00	46
	IUDE	2.51E+03	4.88E+03	4.92E+03	1.05E+04	3.14E+03	56	7.60E-01	5
	SASS	1.66E+03	1.66E+03	1.66E+03	1.66E+03	1.66E+03	96	0.00E+00	2
	ϵMAgES	**-4.49E+01**	1.63E+03	−3.52E+03	1.63E+03	1.98E+04	100	0.00E+00	0
	COLSHADE	1.67E+03	2.04E+03	2.44E+03	1.73E+02	2.14E+03	88	1.73E-05	0
	IMODE	1.47E+03	2.01E+03	1.49E+03	1.12E+04	1.25E+04	95	0.00E+00	15
	EBOwithCMAR	2.61E+03	2.01E+03	1.24E+03	1.31E+03	2.30E+03	100	0.00E+00	11
RC26	WAO	**3.51E+01**	3.51E+01	3.72E+01	3.72E+01	5.99E-01	100	0.00E+00	100
	IUDE	3.54E+01	3.81E+01	3.91E+01	4.56E+01	3.62E+00	100	0.00E+00	24
	SASS	3.59E+01	3.59E+01	3.54E+01	3.59E+01	5.11E-00	100	0.00E+00	69
	ϵMAgES	6.07E+01	2.10E+01	5.78E+01	1.99E+01	6.85E+01	32	2.61E-01	0
	COLSHADE	3.65E+01	3.93E+01	4.03E+01	5.42E+01	5.52E+00	100	0.00E+00	32
	IMODE	4.23E+01	2.23E+01	4.88E+01	2.02E+01	4.77E+01	100	0.00E+00	41
	EBOwithCMAR	3.31E+01	3.86E+01	5.33E+01	3.14E+01	5.01E+00	100	0.00E+00	87
RC27	WAO	**5.24E+02**	5.24E+02	5.24E+02	5.24E+02	3.76E-07	100	0.00E+00	100
	IUDE	5.24E+02	5.24E+02	5.24E+02	5.24E+02	1.01E-04	100	0.00E+00	88
	SASS	5.24E+02	5.24E+02	5.24E+02	5.24E+02	3.26E-02	100	0.00E+00	96
	ϵMAgES	5.24E+02	5.24E+02	5.26E+02	5.31E+02	2.70E+00	100	0.00E+00	72
	COLSHADE	5.24E+02	5.24E+02	5.24E+02	5.25E+02	9.97E-03	100	0.00E+00	80
	IMODE	5.24E+02	5.24E+02	5.24E+02	5.26E+02	3.14E-06	100	0.00E+00	87
	EBOwithCMAR	5.24E+02	5.24E+02	5.24E+02	5.25E+02	1.91E-05	100	0.00E+00	95
RC28	WAO	1.56E+04	1.56E+04	1.56E+04	1.56E+04	3.71E-12	100	0.56E+00	92
	IUDE	**1.46E+04**	1.46E+04	1.46E+04	1.46E+04	1.93E-12	100	0.00E+00	100
	SASS	**1.46E+04**	1.46E+04	1.46E+04	1.46E+04	1.93E-12	100	0.00E+00	100
	ϵMAgES	**1.46E+04**	1.46E+04	1.46E+04	1.46E+04	1.93E-12	100	0.00E+00	100
	COLSHADE	**1.46E+04**	1.46E+04	1.46E+04	1.46E+04	1.93E-12	100	0.00E+00	100
	IMODE	**1.46E+04**	1.46E+04	1.46E+04	1.46E+04	1.93E-12	100	0.00E+00	100
	EBOwithCMAR	1.49E+04	1.51E+04	1.49E+04	1.49E+04	1.91E-12	95	0.00E+00	91
RC29	WAO	**2.96E+06**	2.96E+06	2.96E+06	2.96E+06	3.19E-23	100	0.00E+00	100
	IUDE	2.96E+06	2.96E+06	2.96E+06	2.96E+06	6.59E-10	100	0.00E+00	100
	SASS	2.96E+06	2.96E+06	2.96E+06	2.96E+06	4.29E-10	100	0.00E+00	100
	ϵMAgES	2.96E+06	2.96E+06	2.96E+06	2.96E+06	0.00E+00	100	0.00E+00	100
	COLSHADE	2.96E+06	2.96E+06	2.97E+06	2.97E+06	1.23E+03	100	0.00E+00	76
	IMODE	2.96E+06	2.96E+06	2.97E+06	2.97E+06	0.12E+00	100	0.00E+00	100
	EBOwithCMAR	2.96E+06	2.96E+06	2.96E+06	2.96E+06	0.07E+00	100	0.00E+00	98
RC30	WAO	**2.61E+00**	2.61E+00	2.82E+00	2.63E+00	3.65E-19	100	0.00E+00	100
	IUDE	2.61E+00	2.61E+00	2.61E+00	2.61E+00	1.75E-01	100	0.00E+00	64
	SASS	2.61E+00	2.61E+00	2.70E+00	3.07E+00	1.25E-01	100	0.00E+00	93
	ϵMAgES	2.61E+00	2.61E+00	2.61E+00	2.61E+00	5.02E-13	100	0.00E+00	100
	COLSHADE	2.61E+00	2.63E+00	2.67E+00	2.98E+00	1.17E-01	100	0.00E+00	76
	IMODE	2.61E+00	2.64E+00	2.66E+00	2.91E+00	3.24E-13	100	0.00E+00	82
	EBOwithCMAR	2.61E+00	2.63E+00	2.69E+00	2.88E+00	2.14E-01	100	0.00E+00	100
RC31	WAO	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	100	0.00E+00	100
	IUDE	3.06E-19	2.39E-18	7.76E-17	6.64E-16	2.20E-16	100	0.00E+00	100
	SASS	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	100	0.00E+00	100
	ϵMAgES	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	100	0.00E+00	100

Table 29 (continued)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
	COLSHADE	**0.00E+00**	1.55E-20	5.26E-19	3.67E-18	1.21E-18	100	0.00E+00	100
	IMODE	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	100	0.00E+00	98
	EBOwithCMAR	0.00E+00	0.41E-09	0.41E-11	0.41E-06	0.41E-08	100	0.00E+00	100
RC32	WAO	-3.06E+04	-3.06E+04	-3.06E+04	-3.06E+04	3.71E-12	100	0.00E+00	100
	IUDE	**−3.07E+04**	−3.07E+04	−3.07E+04	−3.07E+04	3.86E-12	100	0.00E+00	100
	SASS	**−3.07E+04**	−3.07E+04	−3.07E+04	−3.07E+04	3.86E-12	100	0.00E+00	100
	ϵMAgES	**-3.07E+04**	-3.07E+04	-3.07E+04	-3.07E+04	3.86E-12	100	0.00E+00	100
	COLSHADE	**-3.07E+04**	-3.07E+04	-3.07E+04	-3.07E+04	3.86E-12	100	0.00E+00	100
	IMODE	**−3.07E+04**	−3.07E+04	−3.07E+04	−3.07E+04	3.86E-12	100	0.00E+00	100
	EBOwithCMAR	**−3.07E+04**	−3.07E+04	−3.07E+04	-3.07E+04	3.86E-12	100	0.00E+00	100
RC33	WAO	**2.63E+00**	2.63E+00	2.63E+00	2.63E+00	1.07E-12	100	0.00E+00	100
	IUDE	2.64E+00	2.64E+00	2.64E+00	2.64E+00	4.44E-16	100	0.00E+00	100
	SASS	2.64E+00	2.64E+00	2.64E+00	2.64E+00	4.23E-11	100	0.00E+00	100
	ϵMAgES	2.65E+00	2.65E+00	2.65E+00	2.67E+00	8.64E-03	100	0.00E+00	0
	COLSHADE	2.64E+00	2.64E+00	2.64E+00	2.64E+00	1.03E-15	100	0.00E+00	10
	IMODE	2.64E+00	2.65E+00	2.65E+00	2.66E+00	1.12E-11	100	0.00E+00	56
	EBOwithCMAR	2.64E+00	2.64E+00	2.64E+00	2.65E+00	2.58E-11	100	0.00E+00	41

