Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 376 (2021) 113609
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

The Arithmetic Optimization Algorithm

Laith Abualigah[a,*], Ali Diabat[b,c], Seyedali Mirjalili[d,e], Mohamed Abd Elaziz[f,g], Amir H. Gandomi[h,*]

[a] *Faculty of Computer Sciences and Informatics, Amman Arab University, 11953, Amman, Jordan*
[b] *Division of Engineering, New York University Abu Dhabi, Saadiyat Island, 129188, Abu Dhabi, United Arab Emirates*
[c] *Department of Civil and Urban Engineering, Tandon School of Engineering, New York University, Brooklyn, NY 11201, USA*
[d] *Centre for Artificial Intelligence Research and Optimisation, Torrens University Australia, Brisbane, Australia*
[e] *YFL (Yonsei Frontier Lab), Yonsei University, Seoul, Korea*
[f] *Department of Mathematics, Faculty of Science, Zagazig University, Egypt*
[g] *School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China*
[h] *Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW 2007, Australia*

Received 15 April 2020; received in revised form 26 October 2020; accepted 24 November 2020
Available online 11 January 2021

Abstract

This work proposes a new meta-heuristic method called Arithmetic Optimization Algorithm (AOA) that utilizes the distribution behavior of the main arithmetic operators in mathematics including (Multiplication (M), Division (D), Subtraction (S), and Addition (A)). AOA is mathematically modeled and implemented to perform the optimization processes in a wide range of search spaces. The performance of AOA is checked on twenty-nine benchmark functions and several real-world engineering design problems to showcase its applicability. The analysis of performance, convergence behaviors, and the computational complexity of the proposed AOA have been evaluated by different scenarios. Experimental results show that the AOA provides very promising results in solving challenging optimization problems compared with eleven other well-known optimization algorithms. Source codes of AOA are publicly available at http://www.mathworks.com/matlabcentral/fileexchange/84742 and https://seyedalimirjalili.com/projects.

Keywords: Arithmetic Optimization Algorithm; AOA; Algorithm; Optimization; Meta-heuristics; Heuristic; Artificial Intelligence; Grey Wolf Optimizer; Whale Optimization Algorithm; Genetic Algorithm; Particle Swarm Optimization; Benchmark; Computational Intelligence

1. Introduction

In recent decades, the ever-increasing complexity and difficulty of real-world problems resulted in the need for more reliable optimization techniques, especially meta-heuristic optimization algorithms. These techniques are mostly stochastic and estimate optimal solutions for different optimization problems [1,2]. Such optimization algorithms supersede conventional optimization algorithms due to gradient-free mechanisms and high local optimal

* Corresponding author.
E-mail addresses: Aligah.2020@gmail.com (L. Abualigah), Diabat@nyu.edu (A. Diabat), ali.mirjalili@laureate.edu.au (S. Mirjalili), sabd_el_aziz_m@yahoo.com (M. Abd Elaziz), Gandomi@uts.edu.au (A.H. Gandomi).
URL: https://seyedalimirjalili.com/ (S. Mirjalili).

https://doi.org/10.1016/j.cma.2020.113609

avoidance capability [3,4]. The optimization process finds the optimal decision variables of a function or a problem by minimizing or maximizing its objective function. Generally speaking, real-world and optimization problems have non-linear restrictions, complex, high computational time, non-convex, and wide search spaces [5,6], which make them challenging to solve.

Meta-heuristic optimization algorithms have two important search strategies: (1) exploration/diversification and (2) exploitation/intensification [7,8]. Exploration is the capability to explore the search space globally. This ability is related to the avoidance of local optima and resolving local optima entrapment. On the contrary, exploitation is the capability to explore nearby promising solutions to improve their quality locally [9]. Excellent performance of an algorithm requires a proper balance between these two strategies [10–12]. All population-based algorithms use these features but with different operators and mechanisms.

One popular classification of meta-heuristics is based on the inspiration of evolutionary algorithms, swarm intelligence algorithms, physics-based methods, and human-based methods [13,14]. Evolutionary algorithms simulate habits in natural evolution and use operators motivated by biology behaviors like crossover and mutation. A conventional evolutionary algorithm is the Genetic Algorithm (GA), which is motivated by Darwinian evolutionary ideas. Conventional methods of this group include Evolutionary Programming [15], Differential Evolution [16], and Evolution Strategy [17].

Swarm intelligence algorithms are another group of meta-heuristics, which simulates the behavior of animals in movement or hunting groups [18,19]. The main characteristic of this group is the sharing of organism information of all animals through the optimization course. Conventional methods of this group include Krill Herd Algorithm [20], Salp Swarm Algorithm [20], Symbiotic Organisms Search [21], Sine Cosine Algorithm [22], and Dolphin Echolocation [23].

Physics-based methods are another group of optimization algorithms. This group originates from physical laws in real-life and typically describes the communication of search solutions based on controlling rules ingrained in physical methods. The most commonly utilized algorithms in this group are Simulated Annealing [24], Gravitational Search Algorithm [25], Multi-verse Optimizer [26], and Charged System Search [27].

The final group of optimization is human-based methods, motivated by human co-operations and human behavior in communities. One of the most used algorithms in this group is the Imperialist Competitive Algorithm [28], which is motivated by the human socio-political growth practice. Another algorithm in this group is the Teaching–Learning-Based Optimization Algorithm [29].

The theoretical studies published in the literature can be classified into three sections: modifying the current algorithms, hybridizing various algorithms, and proposing novel algorithms. All these three areas are very active with a large body of algorithms and applications. The reason why researchers do not use a single algorithm is because there is no optimization algorithm to solve all optimization problems according to the No Free Lunch theorem [30]. Therefore, we need to modify the existing algorithms or propose new ones to be able to better solve the current problems or provide solutions for new problems. This motivates our attempt to propose a new optimization algorithm called Arithmetic Optimization Algorithm (AOA). The remainder of the paper is structured as follows.

The particular implementation of the proposed AOA is illustrated in Section 2. The results of the proposed AOA in solving various benchmark test functions and real-world problems are given in Section 3. Finally, the conclusion and potential future research directions are presented in Section 4.

2. The arithmetic optimization algorithm (AOA)

Generally, population-based algorithms begin their improvement processes (optimization process) with a set of candidate solutions generated randomly. This generated set of solutions is improved by a set of optimization rules incrementally and evaluated by a specific objective function iteratively; that is the essence of the optimization methods. Since population-based algorithms seek to find the optimal solution of optimization problems stochastically, getting a solution in a single run is not guaranteed. Nevertheless, the probability of getting the global optimal solution, for the given problem, is increased by a sufficient number of random solutions and optimization iterations [22].

Despite the differences between meta-heuristic algorithms in the area of population-based optimization methods, the optimization process consists of two main phases: exploration versus exploitation. The former refers to extensive coverage of search space using search agents of an algorithm to avoid local solutions. The latter is the improvement accuracy of obtained solutions during the exploration phase.

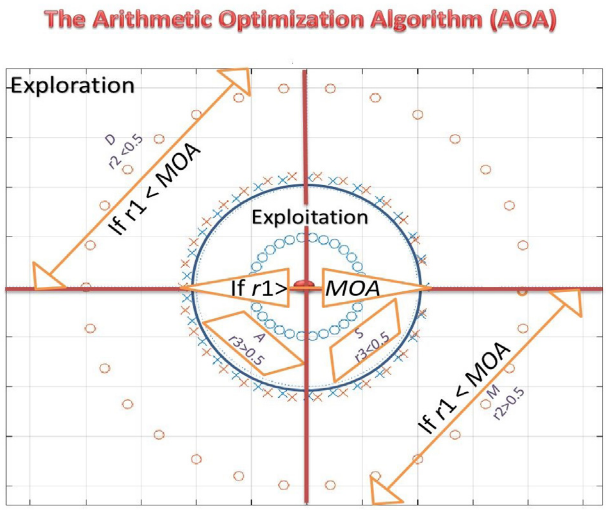


Fig. 1. The search phases of the AOA.

In the following sub-sections, we represent the exploration (diversification) and exploitation (intensification) mechanisms in the proposed AOA, which is achieved by the Arithmetic operators in math (i.e., 1) Multiplication (M " $\times$ "), (2) Division (D " $\div$ "), (3) Subtraction (S " $-$ "), and (4) Addition (A " $+$ ")). Fig. 1 shows the exploratory and exploitative mechanisms in AOA. This algorithm is a population-based meta-heuristic capable of solving optimization problems without calculating their derivitives.

2.1. Inspiration

Arithmetic is a fundamental component of number theory, and it is one of the important parts of modern mathematics, along with geometry, algebra, and analysis. Arithmetic operators (i.e., Multiplication, Division, Subtraction, and Addition) are the traditional calculation measures used usually to study the numbers [31]. We use these simple operators as a mathematical optimization to determine the best element subjected to specific criteria from some set of candidate alternatives (solutions). Optimization problems occur in all quantitative disciplines from engineering, economics, and computer sciences to operations research and industry, and the improvement of solution techniques has attracted the interest of mathematics for eras.

The main inspiration of the proposed AOA arises from the use of Arithmetic operators in solving the Arithmetic problems. In the following subsections, the behavior of Arithmetic operators (i.e., Multiplication, Division, Subtraction, and Addition) and their influence in the proposed algorithm will be discussed. Fig. 2 shows the hierarchy of Arithmetic operators and its dominance from the outside to the inside. AOA is then proposed based on the mathematical model.

2.2. Initialization phase

In AOA, the optimization process begins with a set of candidate solutions (X) as shown in Matrix (1), which is generated randomly, and the best candidate solution in each iteration is considered as the best-obtained solution or nearly the optimum so far.

$$X = \begin{bmatrix} x_{1,1} & \cdots & \cdots & x_{1,j} & x_{1,n-1} & x_{1,n} \\ x_{2,1} & \cdots & \cdots & x_{2,j} & \cdots & x_{2,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N-1,1} & \cdots & \cdots & x_{N-1,j} & \cdots & x_{N-1,n} \\ x_{N,1} & \cdots & \cdots & x_{N,j} & x_{N,n-1} & x_{N,n} \end{bmatrix} \tag{1}$$

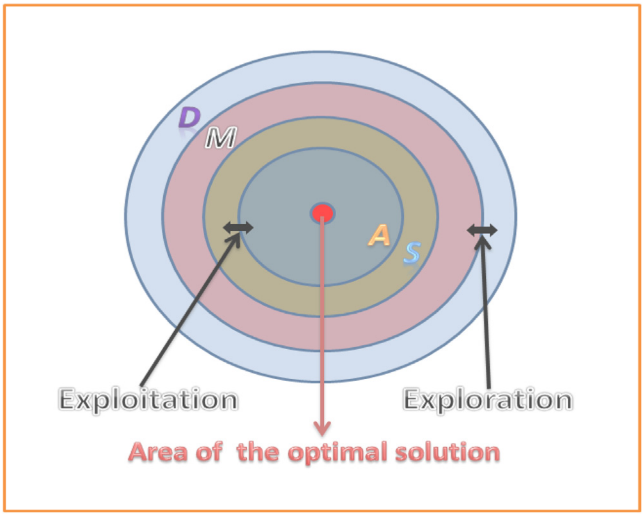


Fig. 2. Hierarchy of Arithmetic operators (dominance decreases from top-down).

Before the AOA starts working, it should select the search phase (i.e., exploration or exploitation). So, Math Optimizer Accelerated (MOA) function is a coefficient calculated by Eq. (2) used in the following search phases.

$$MOA(C\_Iter) = Min + C\_Iter \times \left(\frac{Max - Min}{M\_Iter}\right) \tag{2}$$

where MOA(C_Iter) denotes the function value at the tth iteration, which is calculated by Eq. (2). C_Iter denotes the current iteration, which is between 1 and the maximum number of iterations (M_Iter). Min and Max denote the minimum and maximum values of the accelerated function, respectively.

2.3. Exploration phase

In this section, the exploratory behavior of AOA is introduced. According to the Arithmetic operators, the mathematical calculations using either Division (D) operator or even Multiplication (M) operator got high-distributed values or decisions (refer to various reigns) which commit to the exploration search mechanism. However, these operators (D and M) cannot easily approach the target due to their high dispersion, unlike other operators (S and A). Fig. 3 shows the influence and behavior of Arithmetic operators in mathematical calculations. A function is employed based on using four mathematical operations to show the effect of the different operators' distribution values. Hence, the exploration search detects the near-optimal solution that may be deduced after several endeavours (iterations). In addition, the exploration operators (D and M) were operated at this stage of optimization to support the other stage (exploitation) in the search process through enhanced communication between them.

The exploration operators of AOA explore the search area randomly on several regions and approach to find a better solution based on two main search strategies (Division (D) search strategy and Multiplication search strategy), which are modeled in Eq. (3). This phase of searching (exploration search by executing D or M, see Fig. 4) is conditioned by the Math Optimizer accelerated (MOA) function (see Eq. (2)) for the condition of $r1 >$ MOA ($r1$ is a random number). Fig. 4 shows how the used operators converge toward the optimal area. The first operator (D), in this phase (first rule in Eq. (3)), is conditioned by $r2 < 0.5$ and the other operator (M) will be neglected until this operator finishes its current task. Otherwise, the second operator (M) will be engaged to perform the current task instead of the D ($r2$ is a random number). Note, a stochastic scaling coefficient is considered for the element to produce more diversification courses and explore diverse regions of the search space. We employed the simplest rule, which is able to simulate the behaviors of Arithmetic operators. In this paper, the following position updating equations are proposed for the exploration parts:

$$x_{i,j}(C\_Iter+1) = \begin{cases} best(x_j) \div (MOP + \epsilon) \times ((UB_j - LB_j) \times \mu + LB_j), & r2 < 0.5 \\ best(x_j) \times MOP \times ((UB_j - LB_j) \times \mu + LB_j), & otherwise \end{cases} \tag{3}$$

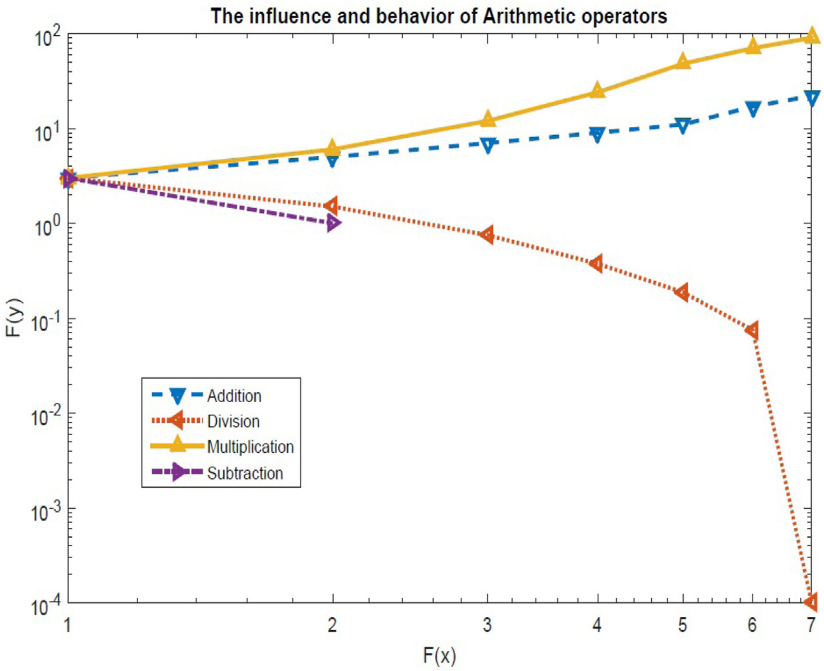


Fig. 3. The influence and behavior of the four math operators (A, D, M, and S) in solving mathematical calculations.