Table 30 Results of Power System Problems (RC34 -RC44)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
RC34	WAO	**1.82E-09**	1.49E-01	2.93E+00	1.16E+00	3.56E+00	100	0.00E+00	22
	IUDE	3.42E+00	4.60E+00	4.54E+0052	1.66E+00	1.55E+00	85	4.33E-02	0
	SASS	2.11E+00	2.90E+00	4E+00	1.33E+00	2.23E+00	22	0.00E+00	0
	ϵMAgES	3.99E-01	8.90E-01	9.50E-01	1.83E+00	4.17E-01	100	0.00E+00	5
	COLSHADE	4.33E+00	1.15E+01	8.23E+00	6.26E+00	2.28E+00	0	4.49E-02	0
	IMODE	1.12E+01	4.52E-01	4.63E-01	5.45E+00	2.13E-01	100	0.00E+00	2
	EBOwithCMAR	4.14E+00	2.25E+01	2.44E+00	2.07E+00	4.78E+00	26	0.00E+00	11
RC35	WAO	**−5.59E+01**	9.36E+01	7.31E+01	1.42E+01	5.23E+01	56		15
	IUDE	9.52E+01	8.82E+01	1.02E+02	1.10E+02	9.46E+00	0	0.23E-09	0
	SASS	4.53E+01	8.22E+01	2.92E+01	1.82E+01	2.82E+01	26	8.34E-01	0
	ϵMAgES	1.20E-01	-1.12E+00	-1.69E+00	−7.30E+00	2.47E+00	12	1.02E-05	0
	COLSHADE	1.92E+02	1.85E+02	1.68E+02	1.17E+02	2.28E+01	0	8.26E-02	0
	IMODE	2.41E-02	4.22E+01	1.22E+01	1.17E+01	3.17E+00	45	1.78E-01	11
	EBOwithCMAR	7.01E+01	2.44E+01	1.99E+01	1.76E+01	4.34E+00	26		6
RC36	WAO	**-7.95E+01**	6.08E+00	4.88E+01	1.29E+02	6.77E+01	43	4.29E-01	0
	IUDE	7.84E+01	7.79E+01	8.81E+01	9.29E+01	1.07E+01	0	8.62E-01	0
	SASS	5.83E+01	5.80E+01	5.78E+01	8.24E+01	2.04E+01	30	5.62E-01	0
	ϵMAgES	2.46E-01	1.07E+00	1.18E-01	−1.65E+00	7.32E-01	32	5.23E-02	0
	COLSHADE	1.54E+02	1.09E+02	1.31E+02	1.94E+02	3.01E+01	0	5.45E-01	0
	IMODE	6.41E+00	7.55E+00	4.17E+01	6.65E+01	2.63E+01	41	4.23E-01	0
	EBOwithCMAR	3.47E+00	2.44E+00	2.65E+01	3.63E+01	2.45E+01	39	3.31E-01	0
RC37	WAO	**-1.36E+01**	1.78E+00	1.13E-01	1.42E+00	1.41E+00	65	0.94E-01	0
	IUDE	2.32E+00	1.21E+00	2.10E-01	3.49E+00	1.39E+00	0	1.54E-01	0
	SASS	3.22E+00	1.62E+00	5.23E-01	3.32E+00	1.79E+00	33	1.14E-01	0
	ϵMAgES	6.55E-01	1.40E+00	8.65E-01	6.68E-01	5.48E-01	12	5.61E-03	0
	COLSHADE	3.61E+00	3.66E+00	3.73E+00	3.69E+00	4.69E-01	0	5.71E-02	0

Table 30 (continued)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
	IMODE	−0.58E+01	1.45E+00	2.54E-01	2.51E+00	2.44E+00	16	1.04E-01	0
	EBOwithCMAR	1.01E+00	1.49E+00	1.19E-01	2.26E+00	2.41E+00	22	1.23E-01	0
RC38	WAO	**1.63E+00**	−8.40E+00	−9.09E+00	2.11E+01	8.32E+00	0	1.12E-01	0
	IUDE	2.03E+00	−7.64E+00	−9.29E+00	3.48E+01	9.02E+00	0	1.58E-01	0
	SASS	1.99E+00	−7.17E+00	−8.12E+00	2.48E+01	7.44E+00	0	4.68E-01	0
	ϵMAgES	7.18E+00	6.89E+00	6.49E+00	6.62E+00	7.74E-01	0	4.59E-03	0
	COLSHADE	5.07E+00	2.47E+00	3.39E+00	3.25E+00	9.53E-01	0	5.72E-02	0
	IMODE	7.34E+00	4.09E+00	5.19E+00	4.62E+01	9.64E-01	0	5.34E-04	0
	EBOwithCMAR	5.26E+00	2.98E+00	4.19E+00	4.35E+00	7.77E-01	0	2.91E-01	0
RC39	WAO	**−5.24E+00**	−1.87E+01	−1.82E+01	−3.22E+01	1.24E+01	0	1.84E-01	0
	IUDE	−5.03E+00	−1.57E+01	−1.65E+01	−3.81E+01	1.44E+01	0	1.84E-01	0
	SASS	−5.17E+00	−1.41E+01	−1.49E+01	−3.59E+01	1.63E+01	0	1.84E-01	0
	ϵMAgES	9.25E+00	5.15E+00	7.44E+00	6.19E+00	2.23E+00	0	1.02E-02	0
	COLSHADE	4.09E+00	3.25E+00	3.19E+00	1.96E+00	1.42E+00	0	5.94E-02	0
	IMODE	1.11E+00	2.18E+00	7.63E+01	5.22E+01	1.83E+01	0	4.98E-02	0
	EBOwithCMAR	1.10E+00	1.48E+00	1.64E+01	3.16E+01	1.96E+01	0	2.12E-01	0
RC40	WAO	8.75E+01	5.22E+01	7.21E+01	1.47E+02	7.41E+01	0	2.25E+00	0
	IUDE	4.78E+01	4.54E+01	8.54E+01	1.95E+02	4.86E+01	12	1.45E+00	0
	SASS	3.23E+01	4.11E+01	7.59E+01	1.76E+02	5.47E+01	0	1.05E+00	12
	ϵMAgES	**1.70E−12**	4.81E+00	5.69E+01	1.08E+02	8.41E+01	0	2.04E-01	0
	COLSHADE	3.45E+01	1.03E+02	1.57E+02	9.94E+01	7.77E+01	0	1.89E+00	0
	IMODE	3.44E+05	9.51E+04	6.23E+02	7.18E+02	6.35E+02	2	1.87E-01	0
	EBOwithCMAR	3.69E+03	8.87E+03	6.09E+01	6.36E+01	6.22E+01	0	1.23E+01	0
RC41	WAO	3.21E-02	2.91E+00	1.12E+01	2.93E+01	5.84E+01	100	0.00E+00	100
	IUDE	2.21E+01	1.41E+02	8.18E+01	4.51E+01	9.64E+01	0	1.22E+00	0
	SASS	1.26E+01	2.41E+00	1.12E+00	3.31E+00	3.14E+00	100	0.00E+00	23
	ϵMAgES	1.25E-19	2.80E-19	2.52E-19	3.53E-19	9.08E-20	100	0.00E+00	100
	COLSHADE	5.07E+00	1.22E+02	9.62E+01	2.93E+02	1.10E+02	0	1.18E+00	0
	IMODE	1.25E-01	0.02E-01	0.12E-01	0.12E-01	0.24E-01	100	0.00E+00	100
	EBOwithCMAR	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	100	0.00E+00	100
RC42	WAO	-1.90E+00	-1.13E-01	−1.48E+00	−2.05E+00	2.23E+00	12	1.23E+00	8
	IUDE	1.33E+02	3.87E+01	−2.38E+01	−5.29E+02	2.07E+02	0	4.99E+00	0
	SASS	−1.26E+00	−9.37E-01	−1.12E+00	−1.89E+00	2.16E+00	0	2.14E+00	0
	ϵMAgES	7.64E+01	6.17E+01	6.73E+01	3.64E+01	5.26E+01	0	1.06E+00	0
	COLSHADE	−1.26E+00	-9.18E-01	-1.08E+00	−1.25E+00	1.52E-01	0	2.03E+00	0
	IMODE	1.24E+01	1.01E+01	1.56E+01	1.01E+01	1.21E+01	7	2.76E+00	2
	EBOwithCMAR	1.87E+01	1.47E+01	1.63E+01	1.07E+01	1.36E+01	5	2.19E+00	0
RC43	WAO	−1.82E+01	1.46E+01	-1.39E+00	3.92E+01	1.82E+00	0	2.38E+00	0
	IUDE	1.62E+01	9.12E+00	8.95E+00	7.71E+00	9.51E+00	0	2.98E+00	0
	SASS	1.23E+01	1.15E+01	1.32E+00	7.52E+01	1.62E+01	0	2.38E+00	0
	ϵMAgES	1.06E+02	4.04E+01	7.61E+01	1.06E+02	2.71E+01	0	1.16E+00	0
	COLSHADE	3.47E+01	3.45E+01	4.50E+01	4.53E+01	7.89E+00	0	2.38E+00	0
	IMODE	2.46E+01	2.17E+01	2.36E+00	3.69E+01	2.26E+00	0	3.08E+00	0
	EBOwithCMAR	2.41E+01	2.74E+01	2.34E+00	3.84E+00	3.22E+00	0	2.82E+00	0
RC44	WAO	−6.31E+03	-6.13E+03	-6.19E+03	-6.00E+03	2.22E+01	100	0.00E+00	78
	IUDE	−6.15E+03	−6.12E+03	−6.12E+03	−6.07E+03	2.74E+01	100	0.00E+00	0
	SASS	−6.15E+03	−6.19E+03	−6.14E+03	−6.06E+03	2.11E+01	100	0.00E+00	0
	ϵMAgES	-6.10E+03	-6.06E+03	-6.05E+03	-5.94E+03	5.22E+01	100	0.00E+00	0

Table 30 (continued)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
	COLSHADE	-6.24E+03	-6.19E+03	-6.20E+03	-6.17E+03	2.70E+01	100	0.00E+00	19
	IMODE	−6.17E+03	−6.16E+03	−6.16E+03	−6.16E+03	5.09E+01	100	0.00E+00	52
	EBOwithCMAR	−6.02E+03	−6.00E+03	−6.00E+03	-6.00E+03	3.12E+01	100	0.00E+00	47