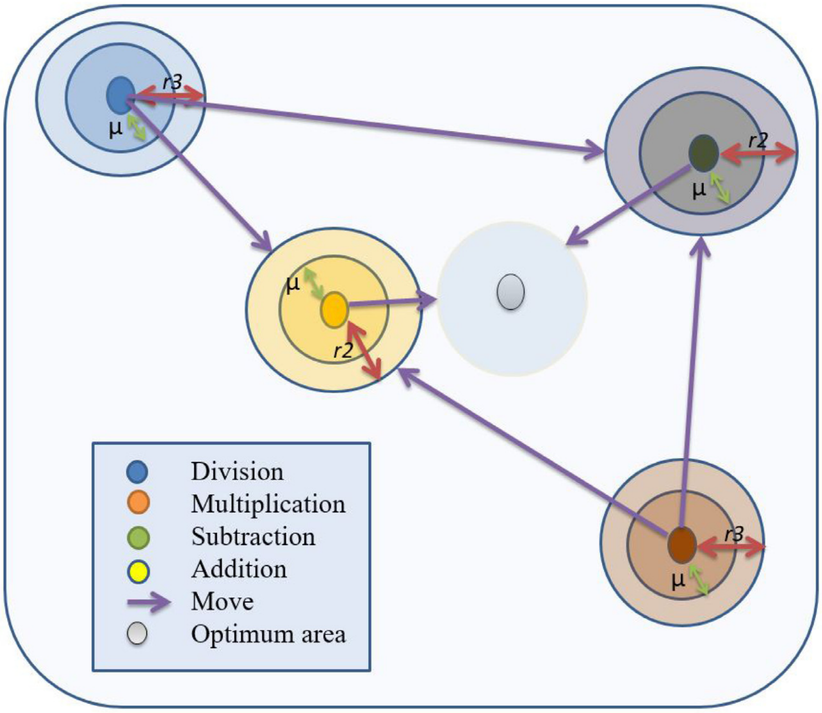


Fig. 4. Model of updating the position of math operators in AOA toward the optimum area.

where x_i(C_Iter+1) denotes the ith solution in the next iteration, $x_{i,j}$(C_Iter) denotes the jth position of the ith solution at the current iteration, and $best(x_j)$ is the jth position in the best-obtained solution so far. ϵ is a small integer number, UB_j and LB_j denote to the upper bound value and lower bound value of the jth position, respectively. μ is a control parameter to adjust the search process, which is fixed equal to 0.5 according to the experiments of this paper.

$$MOP(C\_Iter) = 1 - \frac{C\_Iter^{1/\alpha}}{M\_Iter^{1/\alpha}} \tag{4}$$

where Math Optimizer probability (MOP) is a coefficient, MOP(C_Iter) denotes the function value at the tth iteration, and C_Iter denotes the current iteration and (M_Iter) denotes the maximum number of iterations. α is a sensitive parameter and defines the exploitation accuracy over the iterations, which is fixed equal to 5 according to the experiments of this paper.

2.4. Exploitation phase

In this section, the exploitation strategy of AOA is introduced. According to the Arithmetic operators, the mathematical calculations using either Subtraction (S) or Addition (A) got high-dense results which refer to the exploitation search mechanism. However, these operators (S and A) can easily approach the target due to their low dispersion, unlike other operators, as shown in Fig. 3. Hence, the exploitation search detects the near-optimal solution that may be deduced after several endeavours (iterations). In addition, the exploitation operators (S and A) were operated at this stage of the optimization to support the exploitation stage through enhanced communication between them.

This phase of searching (exploitation search by executing S or A) is conditioned by the MOA function value for the condition of $r1$ is not greater than the current MOA(C_Iter) value (see Eq. (2)). In AOA, the exploitation operators (Subtraction (S) and Addition (A)) of AOA explore the search area deeply on several dense regions and approach to find a better solution based on two main search strategies (i.e., Subtraction (S) search strategy and Addition (A) search strategy), which are modeled in Eq. (5).

$$x_{i,j}(C\_Iter+1) = \begin{cases} best(x_j) - MOP \times ((UB_j - LB_j) \times \mu + LB_j), & r3 < 0.5 \\ best(x_j) + MOP \times ((UB_j - LB_j) \times \mu + LB_j), & otherwise \end{cases} \tag{5}$$

This phase exploits the search space by conducting a deep search, which is very clear in Fig. 3. The first operator (S), in this phase (first rule in Eq. (5)), is conditioned by $r3$ <0.5 and the other operator (A) will be neglected until this operator finishes its current task. Otherwise, the second operator (A) will be engaged to perform the current task instead of the S. These procedures in this phase are similar to the partitions of the previous phase. However, exploitation search operators (S and A) often attempt to avoid getting stuck in the local search area. This procedure assists the exploration search strategies in finding the optimal solution and keeping the diversity of the candidate solutions. We carefully designed μ parameters to produce a stochastic value at each iteration to maintain exploration not only during first iterations but also last iterations. This part of searching is very helpful in the situation of local optima stagnation, particularly in the last iterations.

Fig. 4 explains how a search solution updates its variables (positions) according to D, M, S, and A in a 2-Dimensional search space. It can be seen that the final-obtained position can be in a stochastic position within a range which is determined by the positions of D, M, S, and A in the search scope. In other concepts, D, M, S, and A estimate the position of the near-optimal solution, and other solutions update their positions stochastically around the area of the near-optimal solution.

2.5. Pseudo-code of the arithmetic optimization algorithm (AOA)

To recap, in AOA, the optimization process begins with generating a random set of candidate solutions (population). During the trajectory of repetition, D, M, S, and A estimate the feasible positions of the near-optimal solution. Each solution renews its positions from the best-obtained solution. To emphasize exploration and exploitation, the parameter MOA is increased linearly from 0.2 to 0.9. Candidate solutions seek to diverge from the near-optimal solution when $r1 > MOA$ and converge towards the near-optimal solution when $r1 < MOA$.

Eventually, the AOA algorithm is stopped by reaching the satisfaction of the end criterion. The Pseudo-code of the proposed AOA is described in Algorithm 1. The intuitive and detailed process of AOA is shown in Fig. 5.

Algorithm 1 Pseudo-code of the AOA algorithm

```
Initialize the Arithmetic Optimization Algorithm parameters α, μ.
Initialize the solutions' positions randomly. (Solutions: i=1, ..., N.)
while (C_Iter < M_Iter) do
  Calculate the Fitness Function (FF) for the given solutions
  Find the best solution (Determined best so far).
  Update the MOA value using Eq. (2).
  Update the MOP value using Eq. (4).
  for (i=1 to Solutions) do
    for (j=1 to Positions) do
      Generate a random values between [0, 1] (r1, r2, and r3)
      if r1 >MOA then
        Exploration phase
        if r2 >0.5 then
          (1) Apply the Division math operator (D "÷").
          Update the ith solutions' positions using the first rule in Eq. (3).
        else
          (2) Apply the Multiplication math operator (M "×").
          Update the ith solutions' positions using the second rule in Eq. (3).
        end if
      else
        Exploitation phase
        if r3 >0.5 then
          (1) Apply the Subtraction math operator (S "−").
          Update the ith solutions' positions using the first rule in Eq. (5).
        else
          (2) Apply the Addition math operator (A "+").
          Update the ith solutions' positions using the second rule in Eq. (5).
        end if
      end if
    end for
  end for
  C_Iter=C_Iter+1
end while
Return the best solution (x).
```

2.6. The computational complexity of AOA

Note that the computational complexity of the proposed AOA essentially relies on three factors: initialization processes, fitness function evaluation, and updating of solutions. The complexity of the initialization process is of $O(N)$ where N shows the population size. The complexity of the fitness function is dependent on the problem, so we do not discuss it here. Finally, the complexity of updating solutions is of $O(M \times N) + O(M \times N \times L)$ where M indicates iterations and L is the number of parameters in the problem (dimension). Therefore, the computational complexity of the proposed AOA is of $O(N \times (ML + 1))$. In the next section, various benchmark test functions and real optimization problems are used to validate and confirm the performance of the proposed AOA in addressing optimization problems.

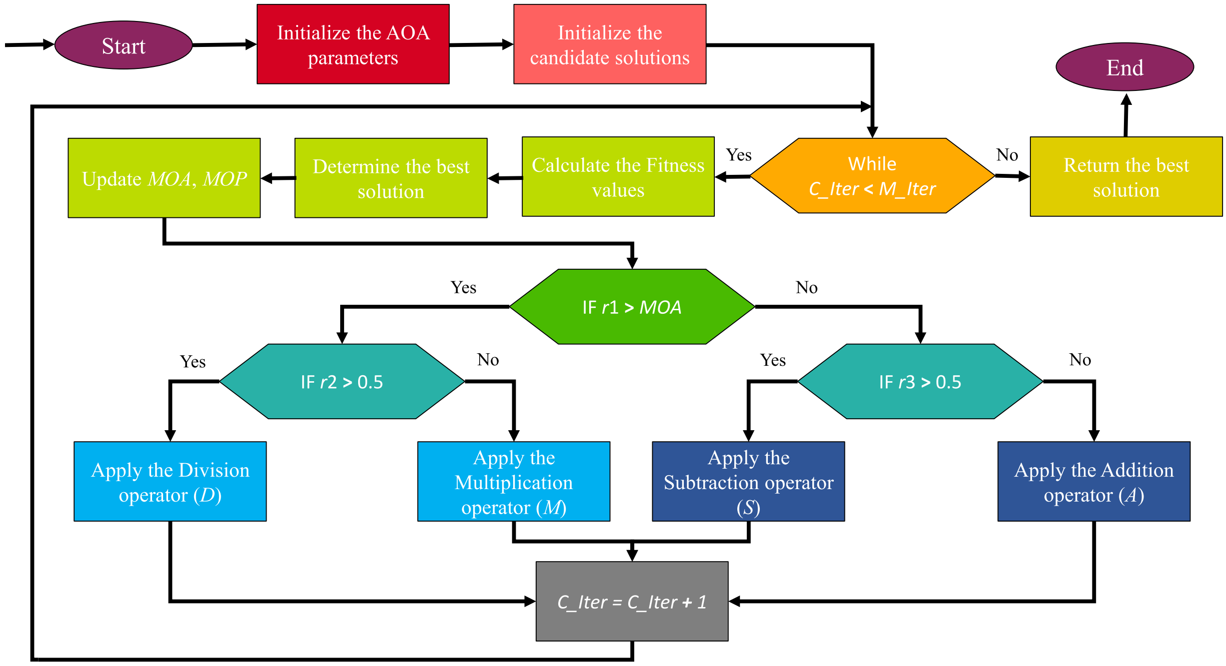


Fig. 5. Flowchart of the proposed AOA.

3. Experimental results and discussions

In this section, the performance of the proposed AOA algorithm is tested on 23 test functions and five engineering design problems. The results are compared with the following algorithms:

- Genetic Algorithm (GA) [32]
- Particle Swarm Optimization (PSO) [33]
- Biogeography-based Optimization (BBO) [34]
- Flower Pollination Algorithm (FPA) [35]
- Grey Wolf Optimizer (GWO) [36]
- Bat Algorithm (BAT) [37]
- Firefly Algorithm (FA) [38]
- Cuckoo Search Algorithm (CS) [39]
- Moth-Flame Optimization (MFO) [40]
- Gravitational Search Algorithm (GSA) [25]
- Differential Evolution (DE) [16].

To achieve a fair comparison, the considered algorithms have implemented using the same number of iterations and population size of GEO 500, 30, respectively, so the number of function evaluations is 15 000. The values used for the main controlling parameters of the comparative algorithms can be seen in Table 1.

The test functions are presented in Tables 2–5 and Figs. 6–9.

The algorithms are compared using mean, standard deviation, Friedman ranking (Rank) test, and Wilcoxon signed-rank test.

3.1. Results comparisons using benchmark test functions

At the beginning of this section, we test the impact of changing the value of the parameters of AOA on its performance. Different scenarios are taken based on the parameters' values (μ and α) of AOA. These parameters are assessed at one value from 0.1, 0.5, and 0.9; therefore, we have nine scenarios (as in Table 6). Table 7 represents

Table 1
Parameter values for the comparative algorithms.

Algorithm	Parameter	Value
PSO	Topology	Fully connected
	Cognitive and social constant	(C1, C2) 2, 2
	Inertia weight	Linear reduction from 0.9 to 0.1
	Velocity limit	10% of dimension range
CS	p_a	0.25
BBO	Probability of modifying a habitat	1
	Probability limits of immigrations	[0, 1]
	Size of each step	1
	I and E	1
	Probability of mutation	0.005
DE	Scaling factor	0.5
	Crossover probability	0.5
GSA	Alpha, G0, Rnorm, Rpower	20, 100, 2, 1
FA	α	0.5
	β	0.2
	γ	1
GA	Type	Real coded
	Selection	Roulette wheel (Proportionate)
	Crossover	Whole arithmetic (Probability = 0.8, α = [−0.5, 1.5])
	Mutation	Gaussian (Probability = 0.05)
MFO	a	[−2 −1]
	b	1
GWO	Convergence parameter (a)	Linear reduction from 2 to 0
BAT	Q_{min}	0
	Q_{max}	2
	A	0.5
	r	0.5
FPA	Probability switch p	0.8
AOA	α	5
	μ	0.5

Table 2
Unimodal test functions.

Function	Description	Dimensions	Range	f_{min}
F1	$f(x) = \sum_{i=1}^{n} x_i^2$	30, 100, 500, 1000	[−100,100]	0
F2	$f(x) = \sum_{i=0}^{n} \lvert x_i\rvert + \prod_{i=0}^{n} \lvert x_i\rvert$	30, 100, 500, 1000	[−10,10]	0
F3	$f(x) = \sum_{i=1}^{d} (\sum_{j=1}^{i} x_j)^2$	30, 100, 500, 1000	[−100,100]	0
F4	$f(x) = max_i\{\lvert x_i\rvert, 1 \le i \le n\}$	30, 100, 500, 1000	[−100,100]	0
F5	$f(x) = \sum_{i=1}^{n-1} [100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2]$	30, 100, 500, 1000	[−30,30]	0
F6	$f(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	30, 100, 500, 1000	[−100,100]	0
F7	$f(x) = \sum_{i=0}^{n} i x_i^4 + \text{random}[0, 1)$	30, 100, 500, 1000	[−128,128]	0

the statistical results achieved at each scenario among the used thirteen benchmark functions. From these results, it can be seen that the fifth scenario (i.e., $\mu = 0.5$ and $\alpha = 5$), among all the tested functions, has better results; it is followed by the sixth and fourth scenarios that allocated the second and third rank, respectively.

The performance of the proposed AOA algorithm is tested in terms of the impact of dimensions, as shown in Table 8. This test is a standard test used in the literature of optimization benchmark test functions, which can show the effects of dimensions on the performance of the AOA to prove its ability not only for low-dimensional problems but also for high-dimensional problems.