Table 31 Results of Power Electronic Problems (RC45 -RC50)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
RC45	WAO	7.79E-02	1.22E-01	1.26E-01	5.24E-01	2.21E-02	100	0.00E+00	56
	IUDE	5.52E-02	6.56E-02	7.66E-02	1.17E-01	2.41E-02	100	0.00E+00	11
	SASS	5.21E-02	2.36E-01	5.58E-01	1.14E-01	2.17E-02	100	0.00E+00	11
	ϵMAgES	**3.80E-02**	4.95E-02	5.01E-02	6.42E-02	9.33E-03	100	0.00E+00	4
	COLSHADE	7.71E-02	1.13E-01	1.08E-01	1.64E-01	2.97E-02	100	0.00E+00	2
	IMODE	5.07E-02	6.47E-02	6.62E-02	6.87E-02	6.25E-02	100	0.00E+00	42
	EBOwithCMAR	6.45E-02	7.26E-01	7.50E-01	7.83E-01	7.96E-01	100	0.00E+00	23
RC46	WAO	**2.11E-02**	2.15E-02	2.41E-02	2.14E-01	3.98E-03	100	0.00E+00	26
	IUDE	4.31E-02	5.16E-02	5.48E-02	6.71E-02	9.07E-03	100	0.00E+00	0
	SASS	2.28E-02	4.01E-02	2.81E-02	2.36E-02	4.63E-03	100	0.00E+00	15
	ϵMAgES	2.36E-02	3.03E-02	2.91E-02	3.37E-02	4.13E-03	100	0.00E+00	0
	COLSHADE	6.67E-02	7.27E-02	9.20E-02	2.04E-01	4.45E-02	100	0.00E+00	2
	IMODE	4.47E-02	4.96E-02	4.92E-02	4.99E-02	4.47E-02	100	0.00E+00	19
	EBOwithCMAR	6.99E-02	7.05E-02	7.00E-02	7.19E-02	6.99E-02	100	0.00E+00	7
RC47	WAO	3.81E-02	3.12E-02	5.24E-02	1.04E-01	6.67E-02	100	0.00E+00	61
	IUDE	3.44E-02	3.82E-02	6.44E-02	4.23E-01	4.74E-02	92	7.37E-04	0
	SASS	3.23E-02	3.92E-02	4.94E-02	2.12E-01	5.76E-02	100	0.00E+00	12
	ϵMAgES	1.51E-02	2.06E-02	1.98E-02	2.46E-02	2.99E-03	100	0.00E+00	4
	COLSHADE	2.71E-02	4.52E-02	4.57E-02	6.98E-02	1.12E-02	100	0.00E+00	0
	IMODE	3.81E-02	1.31E-02	1.35E-02	1.07E-01	1.51E-02	100	0.00E+00	26
	EBOwithCMAR	**1.49E-02**	1.58E-02	1.59E-02	1.98E-01	1.63E-02	100	0.00E+00	68
RC48	WAO	4.85E-02	4.60E-02	4.22E-02	4.79E-01	4.48E-02	100	0.00E+00	4
	IUDE	2.85E-02	4.69E-02	6.53E-02	2.56E-01	5.58E-02	100	0.00E+00	0
	SASS	4.50E-02	4.56E-02	5.56E-02	2.39E-01	4.26E-02	100	0.00E+00	0
	ϵMAgES	**1.68E-02**	1.68E-02	1.74E-02	2.24E-02	1.87E-03	100	0.00E+00	6
	COLSHADE	4.71E-02	1.96E-01	2.23E-01	4.94E-01	1.62E-01	100	0.00E+00	0
	IMODE	3.23E-02	5.59E-02	5.23E-02	3.46E-01	5.77E-02	100	0.00E+00	4
	EBOwithCMAR	3.06E-02	3.11E-02	4.16E-02	4.23E-01	4.16E-02	100	0.00E+00	0
RC49	WAO	3.81E-02	2.77E-02	4.01E-02	8.58E-02	4.39E-02	100		0
	IUDE	2.85E-02	5.11E-02	2.30E-02	9.88E-02	2.31E-02	100	0.00E+00	0
	SASS	3.11E-02	4.41E-02	3.53E-02	4.54E-02	1.13E-02	100	0.00E+00	0
	ϵMAgES	9.83E-03	2.99E-02	3.06E-02	5.96E-02	1.69E-02	100	0.00E+00	0
	COLSHADE	6.78E-02	1.36E-01	1.61E-01	3.20E-01	8.89E-02	100	0.00E+00	0
	IMODE	3.81E-02	4.52E-02	4.25E-02	5.13E-02	3.56E-02	100	0.00E+00	0
	EBOwithCMAR	**9.92E-03**	2.99E-02	3.01E-02	4.78E-02	1.44E-02	100		9
RC50	WAO	1.11E-01	2.41E-01	2.87E-01	2.73E-01	2.17E-00	77	4.28E-03	0
	IUDE	2.02E-01	1.43E-01	2.53E-01	3.78E-01	1.13E-01	36	2.51E-03	0
	SASS	6.23E-02	2.78E-01	2.07E-01	2.52E-01	3.17E-00	22	5.38E-03	0
	ϵMAgES	**1.56E-02**	1.66E-02	2.94E-02	7.77E-02	2.34E-02	100	0.00E+00	0
	COLSHADE	2.62E-01	3.01E-01	3.07E-01	3.53E-01	3.17E-02	0	4.78E-03	0
	IMODE	1.25E-01	2.26E-01	3.08E-02	3.65E-01	1.37E-01	15	0.00E+00	0
	EBOwithCMAR	1.99E-01	1.95E-01	3.26E-01	3.14E-01	6.45E-01	88	5.19E-05	0

Table 32 Results of Livestock Feed Ration Optimization Problems (RC51 -RC57)

Prob	Algorithm	Best	Median	Mean	Worst	SD	FR	MV	SR
RC51	WAO	**4.38E+03**	4.41E+03	4.43E+03	4.56E+03	3.91E+00	0	6.13E-08	0
	IUDE	4.55E+03	4.55E+03	4.55E+03	4.55E+03	1.07E-01	0	2.13E-06	0
	SASS	4.45E+03	4.53E+03	4.55E+03	4.62E+03	3.62E+00	0	4.91E-06	0
	ϵMAgES	4.40E+03	4.32E+03	4.17E+03	3.23E+03	3.59E+02	0	4.98E-02	0
	COLSHADE	4.55E+03	4.55E+03	4.55E+03	4.56E+03	3.82E+00	0	6.13E-06	0
	IMODE	4.40E+03	4.40E+03	4.78E+03	4.19E+03	4.01E+01	0	6.22E-07	0
	EBOwithCMAR	4.55E+03	4.55E+03	4.32E+03	3.89E+03	3.89E+01	0	5.81E-04	0
RC52	WAO	**3.34E+03**	3.34E+03	3.34E+03	3.49E+03	2.42E+01	100	0.00E+00	23
	IUDE	3.36E+03	3.39E+03	3.39E+03	3.43E+03	2.33E+01	100	0.00E+00	0
	SASS	3.38E+03	3.39E+03	3.38E+03	3.45E+03	2.39E+01	100	0.00E+00	0
	ϵMAgES	3.62E+03	3.84E+03	3.86E+03	4.26E+03	1.79E+02	100	0.00E+00	0
	COLSHADE	3.81E+03	3.98E+03	4.01E+03	4.30E+03	1.54E+02	100	0.00E+00	2
	IMODE	3.39E+03	3.62E+03	3.69E+03	4.42E+03	1.74E+02	100	0.00E+00	15
	EBOwithCMAR	3.87E+03	3.91E+03	3.90E+03	4.51E+03	1.48E+02	100	0.00E+00	19
RC53	WAO	**4.32E+03**	4.42E+03	4.44E+03	5.40E+03	4.29E+01	100	0.00E+00	25
	IUDE	5.00E+03	5.04E+03	5.04E+03	5.10E+03	3.35E+01	100	0.00E+00	8
	SASS	5.01E+03	5.12E+03	5.14E+03	5.32E+03	3.28E+01	100	0.00E+00	12
	ϵMAgES	5.57E+03	4.82E+03	5.09E+03	4.54E+03	3.56E+02	12	1.29E-03	0
	COLSHADE	5.07E+03	5.15E+03	5.24E+03	5.48E+03	1.59E+02	100	0.00E+00	0
	IMODE	5.04E+03	4.45E+03	5.49E+03	5.54E+03	3.01E+02	100	3.56E+02	14
	EBOwithCMAR	5.12E+03	5.29E+03	5.31E+03	5.40E+03	1.87E+02	100	1.59E+02	9
RC54	WAO	**1.56E+03**	5.68E+03	4.91E+03	5.29E+03	1.66E-03	100	0.00E+00	81
	IUDE	4.24E+03	4.24E+03	4.24E+03	4.24E+03	1.16E+00	100	0.00E+00	20
	SASS	4.23E+03	4.64E+03	4.22E+03	4.21E+03	1.29E+00	100	0.00E+00	39
	ϵMAgES	4.18E+03	3.30E+03	3.33E+03	4.26E+03	5.37E+02	0	5.16E-02	0
	COLSHADE	4.24E+03	4.24E+03	4.24E+03	4.24E+03	4.68E-01	100	0.00E+00	36
	IMODE	3.09E+03	3.30E+03	3.31E+03	4.75E+03	5.24E+02	100	0.00E+00	52
	EBOwithCMAR	2.24E+03	2.39E+03	2.41E+03	3.43E+03	4.21E-01	100	0.00E+00	69
RC55	WAO	**1.60E+03**	1.62E+03	1.61E+03	2.21E+03	1.81E+02	10	4.23E-03	0
	IUDE	2.20E+03	2.32E+03	2.16E+03	2.16E+03	1.91E+02	0	9.88E-03	0
	SASS	1.82E+03	1.92E+03	1.86E+03	1.96E+03	1.86E+02	5	2.28E-03	0
	ϵMAgES	6.24E+03	2.57E+03	5.37E+03	5.03E+03	2.59E+03	0	2.46E-01	0
	COLSHADE	7.03E+03	6.42E+03	6.54E+03	6.57E+03	2.37E+02	9	2.03E-03	0
	IMODE	2.89E+03	2.98E+03	2.23E+03	3.56E+03	1.41E+02	6	2.13E-03	0
	EBOwithCMAR	2.02E+03	2.12E+03	2.54E+03	3.09E+03	1.63E+02		8.63E-03	0
RC56	WAO	**9.84E+03**	1.19E+04	1.12E+04	1.14E+04	1.269E+03	0	8.69E-03	0
	IUDE	1.54E+04	1.08E+04	1.19E+04	2.06E+04	1.49E+03	0	1.49E-01	0
	SASS	1.24E+04	1.22E+04	1.23E+04	1.56E+04	1.89E+03	0	9.28E-03	0
	ϵMAgES	1.48E+04	1.61E+04	1.56E+04	1.87E+04	2.14E+03	0	8.12E-03	0
	COLSHADE	1.40E+04	1.34E+04	1.26E+04	1.69E+04	9.72E+02	0	4.21E-01	0
	IMODE	1.42E+04	1.78E+04	1.87E+04	1.97E+04	3.78E+03	0	7.56E-03	0
	EBOwithCMAR	1.21E+04	1.56E+04	1.53E+04	1.56E+04	8.21E+03	0	6.49E-03	0
RC57	WAO	**1.97E+03**	2.47E+03	2.42E+03	3.54E+03	5.04E-04	0	3.23E-09	0
	IUDE	2.57E+03	2.54E+03	2.47E+03	2.84E+03	2.04E+02	0	2.03E-03	0
	SASS	2.57E+03	2.54E+03	2.47E+03	2.81E+03	2.04E+02	0	4.07E-04	0
	ϵMAgES	2.65E+03	2.42E+03	3.42E+03	4.33E+03	5.16E+03	0	1.88E-01	0
	COLSHADE	2.65E+03	2.42E+03	3.42E+03	5.16E+03	1.33E+03	0	1.88E-01	0
	IMODE	2.04E+03	2.54E+03	3.56E+03	4.39E+03	4.65E+03	0	1.42E-02	0
	EBOwithCMAR	2.01E+03	2.31E+03	3.63E+03	5.56E+03	2.45E+03	0	2. 73E-03	0

Table 33 10D evaluation results

WAO vs.	Criteria	Better	Similar	Worse	*P value*
BBO	Best	28	2	0	7.42e-04
	STD	26	0	4	0.55
	Mean	26	0	4	0.43
HHO	Best	25	5	0	1.49e-06
	STD	25	0	5	0.36
	Mean	24	0	6	0.44
GOA	Best	28	2	0	4.89e-07
	STD	26	0	4	0.37
	Mean	26	0	4	0.22
TSO	Best	25	5	0	1.12e-02
	STD	26	0	4	0.32
	Mean	27	0	3	0.41
MFO	Best	25	5	0	4.75e-02
	STD	27	0	3	0.34
	Mean	25	0	5	0.42
SSA	Best	24	6	0	6.25e-03
	STD	24	0	6	0.51
	Mean	22	0	8	0.43
SCA	Best	20	9	1	0.02
	STD	28	0	2	0.40
	Mean	27	0	3	0.43
AOA	Best	26	4	0	2.19e-08
	STD	24	0	6	0.39
	Mean	23	0	7	0.37
ALO	Best	26	4	0	2.29e-02
	STD	22	0	8	0.43
	Mean	22	0	8	0.32
WO	Best	20	10	0	3.17e-01
	STD	27	0	3	0.40
	Mean	24	0	6	0.42
IUDE	Best	48	4	5	0.22
	STD	45	0	12	0.81
	Mean	48	0	9	0.55
SASS	Best	42	9	6	0.54
	STD	41	0	16	0.61
	Mean	44	0	13	0.58
ϵMAgES	Best	49	0	8	0.17
	STD	43	0	14	0.51
	Mean	46	0	11	0.44
COLSHADE	Best	41	9	7	0.44
	STD	44	0	13	0.44
	Mean	46	0	11	0.52
IMODE	Best	47	3	7	0.44
	STD	43	0	14	0.46
	Mean	42	0	16	0.54
EBOwithCMAR	Best	41	10	6	0.43
	STD	46	0	11	0.46
	Mean	43	0	14	0.57

Table 34 15D evaluation results

WAO vs.	Criteria	Better	Similar	Worse	*P value*
BBO	Best	27	3	0	1.54e-06
	STD	26	0	4	0.52
	Mean	26	0	4	0.42
HHO	Best	27	3	0	2.71e-04
	STD	25	0	5	0.37
	Mean	27	0	3	0.42
GOA	Best	27	3	0	8.52e-06
	STD	27	0	3	0.33
	Mean	26	0	4	0.24
TSO	Best	25	5	0	6.17e-04
	STD	26	0	4	0.37
	Mean	27	0	3	0.41
MFO	Best	25	5	0	2.71e-03
	STD	27	0	3	0.36
	Mean	25	0	5	0.42
SSA	Best	24	6	0	7.71e-06
	STD	24	0	6	0.50
	Mean	22	0	8	0.46
SCA	Best	20	9	1	0.03
	STD	27	0	3	0.42
	Mean	26	0	4	0.46
AOA	Best	26	4	0	4.28e-07
	STD	24	0	6	0.41
	Mean	23	0	7	0.37
ALO	Best	26	4	0	1.10e-03
	STD	22	0	8	0.46
	Mean	24	0	6	0.36
WO	Best	26	4	0	2.12e-09
	STD	27	0	3	0.40
	Mean	24	0	6	0.45
IUDE	Best	48	4	5	0.25
	STD	45	0	12	0.83
	Mean	48	0	9	0.54
SASS	Best	42	9	6	0.55
	STD	41	0	16	0.63
	Mean	44	0	13	0.56
ϵMAgES	Best	49	0	8	0.20
	STD	46	0	11	0.47
	Mean	46	0	11	0.46
COLSHADE	Best	43	7	7	0.46
	STD	44	0	13	0.46
	Mean	46	0	11	0.54
IMODE	Best	47	3	7	0.48
	STD	43	0	14	0.42
	Mean	42	0	16	0.54
EBOwithCMAR	Best	41	10	6	0.46
	STD	47	0	10	0.45
	Mean	43	0	14	0.56