Furthermore, Table 8 shows how a population-based algorithm can maintain its searching merits in high-dimensional problems. In this part of the experiments, the proposed AOA is employed to address the scalable Unimodal and Multimodal test functions (F1–F13) with various dimension spaces (30, 100, 500, and 1000). The

Table 3
Multimodal test functions.

Function	Description	Dimensions	Range	f_{min}
F8	$f(x)=\sum_{i=1}^{n}(-x_i sin(\sqrt{\lvert x_i \rvert}))$	30, 100, 500, 1000	[−500,500]	−418.9829 ×n
F9	$f(x)=\sum_{i=1}^{n}[x_i^2-10cos(2\pi x_i)+10]$	30, 100, 500, 1000	[−5.12,5.12]	0
F10	$f(x)=-20exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2})-exp(\frac{1}{n}\sum_{i=1}^{n}cos(2\pi x_i))+20+e$	30, 100, 500, 1000	[−32,32]	0
F11	$f(x)=1+\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}cos(\frac{x_i}{\sqrt{i}})$	30, 100, 500, 1000	[−600,600]	0
F12	$f(x)=\frac{\pi}{n}\{10sin(\pi y_1)\}+\sum_{i=1}^{n-1}(y_i-1)^2[1+10sin^2(\pi y_{i+1})+\sum_{i=1}^{n}u(x_i,10,100,4)]$, where $y_i=1+\frac{x_i+1}{4}, u(x_i,a,k,m)\begin{cases}K(x_i-a)^m & \text{if } x_i>a\\ 0 & -a\le x_i\ge a\\ K(-x_i-a)^m & -a\le x_i\end{cases}$	30, 100, 500, 1000	[−50,50]	0
F13	$f(x)=0.1(sin^2(3\pi x_1)+\sum_{i=1}^{n}(x_i-1)^2[1+sin^2(3\pi x_i+1)]+(x_n-1)^2 1+sin^2(2\pi x_n))+\sum_{i=1}^{n}u(x_i,5,100,4)$	30, 100, 500, 1000	[−50,50]	0

Table 4
Fixed-dimension multimodal test functions.

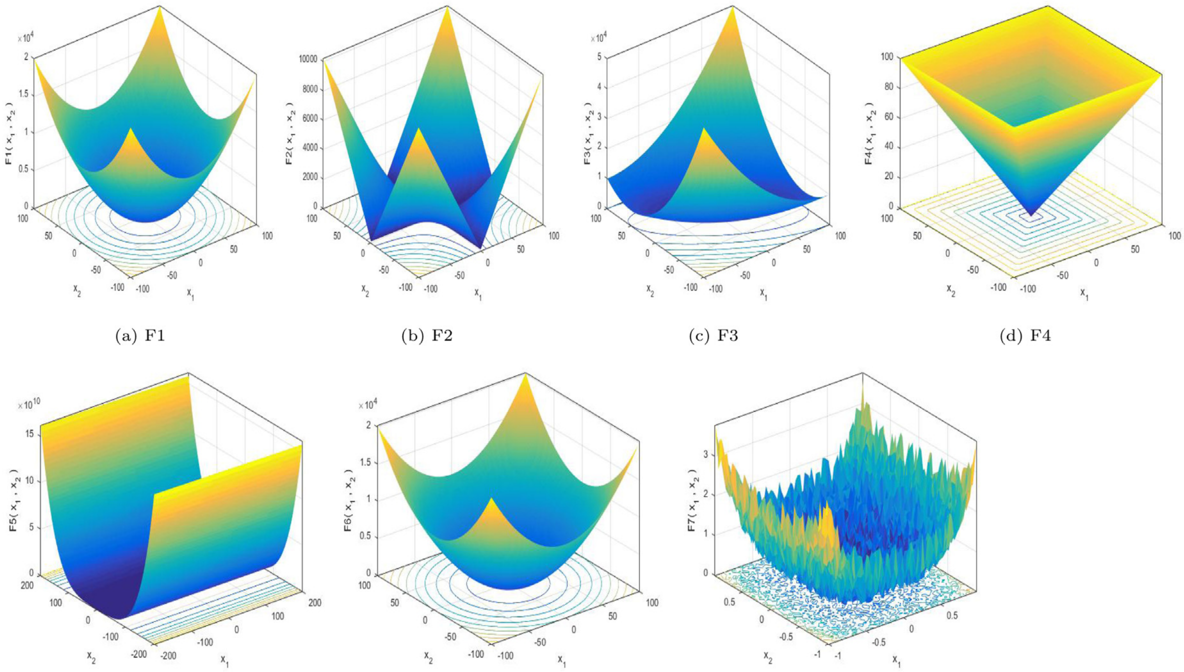
Function	Description	Dimensions	Range	f_{min}
F14	$f(x)=\left(\frac{1}{500}+\sum_{j=1}^{25}\frac{1}{j+\sum_{i=1}^{2}}(x_i-a_{ij})\right)^{-1}$	2	[−65,65]	1
F15	$f(x)=\sum_{i=1}^{11}\left[a_i-\frac{x_1(b_i^2+b_ix_2)}{b_i^2+b_ix_3+x_4}\right]^2$	4	[−5,5]	0.00030
F16	$f(x)=4x_1^2-2.1x_1^4+\frac{1}{3}x_1^6+x_1x_2-4x_2^2+4x_2^4$	2	[−5,5]	−1.0316
F17	f(x)=$\left(x_2-\frac{5.1}{4\pi^2}x_1^2+\frac{5}{\pi}x_1-6\right)^2+10(1-\frac{1}{8\pi})cos\ x_1+10$	2	[−5,5]	0.398
F18	$f(x)=\left[1+(x_1+x_2+1)^2(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2)\right]\times\left[30+(2x_1-3x_2)^2\times(18-32x_i+12x_1^2+48x_2-36x_1x_2+27x_2^2)\right]$	2	[−2,2]	3
F19	$f(x)=-\sum_{i=1}^{4}c_i exp\left(-\sum_{i=1}^{3}a_{ij}(x_j-p_{ij})^2\right)$	3	[−1,2]	−3.86
F20	$f(x)=-\sum_{i=1}^{4}c_i exp\left(-\sum_{i=1}^{6}a_{ij}(x_j-p_{ij})^2\right)$	6	[0,1]	−.32
F21	$f(x)=-\sum_{i=1}^{5}\left[(X-a_i)(X-a_i)^T+c_i\right]^{-1}$	4	[0,1]	−10.1532
F22	$f(x)=-\sum_{i=1}^{7}\left[(X-a_i)(X-a_i)^T+c_i\right]^{-1}$	4	[0,1]	−10.4028
F23	$f(x)=-\sum_{i=1}^{10}\left[(X-a_i)(X-a_i)^T+c_i\right]^{-1}$	4	[0,1]	−10.5363

Table 5
Hybrid composition functions F24–F29 (M: Multimodal, R: Rotated, N: Non-Separable, S: Scalable, D: Dimension).

F(CEC5-F)	Description	Properties	Dimension	Range
F24(C16)	Rotated Hybrid Composition Function	M, R, N, S	30	$[-5,5]^D$
F25(C18)	Rotated Hybrid Composition Function	M, R, N, S	30	$[-5,5]^D$
F26(C19)	Rotated Hybrid Composition Function with narrow basin global optimum	M, N, S	30	$[-5,5]^D$
F27(C20)	Rotated Hybrid Composition Function with Global Optimum on the Bounds	M, N, S	30	$[-5,5]^D$
F28(C21)	Rotated Hybrid Composition Function	M, R, N, S	30	$[-5,5]^D$
F29(C25)	Rotated Hybrid Composition Function without bounds	M, N, S	30	$[-5,5]^D$

average fitness values (Ave) and standard deviation (Std) of the achieved results of all comparative algorithms across 30 individualistic runs and 1000 iterations are reported and examined for each dimension size. Table 8 illustrates that the obtained-results of the proposed AOA in dealing with thirteen test functions (F1–F13) with different dimensions are competitive; AOA got the best ranking when evaluated in 30 dimensions. This confirms that any optimization algorithm works more efficiently when the dimension is low; moreover, the obtained results when it uses high-dimensions size are competitive and not so far from the results of low-dimensions test functions.

The AOA is implemented in highly scalable optimization problems to prove its ability to solve complicated optimization problems. In this part of the experiments, as shown in Tables 9–12, the proposed AOA algorithm is



(a) F1 (b) F2 (c) F3 (d) F4

(e) F5 (f) F6 (g) F7

Fig. 6. 2-D versions of unimodal test functions (F1–F7).

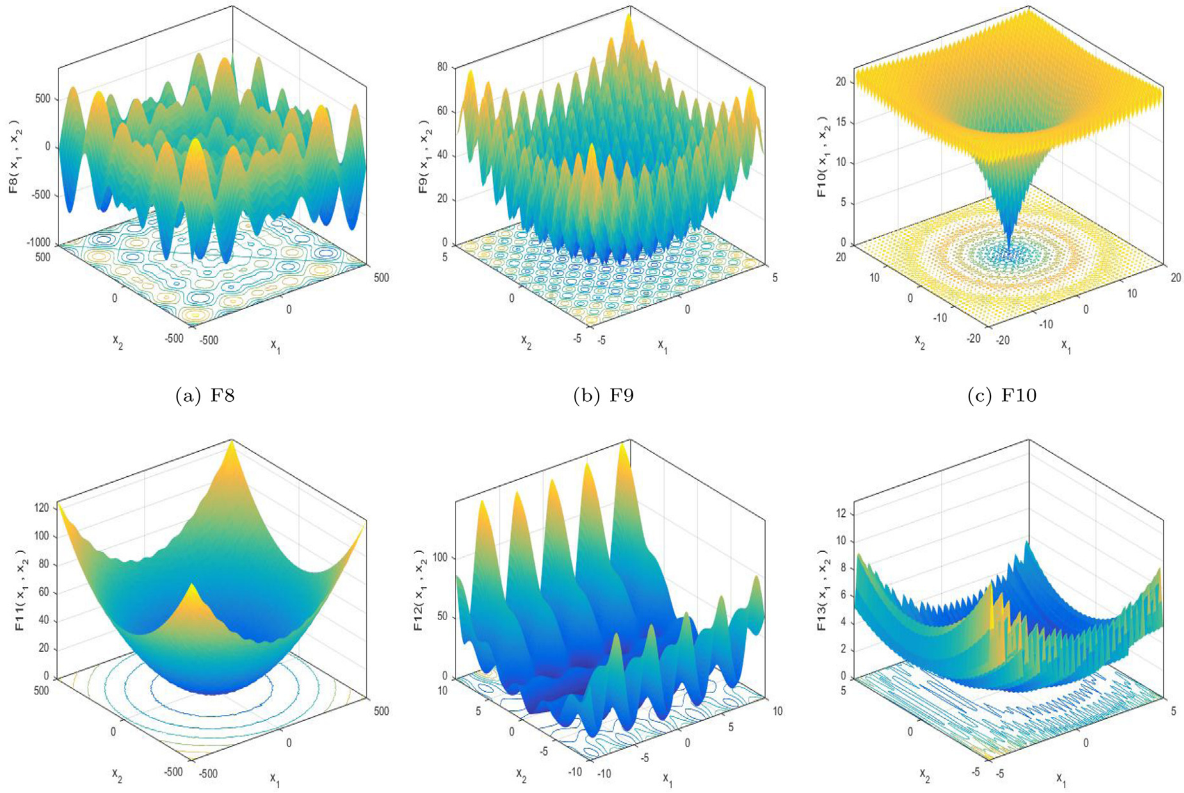
compared with other well-known optimization algorithms using thirteen benchmark functions (F1–F13) with several dimensions (30, 100, 500, and 1000). From these results, it is observed that the performance of the AOA is superior in most cases, and it is a competitor in some other cases compared to other optimization algorithms across various dimensions.

Besides the given evaluation metrics (i.e., average and standard deviation), Friedman ranking test has been carried for conducting ranking comparisons for the above mentioned algorithms in a statistical way, as shown in Table 13. The Friedman ranking test has been employed to investigate the ranking of the comparative algorithms using thirteen test functions (F1–F13) with different dimension sizes (10, 100, 500, and 1000). The obtained results show that the proposed AOA is ranked first compared to other comparative algorithms, followed by GWO is ranked second, CS is ranked third, FA is ranked fourth, GSA is ranked fifth, BBO is ranked sixth, FPA is ranked seventh, GA is ranked eighth, DE is ranked ninth, MSO is ranked tenth, PSO is ranked eleventh, and finally BAT is ranked twelfth. According to this test, the proposed AOA algorithm proved its ability to get the optimal solution by getting the first rank in different test functions compared to other comparative optimization algorithms.

In order to prove the convergence behavior of the proposed AOA, three metrics are also applied in 2D environments, which are shown in Fig. 10. The diagrams are search history, trajectory, and convergence curve.

In order to analyze the convergence performance of the proposed AOA compared to other six well-known optimization algorithms, Fig. 11 presents exemplary convergence curves using thirteen test functions (F1–F13). It is clear from Fig. 11 that the proposed AOA has a steady convergence and a slow convergence acceleration on these test functions compared with other comparative algorithms (GA, FPA, BBO, BAT, PSO, and GWO). Furthermore, the AOA obtained better solutions than other algorithms on these test functions in terms of the global search experience and convergence speed. This means that the AOA got a faster convergence rate and a more effective global search ability.

Moreover, it is obvious from Fig. 11(i), (j), and (k) that AOA does not give a distinct advantage in the first iterations. One reason causing this phenomenon is that AOA distributes the solutions' positions to various local research areas instead of accumulating all the positions in a local area based on the current best-obtained solutions. Nevertheless, the distribution mechanism, as mentioned earlier, increases the global search capability of AOA. AOA



(a) F8 (b) F9 (c) F10

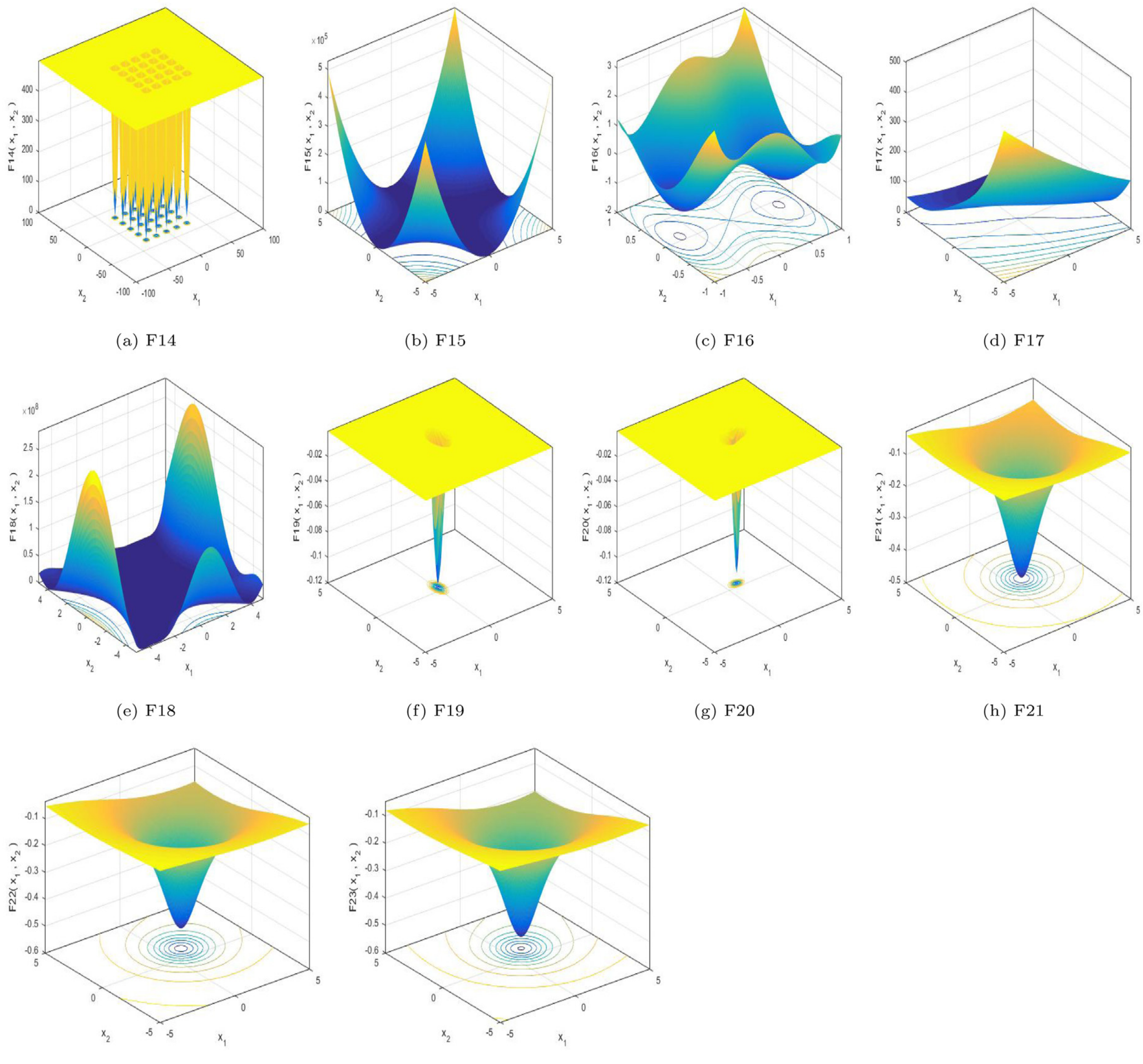
(d) F11 (e) F12 (f) F13

Fig. 7. 2-D versions of multimodal test functions (F8–F13).

can dramatically avoid sticking in local search areas and has the competitive experience of searching for global search areas.

In this part of the experiments, the average running time of the proposed AOA algorithm compared to other well-known optimization algorithms is presented in Table 14. It can be observed that the proposed AOA needs less running time than other comparative algorithms in terms of seconds. Since AOA is a population-based algorithm, there is no need for optimization processes, i.e., Multiplication, Division, Subtraction, and Addition. Consequently, we concluded that the computational performance of the proposed AOA algorithm is sufficiently better than the other comparative algorithms. Moreover, according to the Friedman ranking test in Table 14, the results of the algorithms are ranked based on its average running time for the given thirteen test functions (F1–F13). The proposed AOA algorithm obtained the first rank, followed by BBO the second-ranked and so on. These marks are also under the computational complexity of AOA.

The reported results in Table 15 confirmed that AOA obtained superior and highly competitive results on the given test functions (F14–F29). The results are superior in almost all test cases and competitive in one test case (F26). All optimization algorithms in these experiments obtained high-quality results. According to the given results in Table 15, the proposed AOA has always obtained the best results in the given test cases (F14–F29) in comparison with other well-known optimization algorithms. The AOA is proficient in achieving high-quality solutions and in overwhelming other competitors. Moreover, Friedman ranking test has also been applied for these results; the proposed AOA got the first ranking compared to other comparative methods followed by DE, CS, TLBO, FA, MFO, GWO, FPA, BBO, PSO, GA, and BAT. We can conclude that the proposed AOA is a highly competitive optimization algorithm compared with the twelve well-known optimization algorithms. Fig. 12 also confirms the superiority of the AOA algorithm based on the average fitness values of the test functions (F14-F29).



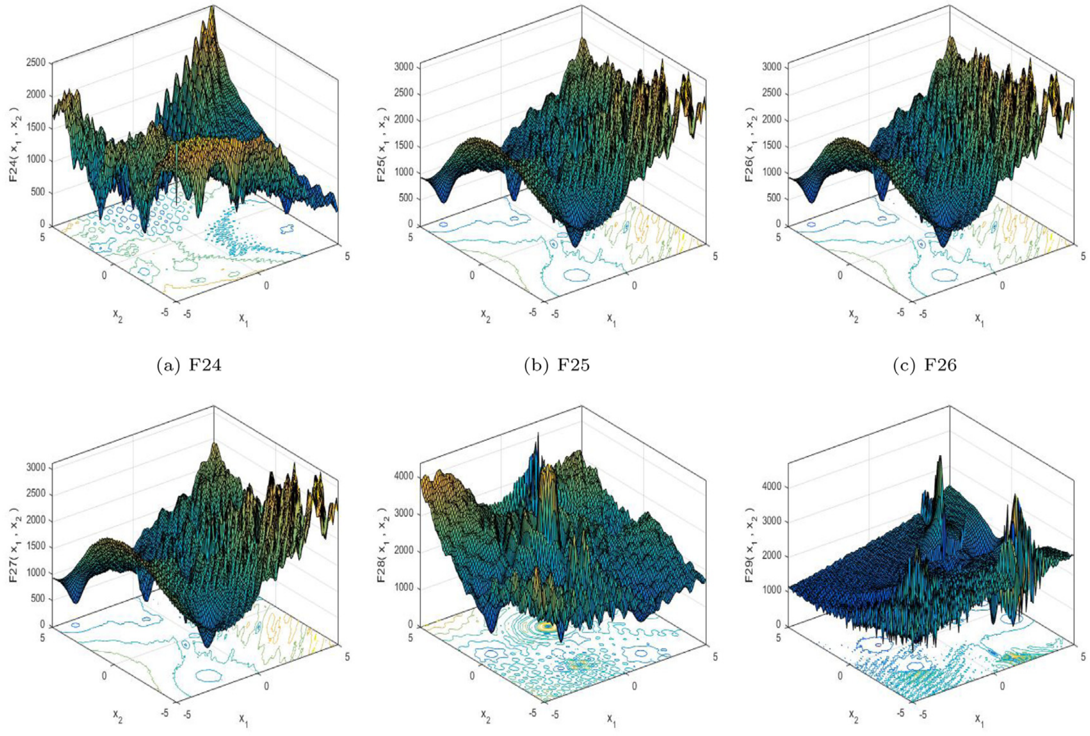
(a) F14 (b) F15 (c) F16 (d) F17

(e) F18 (f) F19 (g) F20 (h) F21

(i) F22 (j) F23

Fig. 8. 2-D versions of fixed-dimension multimodal test functions (F14-F23).

Table 16 gives the Wilcoxon signed-rank test results with a significance level at $\alpha = 0.05$ among seven comparative algorithms (GWO, BAT, FA, CS, MFO, GSA, and DE) for twenty-nine test functions (F1–F29). From Table 16, AOA outperformed GWO on all test functions except F1, F6, F8, F10, and F11. On F5 and F7, AOA and GWO got identical results, which means that there is no significant difference in two test cases. AOA is superior to BAT on all test functions. AOA outperformed FA on all test functions except F6, F8, and F11. AOA beats CS on all test functions except F6, F8, F11, and F21. AOA outperformed MFO on all test functions except F8. AOA beats GSA on all test functions except F6, F8, and F11. Finally, AOA overwhelmed DE in all test functions. Evidently, AOA can offer high-quality solutions compared to other comparative optimization algorithms in almost all test functions.



(a) F24 (b) F25 (c) F26

(d) F27 (e) F28 (f) F29

Fig. 9. 2-D versions of hybrid composition test functions (F24-F29).

Table 6
Scenarios of the tuning parameters.

Scenario No.	μ value	α value
1	0.1	1
2	0.1	5
3	0.1	9
4	0.5	1
5	0.5	5
6	0.5	9
7	0.9	1
8	0.9	5
9	0.9	9

3.2. Real-world applications

This section solves five engineering design problems using the proposed algorithm: welded beam design problem, tension/compression spring design problem, pressure vessel design problem, 3-bar truss design problem, and speed reducer problem. To address these problems, a set of 30 solutions and 500 iterations are used in each run [41,42]. The obtained results are compared with several similar techniques published in the literature. The following subsections show the results of the proposed AOA compared with the results of the state-of-art methods.

In this paper, bound-constrained and general constrained optimization problems are chosen to examine the effectiveness of the proposed AOA. For the bound-constrained optimization problems [43,44], each pattern variable is often required to provide a boundary limitation:

$$LB_j \leq x_{ij} \leq UB_j, \quad j = 1, 2, \ldots, n \tag{6}$$

Table 7
The influence of the AOA parameters (i.e., μ and α) on various classical test functions.

Function	Measure	Scenario No.								
		Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6	Scenario 7	Scenario 8	Scenario 9
F1	Worst	1.20E−06	2.07E−14	2.04E−19	2.24E−24	1.73E−48	8.78E−35	3.23E−17	3.88E−07	3.09E−07
	Average	4.53E−07	4.23E−15	5.21E−20	4.48E−25	6.22E−49	1.76E−35	9.89E−18	7.78E−08	6.40E−08
	Best	2.38E−14	1.30E−24	1.45E−30	2.48E−34	5.49E−69	2.19E−47	5.14E−20	4.60E−17	5.25E−14
	STD	6.21E−07	9.23E−15	8.82E−20	1.00E−24	8.60E−49	3.92E−35	1.44E−17	1.74E−07	1.37E−07
	Rank	9	6	4	3	1	2	5	8	7
F2	Worst	1.34E−09	2.28E−22	5.61E−40	4.97E−41	2.88E−90	2.70E−52	1.47E−14	3.49E−23	7.57E−09
	Average	2.69E−10	4.56E−23	1.12E−40	9.93E−42	5.77E−91	5.40E−53	2.95E−15	7.65E−24	1.52E−09
	Best	2.08E−19	2.50E−35	5.31E−48	1.48E−56	1.23E−115	1.81E−69	2.98E−24	1.10E−35	1.21E−24
	STD	6.01E−10	1.02E−22	2.51E−40	2.22E−41	1.29E−90	1.21E−52	6.59E−15	1.53E−23	3.38E−09
	Rank	8	6	4	3	1	2	7	5	9
F3	Worst	3.70E−05	7.56E−09	2.54E−17	2.63E−27	1.93E−42	5.48E−27	6.77E−06	1.41E−07	1.41E−06
	Average	8.76E−06	1.54E−09	5.09E−18	5.69E−28	3.89E−43	1.10E−27	1.35E−06	2.81E−08	2.88E−07
	Best	1.45E−13	6.16E−22	1.19E−27	7.43E−34	1.28E−62	9.74E−38	9.60E−19	5.42E−18	1.32E−10
	STD	1.61E−05	3.37E−09	1.14E−17	1.15E−27	8.61E−43	2.45E−27	3.03E−06	6.29E−08	6.29E−07
	Rank	9	5	4	2	1	3	8	6	7
F4	Worst	1.24E−03	1.76E−05	8.22E−09	3.86E−14	1.66E−23	7.24E−16	1.81E−07	6.03E−05	1.47E−03
	Average	2.58E−04	3.59E−06	1.64E−09	7.78E−15	3.33E−24	3.36E−16	3.79E−08	1.21E−05	4.94E−04
	Best	1.84E−06	1.66E−10	2.19E−15	4.07E−21	1.66E−30	9.95E−19	1.95E−10	5.75E−12	7.18E−08
	STD	5.49E−04	7.85E−06	3.68E−09	1.72E−14	7.43E−24	3.32E−16	8.00E−08	2.69E−05	6.72E−04
	Rank	8	6	4	3	1	2	5	7	9
F5	Worst	8.77E+00	8.66E+00	8.53E+00	8.31E+00	8.18E+00	8.26E+00	8.54E+00	8.67E+00	8.74E+00
	Average	8.64E+00	8.57E+00	8.33E+00	8.15E+00	7.83E+00	8.11E+00	8.50E+00	8.54E+00	8.65E+00
	Best	8.56E+00	8.41E+00	8.04E+00	7.99E+00	7.71E+00	7.96E+00	8.43E+00	8.49E+00	8.51E+00
	STD	8.28E−02	9.49E−02	1.80E−01	1.44E−01	1.97E−01	1.37E−01	4.34E−02	7.65E−02	8.90E−02
	Rank	8	7	4	3	1	2	5	6	9
F6	Worst	5.78E+00	5.70E+00	5.88E+00	5.89E+00	5.32E+00	5.58E+00	6.06E+00	6.00E+00	5.94E+00
	Average	5.47E+00	5.15E+00	5.55E+00	5.13E+00	4.92E+00	5.14E+00	5.60E+00	5.28E+00	5.64E+00
	Best	4.89E+00	4.62E+00	5.28E+00	4.40E+00	4.57E+00	4.40E+00	4.92E+00	4.49E+00	5.28E+00
	STD	3.68E−01	4.15E−01	2.19E−01	6.20E−01	3.12E−01	4.64E−01	4.90E−01	5.36E−01	2.44E−01
	Rank	6	4	7	2	1	3	8	5	9
F7	Worst	2.52E−03	2.63E−04	6.35E−04	7.72E−04	3.68E−04	4.29E−04	6.85E−04	3.19E−03	3.45E−03
	Average	7.30E−04	1.47E−04	4.65E−04	2.21E−04	2.00E−04	1.36E−04	2.96E−04	1.13E−03	1.29E−03
	Best	4.83E−05	6.35E−05	2.15E−04	6.33E−06	9.57E−05	4.48E−06	7.78E−05	5.83E−05	3.83E−05
	STD	1.01E−03	7.95E−05	1.61E−04	3.23E−04	1.11E−04	1.70E−04	2.52E−04	1.24E−03	1.58E−03
	Rank	7	2	6	4	3	1	5	8	9
F8	Worst	−1.29E+03	−1.50E+03	−1.77E+03	−1.49E+03	−1.56E+03	−1.84E+03	−1.79E+03	−1.29E+03	−1.70E+03
	Average	−1.63E+03	−1.90E+03	−1.96E+03	−1.84E+03	−1.83E+03	−1.95E+03	−1.96E+03	−2.03E+03	−1.91E+03
	Best	−1.78E+03	−2.48E+03	−2.29E+03	−2.08E+03	−2.12E+03	−2.18E+03	−2.29E+03	−2.36E+03	−2.11E+03
	STD	1.95E+02	3.90E+02	2.04E+02	2.31E+02	2.11E+02	1.36E+02	2.23E+02	4.64E+02	1.85E+02
	Rank	9	6	3	7	8	4	2	1	5
F9	Worst	7.51E−05	1.90E−05	2.17E−05	9.62E−06	2.53E−06	8.60E−06	2.23E−05	3.75E−05	6.39E−05
	Average	4.27E−05	1.37E−05	1.35E−05	6.04E−06	1.61E−06	4.71E−06	1.50E−05	3.34E−05	5.10E−05
	Best	3.68E−06	1.03E−05	8.82E−06	3.51E−06	6.67E−07	1.96E−06	2.62E−06	2.73E−05	3.22E−05
	STD	3.12E−05	3.24E−06	4.86E−06	2.27E−06	8.02E−07	2.54E−06	7.76E−06	4.55E−06	1.26E−05
	Rank	8	5	4	3	1	2	6	7	9
F10	Worst	8.70E−08	1.51E−14	8.88E−16	8.88E−16	8.88E−16	8.88E−16	1.51E−14	1.68E−13	9.27E−09
	Average	1.74E−08	3.73E−15	8.88E−16	8.88E−16	8.88E−16	8.88E−16	7.28E−15	3.43E−14	1.87E−09
	Best	7.99E−15	8.88E−16	8.88E−16	8.88E−16	8.88E−16	8.88E−16	8.88E−16	8.88E−16	6.26E−13
	STD	3.89E−08	6.36E−15	0.00E+00	0.00E+00	0.00E+00	0.00E+00	5.27E−15	7.47E−14	4.14E−09
	Rank	9	5	1	1	1	1	6	7	8
F11	Worst	1.04E+00	1.48E−01	2.84E−01	1.24E−01	2.73E−05	6.60E−05	8.24E−01	1.02E+00	1.02E+00
	Average	7.51E−01	7.55E−02	8.53E−02	2.48E−02	2.23E−05	5.13E−05	2.44E−01	6.48E−01	7.30E−01
	Best	3.31E−01	5.61E−04	1.13E−04	3.61E−05	1.52E−05	3.78E−05	1.70E−04	1.17E−02	2.91E−01

(*continued on next page*)

Table 7 (*continued*).