Table 35 20D evaluation results

WAO vs.	Criteria	Better	Similar	Worse	*P value*
BBO	Best	30	0	0	4.27e-06
	STD	26	0	4	0.51
	Mean	26	0	4	0.38
HHO	Best	26	4	0	7.22e-05
	STD	26	0	4	0.32
	Mean	24	0	6	0.42
GOA	Best	27	3	0	3.73e-06
	STD	27	0	3	0.30
	Mean	26	0	4	0.22
TSO	Best	25	5	0	8.19e-05
	STD	27	0	3	0.35
	Mean	27	0	3	0.40
MFO	Best	25	5	0	6.88e-04
	STD	27	0	3	0.36
	Mean	25	0	5	0.40
SSA	Best	26	4	0	5.72e-05
	STD	24	0	6	0.50
	Mean	22	0	8	0.41
SCA	Best	25	4	1	0.01
	STD	27	0	3	0.35
	Mean	26	0	4	0.42
AOA	Best	26	4	0	3.25e-04
	STD	24	0	6	0.42
	Mean	23	0	7	0.39
ALO	Best	26	4	0	1.26e-03
	STD	22	0	8	0.44
	Mean	24	0	6	0.36
WO	Best	21	9	0	2.88e-03
	STD	27	0	3	0.41
	Mean	24	0	6	0.43
IUDE	Best	48	4	5	0.25
	STD	45	0	12	0.81
	Mean	48	0	9	0.55
SASS	Best	42	9	6	0.54
	STD	41	0	16	0.62
	Mean	44	0	13	0.54
ϵMAgES	Best	49	0	8	0.19
	STD	43	0	14	0.52
	Mean	46	0	11	0.44
COLSHADE	Best	41	9	7	0.43
	STD	44	0	13	0.43
	Mean	46	0	11	0.54
IMODE	Best	47	3	7	0.44
	STD	43	0	14	0.47
	Mean	42	0	16	0.56
EBOwithCMAR	Best	41	10	6	0.44
	STD	46	0	11	0.47
	Mean	43	0	14	0.54

Table 36 50D evaluation results

WAO vs.	Criteria	Better	Similar	Worse	*P value*
BBO	Best	27	3	0	3.52e-04
	STD	26	0	4	0.56
	Mean	26	0	4	0.41
HHO	Best	26	4	0	4.25e-03
	STD	25	0	5	0.36
	Mean	24	0	6	0.45
GOA	Best	27	3	0	7.03e-04
	STD	27	0	3	0.31
	Mean	26	0	4	0.23
TSO	Best	25	5	0	7.10e-03
	STD	26	0	4	0.37
	Mean	27	0	3	0.41
MFO	Best	25	5	0	2.71e-03
	STD	27	0	3	0.37
	Mean	25	0	5	0.41
SSA	Best	24	6	0	7.70e-04
	STD	24	0	6	0.51
	Mean	22	0	8	0.48
SCA	Best	20	9	1	0.03
	STD	27	0	3	0.41
	Mean	26	0	4	0.47
AOA	Best	26	4	0	6.78e-04
	STD	24	0	6	0.41
	Mean	23	0	7	0.39
ALO	Best	26	4	0	4.21e-02
	STD	22	0	8	0.46
	Mean	24	0	6	0.37
WO	Best	21	9	0	4.82e-02
	STD	27	0	3	0.41
	Mean	24	0	6	0.44
IUDE	Best	48	4	5	0.24
	STD	45	0	12	0.83
	Mean	48	0	9	0.54
SASS	Best	42	9	6	0.55
	STD	41	0	16	0.63
	Mean	44	0	13	0.56
ϵMAgES	Best	49	0	8	0.19
	STD	43	0	14	0.55
	Mean	46	0	11	0.45
COLSHADE	Best	41	9	7	0.44
	STD	44	0	13	0.47
	Mean	46	0	11	0.56
IMODE	Best	47	3	7	0.44
	STD	43	0	14	0.47
	Mean	42	0	16	0.56
EBOwithCMAR	Best	41	10	6	0.44
	STD	46	0	11	0.47
	Mean	43	0	14	0.56

Table 37 Adjustment of Taguchi control parameters in five types, for 10 algorithms compared with WAO

Algorithm								
WAO	Parametrs Types	bestBound	boundNum	Htemp	nVar	H		
	Type 1	2	0.1	0.3	2	10		
	Type 2	5	0.2	0.5	4	20		
	Type 3	10	0.3	0.6	5	30		
	Type 4	20	0.5	0.8	8	40		
	Type 5	30	1.0	1.0	10	50		
BBO	Parametrs Types	Keep	lambdaLower	lambdaUpper	I	E	popSize	dim
	Type 1	1	0.0	1	1	1	10	2
	Type 2	2	0.2	2	2	2	20	4
	Type 3	3	0.4	3	3	3	30	5
	Type 4	4	0.6	4	4	4	40	8
	Type 5	5	0.8	5	5	5	50	10
HHO	Parametrs Types	searchAgents_no	dim	j				
	Type 1	10	2	0				
	Type 2	20	4	1				
	Type 3	30	5	2				
	Type 4	40	8	3				
	Type 5	50	10	4				
GOA	Parametrs Types	cMax	cMin	dim	VOC	popSize		
	Type 1	1	0.00004	2	1	10		
	Type 2	2	0.00001	4	2	20		
	Type 3	3	0.00003	5	3	30		
	Type 4	4	0.00005	8	4	40		
	Type 5	5	0.00002	10	5	50		
TSO	Parametrs Types	searchAgents_no	dim	Elites				
	Type 1	10	2	2				
	Type 2	20	4	3				
	Type 3	30	5	4				
	Type 4	40	8	5				
	Type 5	50	10	6				
MFO	Parametrs Types	searchAgents_no	Boundary_no	dim	Logarithmic spiral			
	Type 1	10	2	2	0.75			
	Type 2	20	3	4	0.85			
	Type 3	30	4	5	0.95			
	Type 4	40	5	8	0.65			
	Type 5	50	6	10	0.55			
SSA	Parametrs Types	searchAgents_no	dim					
	Type 1	10	2					
	Type 2	20	4					
	Type 3	30	5					
	Type 4	40	8					
	Type 5	50	10					

Table 37 (continued)

SCA	Parametrs		dim	a	Elites
	Types	searchAgents_no			
	Type 1	10	2	2	2
	Type 2	20	4	3	3
	Type 3	30	5	4	4
	Type 4	40	8	5	5
	Type 5	50	10	6	6
AOA	Parametrs		dim		
	Types	searchAgents_no			
	Type 1	10	2		
	Type 2	20	4		
	Type 3	30	5		
	Type 4	40	8		
	Type 5	50	10		
ALO	Parametrs	Current_iter		dim	
	Types		searchAgents_no		
	Type 1	2	10	2	
	Type 2	3	20	4	
	Type 3	4	30	5	
	Type 4	5	40	8	
	Type 5	6	50	10	
WO	Parametrs		dim	B	
	Types	searchAgents_no			
	Type 1	10	2	1	
	Type 2	20	4	2	
	Type 3	30	5	3	
	Type 4	40	8	4	
	Type 5	50	10	5	

References

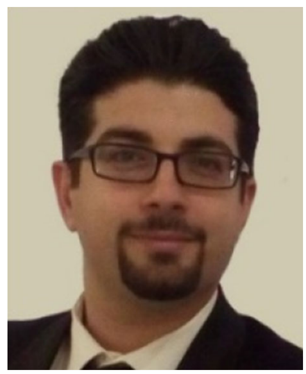
1. Duan QY, Gupta VK, Sorooshian S (1993) Shuffled complex evolution approach for effective and efficient global minimization. J Optim Theory Appl 76(3):501–521
2. Sahai T (2020) Dynamical systems theory and algorithms for NP-hard problems. In: Proceedings of the Workshop on Dynamics, Optimization and Computation held in honor of the 60th birthday of Michael Dellnitz, pp 183–206
3. Woeginger GJ (2003) Exact algorithms for NP-hard problems: a survey. In: Combinatorial optimization—eureka, you shrink! Springer, pp 185–207
4. Dokeroglu T, Sevinc E, Kucukyilmaz T, Cosar A (2019) A survey on new generation metaheuristic algorithms. Comput Ind Eng 137: 106040
5. Mitchell M, Forrest S (1994) Genetic algorithms and artificial life. Artif Life 1(3):267–289
6. Chelouah R, Siarry P (2000) A continuous genetic algorithm designed for the global optimization of multimodal functions. J Heuristics 6(2):191–213
7. Faradonbeh RS, Monjezi M, Armaghani DJ (2016) Genetic programing and non-linear multiple regression techniques to predict backbreak in blasting operation. Eng Comput 32(1):123–133
8. Dorigo M, Birattari M, Stutzle T (2006) Ant colony optimization. IEEE Comput Intell Mag 1(4):28–39
9. Dhiman G, Kumar V (2017) Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications. Adv Eng Softw 114:48–70
10. Burke EK, Gendreau M, Hyde M, Kendall G, Ochoa G, Özcan E, Qu R (2013) Hyper-heuristics: a survey of the state of the art. J Oper Res Soc 64(12):1695–1724
11. Pappa GL, Ochoa G, Hyde MR, Freitas AA, Woodward J, Swan J (2014) Contrasting meta-learning and hyper-heuristic research: the role of evolutionary algorithms. Genet Program Evolvable Mach 15(1):3–35
12. Xavier-Júnior JC, Freitas AA, Ludermir TB, Feitosa-Neto A, Barreto CA (2020) An evolutionary algorithm for automated machine learning focusing on classifier ensembles: an improved algorithm and extended results. Theor Comput Sci 805:1–18
13. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1(1):67–82
14. Hemasian-Etefagh F, Safi-Esfahani F (2019) Dynamic scheduling applying new population grouping of whales meta-heuristic in cloud computing. J Supercomput 75(10):6386–6450
15. Coello CAC (1999) A comprehensive survey of evolutionary-based multiobjective optimization techniques. Knowl Inf Syst 1(3):269–308
16. Hashim FA, Hussain K, Houssein EH, Mabrouk MS, Al-Atabany W (2021) Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems. Appl Intell 51(3): 1531–1551
17. Rajakumar R, Dhavachelvan P, Vengattaraman T (2016) A survey on nature inspired meta-heuristic algorithms with its domain specifications. In: 2016 international conference on communication and electronics systems (ICCES), pp 1–6
18. Parpinelli RS, Lopes HS (2011) New inspirations in swarm intelligence: a survey. Int J Bio-Inspired Comput 3(1):1–16

19. Dhiman G (2019) ESA: a hybrid bio-inspired metaheuristic optimization approach for engineering problems. Eng Comput 37:1–31
20. Qin AK, Huang VL, Suganthan PN (2008) Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE Trans Evol Comput 13(2):398–417
21. Trivedi A, Sanyal K, Verma P, Srinivasan D (2017) A unified differential evolution algorithm for constrained optimization problems. In: 2017 IEEE congress on evolutionary computation (CEC), pp 1231–1238
22. Pereira-Neto A, Unsihuay C, Saavedra OR (2005) Efficient evolutionary strategy optimisation procedure to solve the nonconvex economic dispatch problem with generator constraints. IEE Proc-Gener Transm Distrib 152(5):653–660
23. Zhang J-H, Xu X-H (1999) An efficient evolutionary programming algorithm. Comput Oper Res 26(7):645–663
24. Van Laarhoven PJ, Aarts EH (1987) Simulated annealing. In: Simulated annealing: theory and applications. Springer, pp 7–15
25. Rao RV, Savsani VJ, Vakharia DP (2011) Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. Comput Aided Des 43(3):303–315
26. Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179(13):2232–2248
27. Qais MH, Hasanien HM, Alghuwainem S (2020) Transient search optimization: a new meta-heuristic optimization algorithm. Appl Intell 50(11):3926–3941
28. Kumar A, Das S, Zelinka I (2020) A modified covariance matrix adaptation evolution strategy for real-world constrained optimization problems. In: Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion, pp 11–12
29. Mahdavi M, Fesanghary M, Damangir E (2007) An improved harmony search algorithm for solving optimization problems. Appl Math Comput 188(2):1567–1579
30. Knowles JD, Corne DW (2000) M-PAES: A memetic algorithm for multiobjective optimization. In: Proceedings of the 2000 Congress on evolutionary computation. CEC00 (Cat No 00TH8512), vol 1, pp 325–332
31. Birbil Şİ, Fang S-C (2003) An electromagnetism-like mechanism for global optimization. J Glob Optim 25(3):263–282
32. Kora P, Krishna KSR (2016) Hybrid firefly and particle swarm optimization algorithm for the detection of bundle branch block. Int J Cardiovasc Acad 2(1):44–48
33. Gandomi AH, Yang X-S, Alavi AH (2013) Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng Comput 29(1):17–35
34. Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H (2019) Harris hawks optimization: algorithm and applications. Future Gener Comput Syst 97:849–872
35. Alimoradi M, Azgomi H, Asghari A (2021) Trees social relations optimization algorithm: a new swarm-based metaheuristic technique to solve continuous and discrete optimization problems. Math Comput Simul 194:629–664
36. Rahmati SHA, Zandieh M (2012) A new biogeography-based optimization (BBO) algorithm for the flexible job shop scheduling problem. Int J Adv Manuf Technol 58(9):1115–1129
37. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61
38. Mirjalili S (2015) Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl-Based Syst 89:228–249
39. Mirjalili S (2015) The ant lion optimizer. Adv Eng Softw 83:80–98
40. Mirjalili SZ, Mirjalili S, Saremi S, Faris H, Aljarah I (2018) Grasshopper optimization algorithm for multi-objective optimization problems. Appl Intell 48(4):805–820
41. Yang X-S (2010) A new metaheuristic bat-inspired algorithm. In: Nature inspired cooperative strategies for optimization (NICSO 2010). Springer, pp 65–74
42. Karimkashi S, Kishk AA (2010) Invasive weed optimization and its features in electromagnetics. IEEE Trans Antennas Propag 58(4): 1269–1278
43. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67
44. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, vol 4, pp 1942–1948
45. Saha S, Mukherjee V (2018) A novel quasi-oppositional chaotic antlion optimizer for global optimization. Appl Intell 48(9):2628–2660
46. Wang H, Fan C-C, Tu X (2016) AFSAOCP: a novel artificial fish swarm optimization algorithm aided by ocean current power. Appl Intell 45(4):992–1007
47. Mousavirad SJ, Ebrahimpour-Komleh H (2017) Human mental search: a new population-based metaheuristic optimization algorithm. Appl Intell 47(3):850–887
48. Gurrola-Ramos J, Hernàndez-Aguirre A, Dalmau-Cedeño O (2020) COLSHADE for real-world single-objective constrained optimization problems. In: 2020 IEEE Congress on Evolutionary Computation (CEC), pp 1–8
49. Kumar A, Das S, Zelinka I (2020) A self-adaptive spherical search algorithm for real-world constrained optimization problems. In: Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion, pp 13–14
50. Karaboga D (2010) Artificial bee colony algorithm. scholarpedia 5(3):6915
51. Liu B, Zhou Y (2008) Artificial fish swarm optimization algorithm based on genetic algorithm. Comput Eng Des 22
52. Shah-Hosseini H (2009) The intelligent water drops algorithm: a nature-inspired swarm-based optimization algorithm. Int J Bio-Inspired Comput 1(1–2):71–79
53. Mirjalili S (2016) SCA: a sine cosine algorithm for solving optimization problems. Knowl-Based Syst 96:120–133
54. Timmis J, Hone A, Stibor T, Clark E (2008) Theoretical advances in artificial immune systems. Theor Comput Sci 403(1):11–32
55. Wang G-G, Guo L, Gandomi AH, Hao G-S, Wang H (2014) Chaotic krill herd algorithm. Inf Sci 274:17–34
56. Kim JY, Kim Y, Kim YK (2001) An endosymbiotic evolutionary algorithm for optimization. Appl Intell 15(2):117–130
57. Passino KM (2010) Bacterial foraging optimization. Int J Swarm Intell Res IJSIR 1(1):1–16
58. Greensmith J (2007) The dendritic cell algorithm. PhD Thesis, Citeseer
59. Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM (2017) Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv Eng Softw 114:163–191
60. Chen B, Ivanov I, Klein ML, Parrinello M (2003) Hydrogen bonding in water. Phys Rev Lett 91(21):215503
61. Ball P (2015) H2O: a biography of water. Hachette UK
62. Rohr K, Linder F (1976) Vibrational excitation of polar molecules by electron impact. I. Threshold resonance in HF and HCl. J Phys B At Mol Phys 9(14):2521
63. Aveyard R, Binks BP, Fletcher PDI, Lu JR (1990) The resolution of water-in-crude oil emulsions by the addition of low molar mass demulsifiers. J Colloid Interface Sci 139(1):128–138
64. Ladenstein R, Antranikian G (1998) Proteins from hyperthermophiles: stability and enzymatic catalysis close to the boiling point of water. Biotechnol Extrem:37–85
65. Kumar A, Wu G, Ali MZ, Mallipeddi R, Suganthan PN, Das S (2020) A test-suite of non-convex constrained optimization problems from the real-world and some baseline results. Swarm Evol Comput 56:100693
66. Digalakis JG, Margaritis KG (2002) An experimental study of benchmarking functions for genetic algorithms. Int J Comput Math 79(4):403–416