Function	Measure	Scenario No.								
		Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	Scenario 6	Scenario 7	Scenario 8	Scenario 9
	STD	2.91E−01	7.00E−02	1.27E−01	5.53E−02	4.63E−06	1.21E−05	3.67E−01	4.31E−01	3.61E−01
	Rank	9	4	5	3	1	2	6	7	8
F12	Worst	1.56E+00	1.57E+00	9.14E−01	6.50E−01	5.47E−01	6.09E−01	9.16E−01	1.14E+00	1.64E+00
	Average	1.22E+00	1.04E+00	7.57E−01	5.82E−01	4.18E−01	5.09E−01	8.23E−01	8.62E−01	1.17E+00
	Best	1.01E+00	7.13E−01	6.78E−01	3.87E−01	2.42E−01	4.47E−01	7.80E−01	6.86E−01	8.32E−01
	STD	2.25E−01	3.29E−01	9.14E−02	1.12E−01	1.12E−01	6.33E−02	5.33E−02	1.68E−01	3.09E−01
	Rank	9	7	4	3	1	2	5	6	8
F13	Worst	9.89E−01	9.89E−01	9.89E−01	9.89E−01	9.89E−01	9.89E−01	9.89E−01	9.89E−01	9.89E−01
	Average	9.38E−01	9.17E−01	8.93E−01	9.24E−01	8.98E−01	8.98E−01	9.57E−01	9.89E−01	9.55E−01
	Best	8.90E−01	8.76E−01	8.43E−01	8.34E−01	7.89E−01	7.54E−01	8.84E−01	9.89E−01	9.33E−01
	STD	4.95E−02	4.53E−02	5.72E−02	6.87E−02	7.93E−02	8.91E−02	4.73E−02	5.17E−05	3.06E−02
	Rank	6	4	1	5	2	3	8	9	7
Mean	Rank	8.08	5.15	3.92	3.23	1.77	2.23	5.85	6.31	8.00
Final	Ranking	9	5	4	3	1	2	6	7	8

Table 8
Results of the test functions (F1–F13), with 30, 100, 500, and 1000 dimensions.

Dim	30			100			500			1000		
F	Ave	Std	Rank	Ave	Std	Rank	Ave	Std	Rank	Ave	Std	Rank
F1	6.67E−07	7.45E−07	1	2.41E−06	6.31E−06	2	2.70E−02	3.14E−02	3	7.50E−02	4.12E−01	4
F2	0.00E+00	0.00E+00	1	3.58E−08	5.25E−08	2	1.09E−02	1.54E−03	3	3.30E−01	4.54E−01	4
F3	6.87E−06	6.87E−06	1	1.99E−04	2.46E−02	2	3.57E−01	3.01E−02	3	2.21E+00	4.94E+00	4
F4	1.40E−03	1.90E−03	1	3.40E−03	3.23E−03	2	4.00E−02	4.42E−02	3	4.00E−02	4.00E−02	3
F5	2.49E+01	3.64E−01	1	9.79E+01	6.55E−01	2	4.97E+02	5.32E−01	3	9.21E+02	9.69E+02	4
F6	3.47E−04	3.47E−04	1	2.38E+01	2.64E+01	2	8.22E+01	2.21E+01	3	2.01E+02	6.21E+02	4
F7	3.92E−06	3.92E−06	1	2.03E−04	3.24E−03	3	2.98E−05	2.95E−05	2	2.18E−03	2.18E−03	4
F8	−1.22E+04	1.22E+03	3	−1.21E+04	−2.45E+04	4	−1.73E+04	1.24E+04	2	−2.07E+05	3.24E+04	1
F9	3.42E−07	3.42E−07	1	8.46E−06	9.24E−06	2	1.58E−02	1.58E−02	3	3.99E−02	2.64E−02	4
F10	8.88E−16	8.88E−16	1	2.02E−04	4.14E−03	2	1.93E−03	3.50E−04	3	1.15E−02	2.36E−03	4
F11	0.00E+00	0.00E+00	1	1.22E+00	1.22E+00	4	1.50E−02	3.20E−02	3	6.68E−03	9.36E−03	2
F12	4.28E−06	4.28E−06	1	2.40E−02	3.84E−01	2	2.09E−01	2.14E−01	3	5.18E−01	5.65E−01	4
F13	3.10E−01	3.10E−01	1	4.42E+00	9.54E+00	2	4.98E+01	6.25E+01	3	1.00E+02	1.00E+02	4
Mean	Rank		1.153			2.384			2.846			3.538
Final	Ranking		1			2			3			4

where LB_j and UB_j are the lower bound and upper bound of the position x_{ij}, and n is the number of given positions. Furthermore, a general constrained problem can be usually presented as:

$$\begin{aligned} &min \quad f(X) \\ &X = \{x_{11}, x_{1j}, \ldots.., x_{1n}\} \\ s.t. \quad &g_i(X) \leq 0,\ j = 1, 2, \ldots, m \\ &h_k(X) = 0,\ k = 1, 2, \ldots, l \\ &LB_j \leq x_{ij} \leq UB_j, \quad j = 1, 2, \ldots, n \end{aligned} \tag{7}$$

where m is the number of various constraints, and l is the number of equilibrium constraints.

In the performance evaluation of the proposed AOA, all the constrained optimization problems in Eq. (7) are mapped into the bound-constrained design by applying the static cost function. For any infeasible solution, a cost function will be combined into the underlying objective function. Due to its convenience in employment, the static cost function is simplified. It only needs an auxiliary cost function and is proper for all various problems [6,45,46].

Table 9
Results of the test functions (F1–F13), with 30 dimensions.

F	GA		PSO		BBO		FPA	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	1.03E+03	5.79E+02	1.83E+04	3.01E+03	7.59E+01	2.75E+01	2.01E+13	5.60E+02
F2	2.47E+01	5.68E+00	3.58E+02	1.35E+03	1.36E−03	7.45E−03	3.22E+01	5.55E+00
F3	2.65E+04	3.44E+03	4.05E+04	8.21E+03	1.21E+04	2.69E+03	1.41E+03	5.59E+02
F4	5.17E+01	1.05E+01	4.39E+01	3.64E+00	3.02E+01	4.39E+00	2.38E+01	2.77E+00
F5	1.95E+04	1.31E+04	1.96E+07	6.25E+06	1.82E+03	9.40E+02	3.17E+05	1.75E+05
F6	9.01E+02	2.84E+02	1.87E+04	2.92E+03	6.71E+01	2.20E+01	1.70E+03	3.13E+02
F7	1.91E−01	1.50E−01	1.07E+01	3.05E+00	2.91E−03	1.83E−03	3.44E−01	1.10E−01
F8	−1.26E+04	4.51E+00	−3.86E+03	2.49E+02	−1.24E+04	3.50E+01	−6.45E+03	3.03E+02
F9	9.04E+00	4.58E+00	2.87E+02	1.95E+01	0.00E+00	0.00E+00	1.82E+02	1.24E+01
F10	1.36E+01	1.51E+00	1.75E+01	3.67E−01	2.13E+00	3.53E−01	7.14E+00	1.08E+00
F11	1.01E+01	2.43E+00	1.70E+02	3.17E+01	1.46E+00	1.69E−01	1.73E+01	3.63E+00
F12	4.77E+00	1.56E+00	1.51E+07	9.88E+06	6.68E−01	2.62E−01	3.05E+02	1.04E+03
F13	1.52E+01	4.52E+00	5.73E+07	2.68E+07	1.82E+00	3.41E−01	9.59E+04	1.46E+05
F	GWO		BAT		FA		CS	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	1.18E−27	1.47E−27	6.59E+04	7.51E+03	7.11E−03	3.21E−03	9.06E−04	4.55E−04
F2	9.71E−17	5.60E−17	2.71E+08	1.30E+09	4.34E−01	1.84E−01	1.49E−01	2.79E−02
F3	5.12E−05	2.03E−04	1.38E+05	4.72E+04	1.66E+03	6.72E+02	2.10E−01	5.69E−02
F4	1.24E−06	1.94E−06	8.51E+01	2.95E+00	1.11E−01	4.75E−02	9.65E−02	1.94E−02
F5	2.70E+01	7.78E−01	2.10E+08	4.17E+07	7.97E+01	7.39E+01	2.76E+01	4.51E−01
F6	8.44E−01	3.18E−01	6.69E+04	5.87E+03	6.94E−03	3.61E−03	3.13E−03	1.30E−03
F7	1.70E−03	1.06E−03	4.57E+01	7.82E+00	6.62E−02	4.23E−02	7.29E−02	2.21E−02
F8	−5.91E+03	7.10E+02	−2.33E+03	2.96E+02	−5.85E+03	1.16E+03	−5.19E+01	1.76E+01
F9	2.19E+00	3.69E+00	1.92E+02	3.56E+01	3.82E+01	1.12E+01	1.51E+01	1.25E+00
F10	1.03E−03	1.70E−14	1.92E+01	2.43E−01	4.58E−02	1.20E−02	3.29E−02	7.93E−03
F11	4.76E−03	8.57E−03	6.01E+02	5.50E+01	4.23E−03	1.29E−03	4.29E−05	2.00E−05
F12	4.83E−02	2.12E−02	4.71E+08	1.54E+08	3.13E−04	1.76E−04	5.57E−05	4.96E−05
F13	5.96E−01	2.23E−01	9.40E+08	1.67E+08	2.08E−03	9.62E−04	8.19E−03	6.74E−03
F	MFO		GSA		DE		AOA	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	1.01E+03	3.05E+03	6.08E+02	4.64E+02	1.33E−03	5.92E−04	6.67E−07	7.45E−07
F2	3.19E+01	2.06E+01	2.27E+01	3.36E+00	6.83E−03	2.06E−03	0.00E−00	0.00E−00
F3	2.34E+04	1.41E+04	1.35E+05	4.86E+04	3.97E+04	5.37E+03	6.87E−06	6.87E−06
F4	7.00E+01	7.06E+00	7.87E+01	2.81E+00	1.15E+01	2.37E+00	1.40E−03	1.90E−03
F5	7.35E+03	2.26E+04	7.41E+02	7.81E+02	1.06E+02	1.01E+02	2.49E+01	3.64E−01
F6	2.68E+03	5.84E+03	3.08E+03	8.98E+02	1.44E−03	5.38E−04	3.47E−04	3.47E−04
F7	4.50E+00	9.21E+00	1.12E−01	3.76E−02	5.24E−02	1.37E−02	3.92E−06	3.92E−06
F8	−8.48E+03	7.98E+02	−2.35E+03	3.82E+02	−6.82E+03	3.94E+02	−1.22E+04	1.22E+03
F9	1.59E+02	3.21E+01	3.10E+01	1.36E+01	1.58E+02	1.17E+01	3.42E−07	3.42E−07
F10	1.74E+01	4.95E+00	3.74E+00	1.71E−01	1.21E−02	3.30E−03	8.88E−16	8.88E−16
F11	3.10E+01	5.94E+01	4.86E−01	4.97E−02	3.52E−02	7.20E−02	0.00E+00	0.00E+00
F12	2.46E+02	1.21E+03	4.63E−01	1.37E−01	2.25E−05	1.70E−03	4.28E−06	4.28E−06
F13	2.73E+07	1.04E+08	7.61E+00	1.22E+00	9.12E−03	1.16E−02	3.10E−01	3.10E−01

Using this procedure, the above-mentioned constrained optimization problem can be presented as

$$f(X) = f(X)\sum_{j=1}^{m} Pe_j max\{g_i(X), 0\} + \sum_{k=1}^{n} Pe_k max\{|h_k(X) - \varepsilon|, 0\} \tag{8}$$

where Pe_j and Pe_k are cost functions and usually charged a significant value. ε is the error of equilibrium constraints, which is set to $1e$-6 in this paper.

Table 10
Results of the test functions (F1–F13), with 100 dimensions.

F	GA		PSO		BBO		FPA	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	5.41E+04	1.42E+04	1.06E+05	8.47E+03	2.85E+03	4.49E+02	1.39E+04	2.71E+03
F2	2.53E+02	1.41E+01	6.06E+23	2.18E+24	1.59E+01	3.74E+00	1.01E+02	9.36E+00
F3	2.53E+05	5.03E+04	4.22E+05	7.08E+04	1.70E+05	2.02E+04	1.89E+04	5.44E+03
F4	8.19E+01	3.15E+00	6.07E+01	3.05E+00	7.08E+01	4.73E+00	3.51E+01	3.37E+00
F5	2.37E+07	8.43E+06	2.42E+08	4.02E+07	4.47E+05	2.05E+05	4.64E+06	1.98E+06
F6	5.42E+04	1.09E+04	1.07E+05	9.70E+03	2.85E+03	4.07E+02	1.26E+04	2.06E+03
F7	2.73E+01	4.45E+01	3.41E+02	8.74E+01	1.25E+00	5.18E+00	5.84E+00	2.16E+00
F8	−4.10E+04	1.14E+02	−7.33E+03	4.75E+02	−3.85E+04	2.80E+02	−1.28E+04	4.64E+02
F9	3.39E+02	4.17E+01	1.16E+03	5.74E+01	9.11E+00	2.73E+00	8.47E+02	4.01E+01
F10	1.82E+01	4.35E−01	1.91E+01	2.04E−01	5.53E+33	4.72E−01	8.21E+00	1.14E+00
F11	5.14E+02	1.05E+02	9.49E+02	6.00E+01	2.24E+01	4.35E+00	1.19E+02	2.00E+01
F12	4.55E+06	8.22E+06	3.54E+08	8.75E+07	3.03E+02	1.48E+03	1.55E+05	1.74E+05
F13	5.26E+07	3.76E+07	8.56E+08	2.16E+08	6.82E+04	3.64E+04	2.76E+06	1.80E+06

F	GWO		BAT		FA		CS	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	1.59E−12	1.63E−12	2.72E+05	1.42E+04	3.05E−01	5.60E−02	3.17E−01	5.28E−02
F2	4.31E−08	1.46E−08	6.00E+43	1.18E+44	1.45E+01	6.73E+00	4.05E+00	3.16E−01
F3	4.09E+02	2.77E+02	1.43E+06	6.21E+05	4.65E+04	6.92E+03	6.88E+00	1.02E+00
F4	8.89E−01	9.30E−01	9.41E+01	1.49E+00	1.91E+01	3.12E+00	2.58E−01	2.80E−02
F5	9.79E+01	6.75E−01	1.10E+09	9.47E+07	8.46E+02	8.13E+02	1.33E+02	7.34E+00
F6	1.03E+01	1.05E+00	2.69E+05	1.25E+04	2.95E−01	5.34E−02	2.65E+00	3.94E−01
F7	7.60E−03	2.66E−03	3.01E+02	2.66E+01	5.65E−01	1.64E−01	1.21E+00	2.65E−01
F8	−1.67E+04	2.62E+03	−4.07E+03	9.37E+02	−1.81E+04	3.23E+03	−2.84E+18	6.91E+18
F9	1.03E+01	9.02E+00	7.97E+02	6.33E+01	2.36E+02	2.63E+01	1.72E+02	9.24E+00
F10	1.20E−07	5.07E−08	1.94E+01	6.50E−02	9.81E−01	2.55E−01	3.88E−01	5.23E−02
F11	4.87E−03	1.07E−02	2.47E+03	1.03E+02	1.19E−01	2.34E−02	4.56E−03	9.73E−04
F12	2.87E−01	6.41E−02	2.64E+09	2.69E+08	4.45E+00	1.32E+00	2.47E−02	5.98E−03
F13	6.87E+00	3.32E−01	5.01E+09	3.93E+08	4.50E+01	2.24E+01	5.84E+00	1.21E+00

F	MFO		GSA		DE		AOA	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	6.20E+04	1.25E+04	3.62E−01	4.14E−01	8.26E+03	1.32E+03	2.41E−06	6.31E−06
F2	2.46E+02	4.48E+01	3.27E−01	2.75E−01	1.21E+02	2.33E+01	3.58E−08	5.25E−08
F3	2.15E+05	4.43E+04	4.33E+04	8.20E+04	5.01E+05	5.87E+04	1.99E−04	2.46E−02
F4	9.31E+01	2.13E+00	2.36E−01	6.66E−01	9.62E+01	1.00E+00	3.40E−03	3.23E−03
F5	1.44E+08	7.50E+07	9.67E+02	7.77E+02	1.99E+07	5.80E+06	9.79E+01	6.55E−01
F6	6.68E+04	1.46E+04	3.27E−01	6.98E−02	8.07E+03	1.64E+03	2.38E+01	2.64E+01
F7	2.56E+02	8.91E+01	1.50E−01	5.39E−02	1.96E+01	5.66E+00	2.03E−04	3.24E−03
F8	−2.30E+04	1.98E+03	−1.71E+04	3.54E+03	−1.19E+04	5.80E+02	−1.21E+04	−2.45E+04
F9	8.65E+02	8.01E+01	1.02E+01	5.57E+01	1.03E+03	4.03E+01	8.46E−06	9.24E−06
F10	1.99E+01	8.58E−02	1.66E−01	9.10E−01	1.22E+01	8.31E−01	2.02E−04	4.14E−03
F11	5.60E+02	1.23E+02	1.36E−01	2.64E−02	7.42E+01	1.40E+01	1.22E+00	1.22E+00
F12	2.82E+08	1.45E+08	3.03E−00	1.02E−00	3.90E+07	1.88E+07	2.40E−02	3.84E−01
F13	6.68E+08	3.05E+08	5.47E+01	8.34E+01	7.19E+07	2.73E+07	4.42E+00	9.54E+00

3.2.1. Welded beam design problem

The main objective of this problem, welded beam design, is to find the minimum fabrication cost by defining the optimal value of the given variables, which are four optimization variables as shown in Fig. 13, namely, length of attached part of bar (l), thickness of weld (h), the height of the bar (t), and thickness of the bar (b). The given variables need to be satisfied with seven constraints. The mathematical representation of this problem can be found in the original paper.

The proposed algorithm (AOA) is applied for solving the welded beam design and compared with several optimization algorithms published in the literature. From Table 17, we concluded that the results of the AOA are

Table 11
Results of the test functions (F1–F13), with 500 dimensions.

F	GA		PSO		BBO		FPA	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	6.06E+05	7.01E+04	6.42E+05	2.96E+04	1.60E+05	9.76E+03	8.26E+04	1.32E+04
F2	1.94E+03	7.03E+01	6.08E+09	1.70E+10	5.95E+02	1.70E+01	5.13E+02	4.84E+01
F3	5.79E+06	9.08E+05	1.13E+07	1.43E+06	2.98E+06	3.87E+05	5.34E+05	1.34E+05
F4	9.59E+01	1.20E+00	8.18E+01	1.49E+00	9.35E+01	9.05E−01	4.52E+01	4.28E+00
F5	1.79E+09	4.11E+08	1.84E+09	1.11E+08	2.07E+08	2.08E+07	3.30E+07	8.76E+06
F6	6.27E+05	7.43E+04	6.57E+05	3.29E+04	1.68E+05	8.23E+03	8.01E+04	9.32E+03
F7	9.10E+03	2.20E+03	1.43E+04	1.51E+03	2.62E+03	3.59E+02	2.53E+02	6.28E+01
F8	−1.31E+05	2.31E+04	−1.65E+04	9.99E+02	−1.42E+05	1.98E+03	−3.00E+04	1.14E+03
F9	3.29E+03	1.96E+02	6.63E+03	1.07E+02	7.86E+02	3.42E+01	4.96E+03	7.64E+01
F10	1.96E+01	2.04E−01	1.97E+01	1.04E−01	1.44E+01	2.22E−01	8.55E+00	8.66E−01
F11	5.42E+03	7.32E+02	5.94E+03	3.19E+02	1.47E+03	8.10E+01	6.88E+02	8.17E+01
F12	2.79E+09	1.11E+09	3.51E+09	4.16E+08	1.60E+08	3.16E+07	4.50E+06	3.37E+06
F13	8.84E+09	2.00E+09	6.82E+09	8.45E+08	5.13E+08	6.59E+07	3.94E+07	1.87E+07
F	GWO		BAT		FA		CS	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	1.42E−03	3.99E−04	1.52E+06	3.58E+04	6.30E+04	8.47E+03	6.80E+00	4.93E−01
F2	1.10E−02	1.93E−03	8.34E+09	1.70E+10	7.13E+02	3.76E+01	4.57E+01	2.05E+00
F3	3.34E+05	7.95E+04	3.37E+07	1.41E+07	1.19E+06	1.88E+05	2.03E+02	2.72E+01
F4	6.51E+01	5.72E+00	9.82E+01	3.32E−01	5.00E+01	1.73E+00	4.06E−01	3.03E−02
F5	4.95E+02	5.23E−01	6.64E+09	2.23E+08	2.56E+07	6.14E+06	1.21E+03	7.04E+01
F6	9.22E+01	2.15E+00	1.53E+06	3.37E+04	6.30E+04	8.91E+03	8.27E+01	2.24E+00
F7	4.67E−02	1.12E−02	2.32E+04	1.15E+03	3.71E+02	6.74E+01	8.05E+01	1.37E+01
F8	−5.70E+04	3.12E+03	−9.03E+03	2.12E+03	−7.27E+04	1.15E+04	−2.10E+17	1.14E+18
F9	7.84E+01	3.13E+01	6.18E+03	1.20E+02	2.80E+03	1.42E+02	2.54E+03	5.21E+01
F10	1.93E−03	3.50E−04	2.04E+01	3.25E−02	1.24E+01	4.46E−01	1.07E+00	6.01E−02
F11	1.55E−02	3.5.E-02	1.38E+04	3.19E+02	5.83E+02	7.33E+01	2.66E−02	2.30E−03
F12	7.42E−01	4.38E−02	1.70E+10	6.29E+08	8.67E+05	6.23E+05	3.87E−01	2.47E−02
F13	5.06E+01	1.30E+00	3.17E+10	9.68E+08	2.29E+07	9.46E+06	6.00E+01	1.13E+00
F	MFO		GSA		DE		AOA	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	1.15E+06	3.54E+04	2.14E+06	1.94E+04	7.43E+05	3.67E+04	2.70E−02	3.14E−02
F2	3.00E+08	1.58E+09	2.31E+01	1.63E+01	3.57E+09	1.70E+10	1.09E−02	1.54E−03
F3	4.90E+06	1.02E+06	1.06E+05	3.70E+05	1.20E+07	1.49E+06	3.57E−01	3.01E−02
F4	9.88E+01	4.15E−01	4.02E+01	2.67E+00	9.29E+01	2.33E−01	4.00E−02	4.42E−02
F5	5.01E+09	2.50E+08	4.97E+04	3.07E+03	4.07E+09	1.25E+09	4.97E+02	5.32E−01
F6	1.16E+06	3.48E+04	7.82E+04	2.50E+03	7.23E+05	3.28E+04	8.22E+01	2.21E+01
F7	3.84E+04	2.24E+03	1.71E+01	4.80E+01	2.39E+04	2.72E+03	2.98E−05	2.95E−05
F8	−6.29E+04	5.71E+03	−5.02E+05	1.00E+05	−2.67E+04	1.38E+03	−1.73E+04	1.24E+04
F9	6.96E+03	1.48E+02	3.59E+03	4.84E+03	7.14E+03	1.05E+02	1.58E−02	1.58E−02
F10	2.03E+01	1.48E−01	7.62E+00	2.33E+00	2.06E+01	2.45E−01	1.93E−03	3.50E−04
F11	1.03E+04	4.43E+02	5.32E+02	6.36E+02	6.75E+03	2.97E+02	1.50E−02	3.20E−02
F12	1.20E+10	6.28E+08	4.61E+03	2.40E+02	1.60E+01	2.34E+09	2.09E−01	2.14E−01
F13	2.23E+10	1.13E+09	4.98E+04	9.97E+04	2.42E+10	6.39E+09	4.98E+01	6.25E+01

better than all other comparative algorithms. Hence, it can be declared that the AOA can find the best possible optimal solution (i.e., design) for solving the welded beam design. Moreover, Fig. 14 shows the qualitative results for the welded beam design problem.

3.2.2. Tension/compression spring design problem

The main objective of the tension/compression spring design problem is to find the minimum weight of the tension/compression spring to satisfy its design constraints: shear stress, surge frequency, and deflection as shown

Table 12
Results of the test functions (F1–F13), with 1000 dimensions.

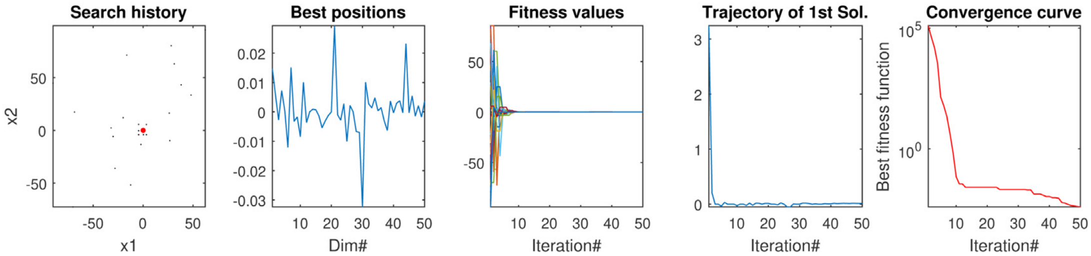
F	GA		PSO		BBO		FPA	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	1.36E+06	1.79E+05	1.36E+06	6.33E+04	6.51E+05	2.37E+04	1.70E+05	2.99E+04
F2	4.29E+03	8.86E+01	1.79E+10	1.79E+10	1.96E+03	2.18E+01	8.34E+02	8.96E+01
F3	2.29E+07	3.93E+06	3.72E+07	1.16E+07	9.92E+06	1.48E+06	1.95E+06	4.20E+05
F4	9.79E+01	7.16E−01	8.92E+01	2.39E+00	9.73E+01	7.62E−01	5.03E+01	5.37E+00
F5	4.73E+09	9.63E+08	3.72E+09	2.76E+08	1.29E+09	6.36E+07	7.27E+07	1.84E+07
F6	1.52E+06	1.88E+05	1.38E+06	6.05E+04	6.31E+05	1.82E+04	1.60E+05	1.86E+04
F7	4.45E+04	8.40E+03	6.26E+04	4.16E+03	3.84E+04	2.91E+03	1.09E+03	3.49E+02
F8	−1.94E+05	9.74E+03	−2.30E+04	1.70E+03	−2.29E+05	3.76E+03	−4.25E+04	1.47E+03
F9	8.02E+03	3.01E+02	1.35E+04	1.83E+02	2.86E+03	9.03E+01	1.10E+04	1.57E+02
F10	1.95E+01	2.55E−01	1.98E+01	1.24E−01	1.67E+01	8.63E−02	2.62E+00	9.10E−01
F11	1.26E+04	1.63E+03	1.23E+04	5.18E+02	5.75E+03	1.78E+02	1.52E+03	2.66E+02
F12	1.14E+01	1.27E+09	7.73E+09	6.72E+08	1.56E+09	1.46E+08	8.11E+06	3.46E+06
F13	1.91E+01	4.21E+09	1.58E+10	1.56E+09	4.17E+09	2.54E+08	8.96E+07	3.65E+07
F	GWO		BAT		FA		CS	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	2.24E−01	4.72E−02	3.12E+06	4.61E+04	3.20E+05	2.11E+04	1.65E+01	1.27E+00
F2	7.11E−01	4.96E−01	1.79E+10	1.79E+10	1.79E+10	1.79E+10	1.02E+02	3.49E+00
F3	1.49E+06	2.43E+05	1.35E+08	4.76E+07	4.95E+06	7.19E+05	8.67E+02	1.10E+02
F4	7.94E+01	2.77E+00	9.89E+01	2.22E−01	6.06E+01	2.69E+00	4.44E−01	2.24E−02
F5	1.06E+03	3.07E+01	1.45E+10	3.20E+08	2.47E+08	3.24E+07	2.68E+03	1.27E+02
F6	2.03E+02	2.45E+00	3.11E+06	6.29E+04	3.18E+05	2.47E+04	2.07E+02	4.12E+00
F7	1.47E−01	3.28E−02	1.25E+05	3.93E+03	4.44E+03	4.00E+02	4.10E+02	8.22E+01
F8	−8.64E+04	1.91E+04	−1.48E+04	3.14E+03	−1.08E+05	1.69E+04	−9.34E+04	2.12E+04
F9	2.06E+02	4.81E+01	1.40E+04	1.85E+02	7.17E+03	1.88E+02	6.05E+03	1.41E+02
F10	1.88E−02	2.74E−03	2.07E+01	2.23E−02	1.55E+01	2.42E−01	1.18E+00	5.90E−02
F11	6.58E−02	8.82E−02	2.83E+04	4.21E+02	2.87E+03	1.78E+02	3.92E−02	3.58E−03
F12	1.15E+00	1.82E−01	3.63E+10	1.11E+09	6.76E+07	1.80E+07	6.53E−01	2.45E−02
F13	1.21E+02	1.11E+01	6.61E+10	1.40E+09	4.42E+08	7.91E+07	1.32E+02	1.48E+00
F	MFO		GSA		DE		AOA	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
F1	2.73E+06	4.70E+04	2.73E+01	7.67E+01	2.16E+06	3.39E+05	7.50E−02	4.12E−01
F2	1.79E+10	1.79E+10	1.79E+03	1.79E+03	1.79E+10	1.79E+10	3.30E−01	4.54E−01
F3	1.94E+07	3.69E+06	8.61E+03	1.33E+03	5.03E+07	4.14E+06	2.21E+00	4.94E+00
F4	9.96E+01	1.49E−01	1.01E+01	5.25E+01	9.95E+01	1.43E−01	4.00E−02	4.00E−02
F5	1.25E+10	3.15E+08	9.97E+03	2.01E+03	1.49E+10	3.06E+01	9.21E+02	9.69E+02
F6	2.73E+06	4.56E+04	3.93E+02	2.35E+02	2.04E+06	2.46E+05	2.01E+02	6.21E+02
F7	1.96E+05	6.19E+03	1.83E+03	5.79E+04	2.27E+05	3.52E+04	2.18E−03	2.18E−03
F8	−9.00E+04	7.20E+03	−6.44E+04	1.92E+04	−3.72E+04	1.23E+03	−2.07E+05	3.24E+04
F9	1.56E+04	1.94E+02	3.21E+05	3.65E+04	1.50E+04	1.79E+02	3.99E−02	2.64E−02
F10	2.04E+01	2.16E−01	5.09E+01	1.94E+01	2.07E+01	1.06E−01	1.15E−02	2.36E−03
F11	2.47E+04	4.51E+02	1.07E+04	2.03E+04	1.85E+04	2.22E+03	6.68E−03	9.36E−03
F12	3.04E+10	9.72E+08	6.94E+03	1.90E+03	3.72E+10	7.67E+08	5.18E−01	5.65E−01
F13	5.62E+10	1.76E+09	9.98E+04	4.31E+05	6.66E+10	2.26E+09	1.00E+02	1.00E+02