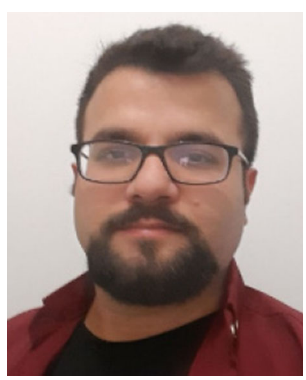
67. Lin M-H, Tsai J-F, Hu N-Z, Chang S-C (2013) Design optimization of a speed reducer using deterministic techniques. Math Probl Eng 2013
68. Coello CAC (2002) Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. Comput Methods Appl Mech Eng 191(11–12):1245–1287
69. Masehian E, Sedighizadeh D (2010) A multi-objective PSO-based algorithm for robot path planning. In: 2010 IEEE International Conference on Industrial Technology, pp 465–470
70. You S, Diao M, Gao L (2019) Implementation of a combinatorial-optimisation-based threat evaluation and jamming allocation system. IET Radar Sonar Navig 13(10):1636–1645
71. Sallam KM, Elsayed SM, Chakrabortty RK, Ryan MJ (2020) Improved multi-operator differential evolution algorithm for solving unconstrained problems. In: 2020 IEEE Congress on Evolutionary Computation (CEC), pp 1–8
72. Kumar A, Misra RK, Singh D (2017) Improving the local search capability of effective butterfly optimizer using covariance matrix adapted retreat phase. In: 2017 IEEE congress on evolutionary computation (CEC), pp 1835–1842
73. Ghani JA, Choudhury IA, Hassan HH (2004) Application of Taguchi method in the optimization of end milling parameters. J Mater Process Technol 145(1):84–92
74. Huang H-C (2015) A Taguchi-based heterogeneous parallel metaheuristic ACO-PSO and its FPGA realization to optimal polar-space locomotion control of four-wheeled redundant mobile robots. IEEE Trans Ind Inform 11(4):915–922

Arman Daliri received the B.S Degree in Software Engineering from the Computer Engineering Faculty, Islamic Azad University, Hashtgerd, Iran. He received M.S Degree in Artificial intelligence and robotics from the Computer Engineering Faculty, Shafagh Institute of Higher Education, Tonekabon, Iran. He is now an Independent researcher that works on machine learning and artificial intelligence in the field of health care. His research interests include Optimization algorithms, evolutionary computing, and machine learning-based healthcare.

Ali Asghari received the B.S and M.S degrees in computer engineering from the Computer Engineering Faculty, Amirkabir University of Technology (Polytechnic of Tehran), Tehran, Iran. He received Ph.D degree in computer Architecture from Computer Engineering Faculty, Islamic Azad University, Semnan Branch, Semnan, Iran. He is now the assistant professor for the Department of Computer Engineering, Shafagh Institute of Higher Education, Tonekabon, Iran. His research interests include cloud computing, evolutionary computing, wireless networks, Metaheuristic Algorithms, and machine learning.



Hossein Azgomi received the B.S. degree in Software engineering from Computer Engineering Faculty, Islamic Azad University, Lahijan Branch, Iran, in 2010. He received M.S. degree in Software engineering from Computer Faculty, Islamic Azad University, Shabestar Branch, Shabestar, Iran, in 2013. He received Ph.D. degree in software engineering from the Computer Faculty, Islamic Azad University, Semnan Branch, Semnan, Iran, in 2017. He is the assistant professor of Engineering Faculty, Rasht Branch, Islamic Azad University, Rasht, Iran. Also, he is a member of the Young Researchers and Elite Club of Rasht Branch. He has more than 20 journal and conference publications describing his research area. His research fields include Data Mining, Data warehousing, Game Theory, and Metaheuristic Algorithms.



Mahmoud Alimoradi received the B.S Degree in Software Engineering from the Computer Engineering Faculty, Islamic Azad University, Lahijan, Iran. He received M.S Degree in Artificial intelligence and robotics from the Computer Engineering Faculty, Shafagh Institute of Higher Education, Tonekabon, Iran. He is now an independent researcher that works on machine learning and artificial intelligence. His research interests include Optimization, Machine learning, Data problems, Planning, and Time series analysis.