in Fig. 15. Three design variables need to be taken into account: wire diameter (d), mean coil diameter (D), and the number of active coils (N). The mathematical representation of this problem can be found in its original paper.

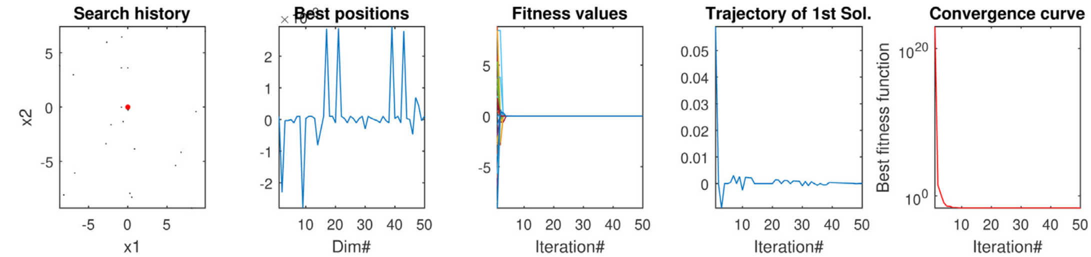
The proposed AOA is applied for solving this engineering problem (Tension/compression spring design) and compared with a mathematical technique and other well-known optimization algorithms, which are published in the literature as shown in Table 18. The obtained results of the proposed AOA are compared with the literature studies in Table 18. It can be observed that AOA outperformed all other comparative algorithms. Fig. 16 shows the qualitative results for the tension/compression spring design problem.

Table 13
Ranking-based Friedman test for the comparative algorithms using the test functions (F1–F13), with 30, 100, 500, and 1000 dimensions.

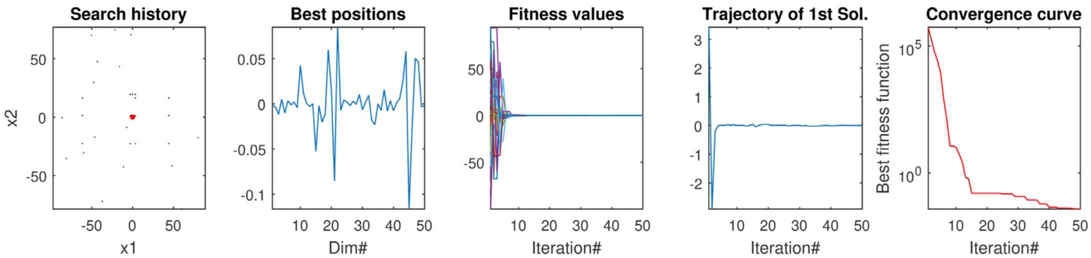
Dim	F	GA	PSO	BBO	FPA	GWO	BAT	FA	CS	MFO	GSA	DE	AOA
30	F1	9	10	6	12	1	11	5	3	8	7	4	2
	F2	8	11	3	10	2	12	6	5	9	7	4	1
	F3	8	10	6	4	2	12	5	3	7	11	9	1
	F4	9	8	7	6	1	12	4	3	10	11	5	2
	F5	9	11	7	10	2	12	4	3	8	6	5	1
	F6	7	11	6	8	5	12	4	3	9	10	2	1
	F7	8	11	3	9	2	12	5	6	10	7	4	1
	F8	1	12	2	6	7	10	8	11	4	9	5	3
	F9	4	12	12	10	3	11	7	5	9	6	8	2
	F10	9	11	6	8	2	12	5	4	10	7	3	1
	F11	8	11	7	9	4	12	3	2	10	6	5	1
	F12	8	11	7	10	5	12	4	3	9	6	2	1
	F13	8	11	6	9	5	12	1	2	10	7	3	4
100	F1	9	11	6	8	1	12	3	4	10	5	7	2
	F2	10	11	6	7	2	12	5	4	9	3	8	1
	F3	9	10	7	4	3	12	6	2	8	5	11	1
	F4	9	7	8	6	4	11	5	3	10	2	12	1
	F5	9	11	6	7	1	12	4	3	10	5	8	1
	F6	9	11	6	8	4	12	1	3	10	2	7	5
	F7	8	11	5	6	1	10	3	4	9	2	7	1
	F8	2	11	3	8	7	12	5	1	4	6	10	9
	F9	7	12	2	9	4	8	6	5	10	3	11	1
	F10	8	9	12	6	1	10	5	4	11	3	7	2
	F11	9	11	6	8	2	12	3	1	10	4	7	5
	F12	8	11	6	7	3	12	5	2	10	4	9	1
	F13	8	11	6	7	3	12	4	2	10	5	9	1
500	F1	7	8	6	5	1	11	4	3	10	12	9	2
	F2	8	11	6	5	2	12	7	4	9	3	10	1
	F3	9	10	7	5	4	12	6	2	8	3	11	1
	F4	10	7	9	4	6	11	5	2	12	3	8	1
	F5	8	9	7	6	1	12	5	3	11	4	10	2
	F6	8	9	7	6	3	12	4	2	11	5	10	1
	F7	8	9	7	5	2	10	6	4	12	3	11	1
	F8	4	11	3	8	7	12	5	1	6	2	9	10
	F9	6	10	3	8	2	9	5	4	11	7	12	1
	F10	8	9	7	5	1	11	6	3	10	4	12	1
	F11	8	9	7	6	2	12	5	3	11	4	10	1
	F12	9	10	8	7	3	12	6	2	11	5	4	1
	F13	9	8	7	6	2	12	5	3	10	4	11	1
1000	F1	8	8	7	5	2	12	6	3	11	4	10	1
	F2	7	8	6	4	2	8	8	3	8	5	8	1
	F3	9	10	7	5	4	12	6	2	8	3	11	1
	F4	9	7	8	4	6	10	5	2	12	3	11	1
	F5	9	8	7	5	2	11	6	3	10	4	12	1
	F6	9	8	7	5	2	12	6	3	11	4	10	1
	F7	8	9	7	4	2	10	6	3	11	5	12	1
	F8	3	11	1	9	7	12	4	5	6	8	10	2
	F9	6	8	3	7	2	9	5	4	11	12	10	1
	F10	7	8	6	4	2	10	5	3	9	12	10	1
	F11	9	8	6	4	3	12	5	2	11	7	10	1
	F12	4	9	8	6	3	11	7	2	10	5	12	1
	F13	1	9	8	6	3	11	7	4	10	5	12	2
Mean	Rank	7.53	9.76	6.19	6.65	2.94	11.28	5.01	3.19	9.50	5.48	8.40	1.75
Final	Ranking	8	11	6	7	2	12	4	3	10	5	9	1



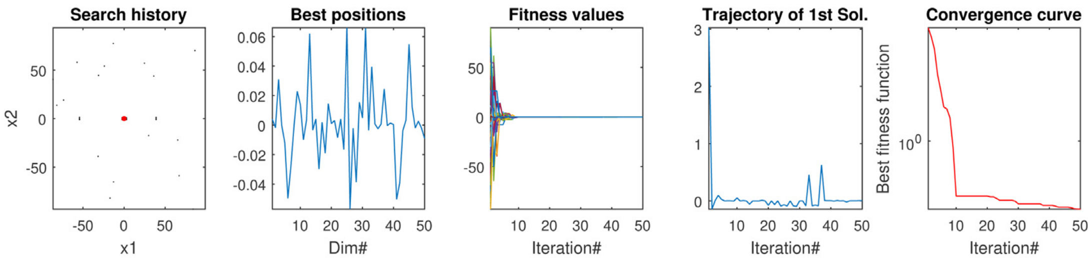
(a) F1



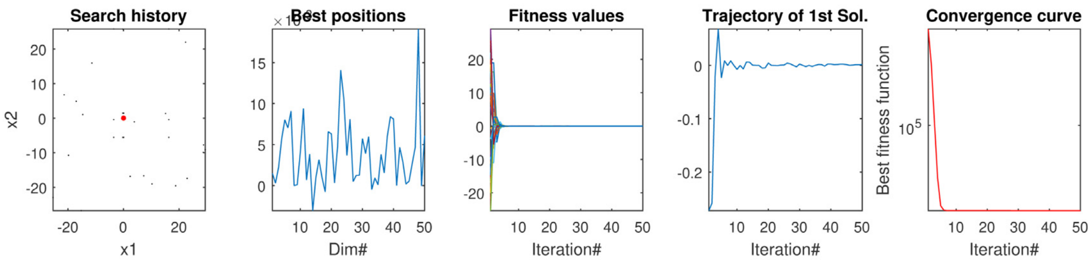
(b) F2



(c) F3

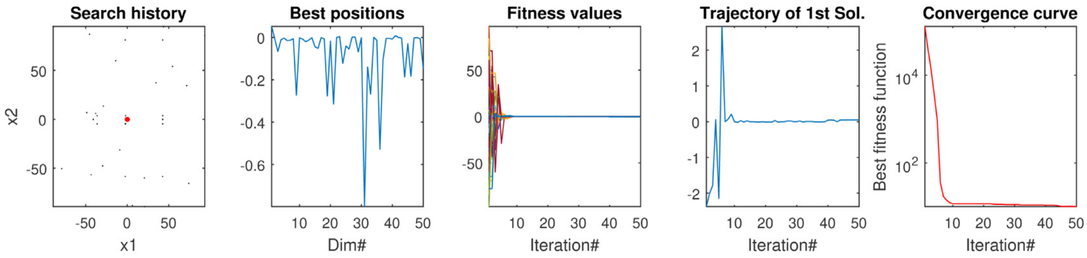


(d) F4

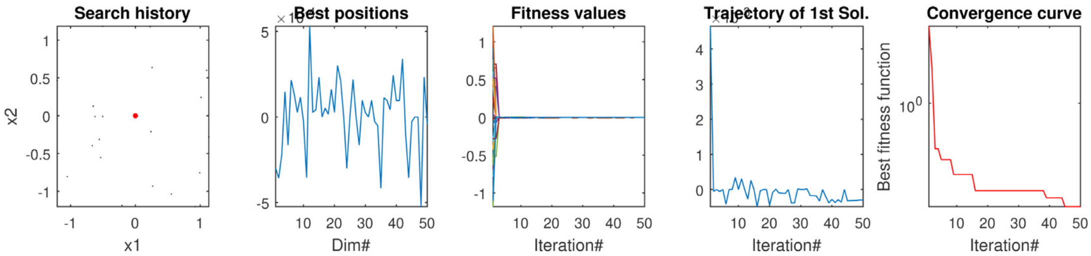


(e) F5

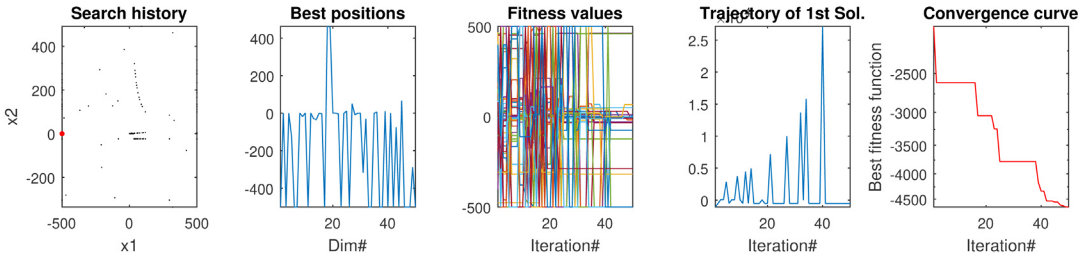
Fig. 10. Qualitative results for the studied problems from F1–F13.



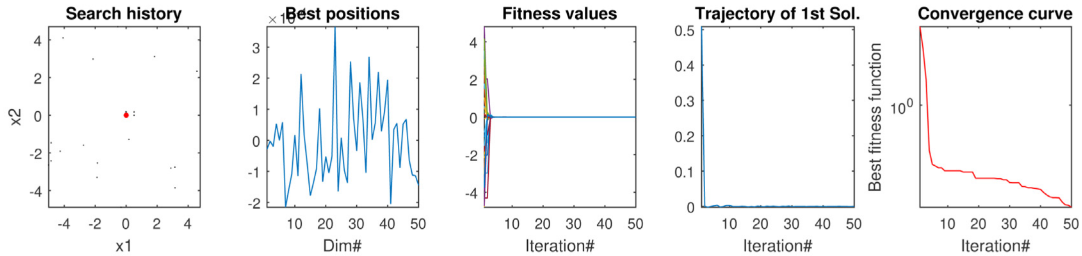
(f) F6



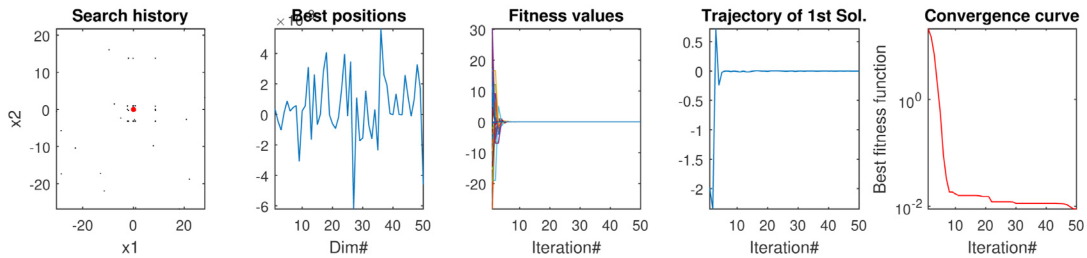
(g) F7



(h) F8

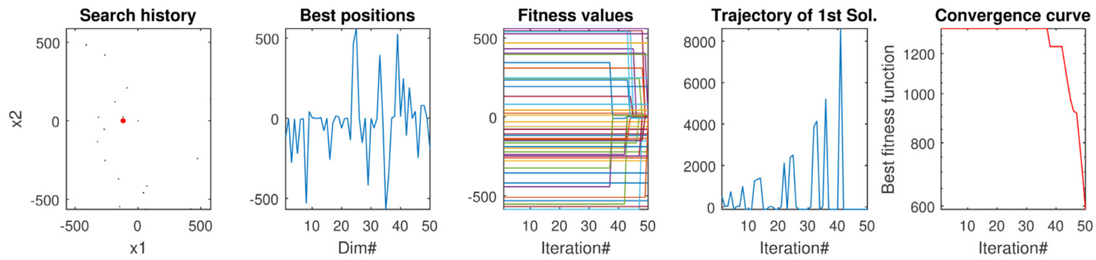


(i) F9

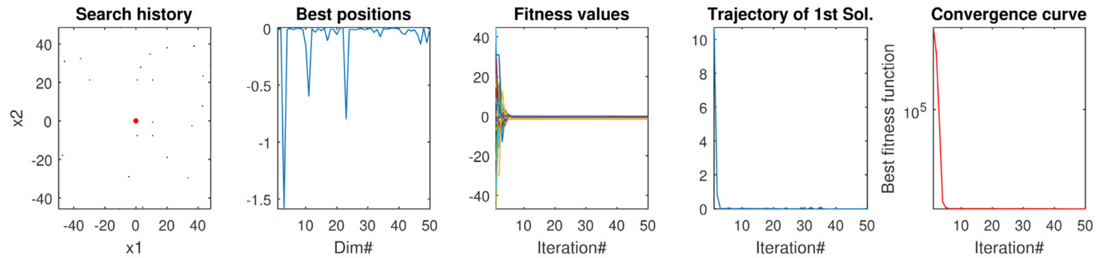


(j) F10

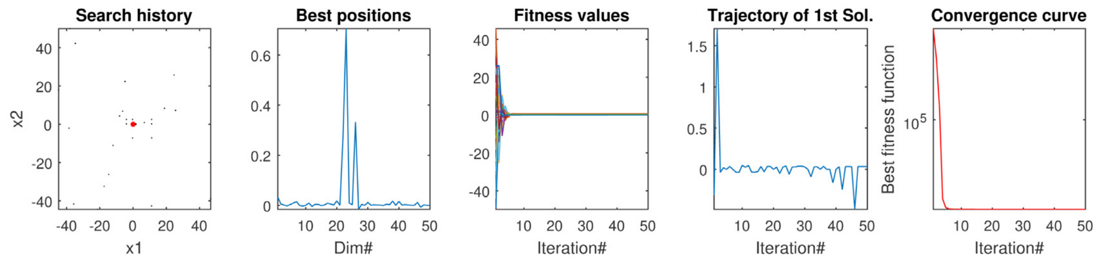
Fig. 10. (*continued*).



(k) F11



(l) F12



(m) F13

Fig. 10. (*continued*).

3.2.3. Pressure vessel design problem

The main objective of the pressure vessel design problem is to find the overall cost of the cylindrical pressure vessel to satisfy its design constraints: forming, material, and welding as shown in Fig. 17. Both edges of the vessel are capped while the top has a hemispherical shape. Four design variables need to be taken into account in the optimization operations to satisfy its four constraints: the inner radius (R), the thickness of the head (T_h), thickness of the shell (T_s), and the length of the cylindrical part without examining the head (L). The mathematical representation of this problem can be found in its original paper.

The obtained results by the AOA for solving the Pressure vessel design problem are compared with other several optimization algorithms as shown in Table 19. From this table, we conclude that the results of AOA are better than almost all other comparative algorithms. It can be observed that the AOA outperformed almost all other optimization algorithms. Fig. 18 shows the qualitative results for the pressure vessel design problem.

3.2.4. 3-bar truss design problem

This engineering design problem is to create a truss with three bars to decrease its weight. This problem has a very restricted search space [39,47]. The structural parameters in this problem are shown in Fig. 19. The mathematical representation of this problem can be found in its original paper.

The results of AOA when solving 3-bar truss design problem are shown in Table 20. The obtained results by the AOA for solving this problem (3-bar truss design problem) are compared with other several optimization algorithms published in the literature. It can be observed that AOA is a competitive algorithm compared to well-known optimization techniques published in the literature. Fig. 20 shows the qualitative results for the 3-bar truss design problem.

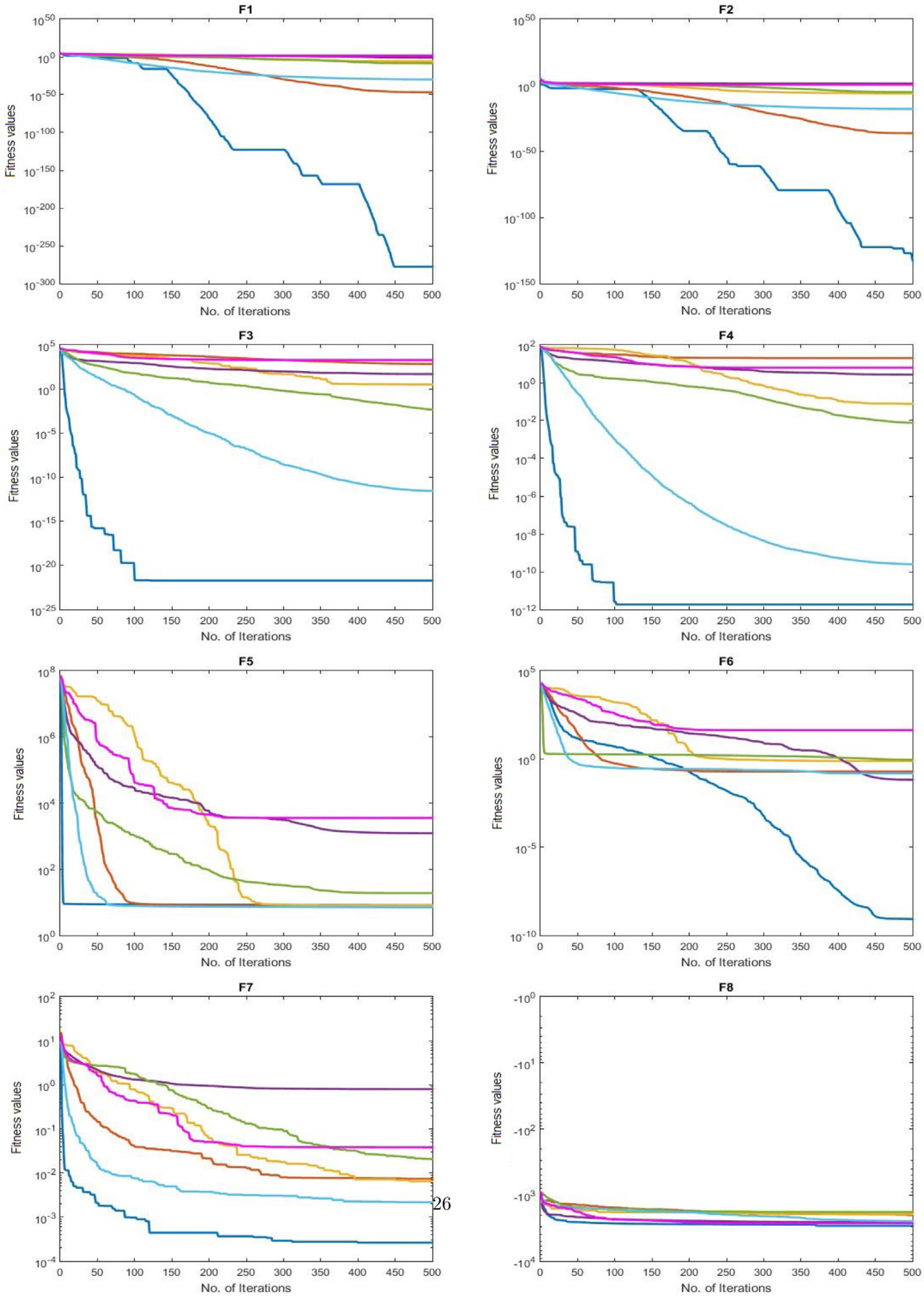


Fig. 11. Convergence behavior results for the studied problems from F1–F13.

3.2.5. Speed reducer problem

This main objective of the Speed reducer design problem [48], which is considered a discrete problem, is to find the minimum weight of the speed reducer to satisfy its four design constraints: bending stress of the gear teeth,

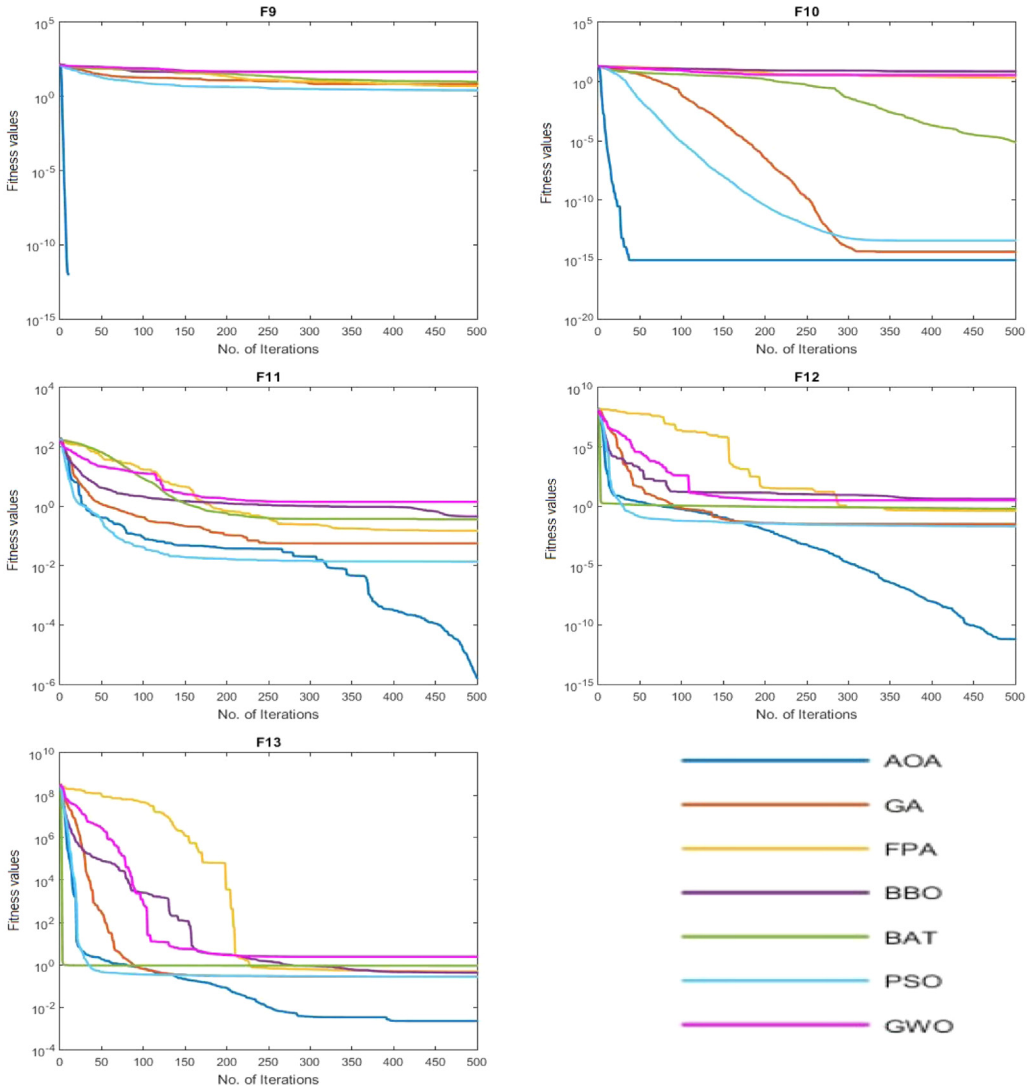


Fig. 11. (*continued*).

covering stress, transverse deflections of the shafts, and stresses in the shafts, as shown in Fig. 21. Consequently, there are one discrete and six continuous variables observed. Here, $x1$ is the face width, $x2$ is the module of teeth, and $x3$ is a discrete design variable that presents the teeth in the pinion. Similarly, $x4$ is the length of the first shaft between bearings, and $x5$ is the length of the second shaft between bearings. The sixth and seven design variables ($x6$ and $x7$) are the diameters of the first and second shaft, respectively. The mathematical representation of this problem can be found in its original paper.

The obtained results by the AOA for solving the speed reducer design problem are compared with other several optimization algorithms published in the literature as shown in Table 21. It is clear that the results of AOA are better than all comparative algorithms. AOA can give very competitive results in addressing this problem. Moreover, Fig. 22 shows the qualitative results for the speed reducer design problem.

Table 14
Comparison results of average running time (seconds) over 30 independent runs for test problems (F1–F13) with 1000 dimensions.

F	Metric	PSO	GA	FPA	BBO	BAT	GWO	CS	FA	GSA	MFO	DE	AOA
F1	Ave	8.29E+01	8.27E+01	2.13E+00	1.17E+02	1.60E+00	4.47E+00	5.47E+00	5.62E+00	2.21E+00	3.23E+00	2.38E+00	**1.55E+00**
	Std	4.04E+00	5.13E+00	2.62E−01	6.04E+00	2.08E−01	2.64E−01	4.00E−01	4.42E−01	3.62E−01	2.06E−01	2.70E−01	2.06E−01
	Rank	11	10	3	12	2	7	8	9	4	6	5	1
F2	Ave	8.28E+01	8.41E+01	2.09E+00	1.16E+02	1.61E+00	4.37E+00	5.50E+00	2.57E+00	1.99E+00	3.25E+00	2.28E+00	**1.60E+00**
	Std	4.08E+00	4.65E+00	8.64E−02	6.28E+00	1.02E−01	1.29E−01	3.48E−01	3.93E−01	1.19E−01	1.56E−01	1.16E−01	1.02E−01
	Rank	10	11	4	12	2	8	9	6	3	7	5	1
F3	Ave	1.30E+02	1.32E+02	5.10E+01	1.65E+02	5.23E+01	5.20E+01	1.02E+02	**3.70E+01**	9.76E+01	5.11E+01	5.04E+01	3.80E+01
	Std	5.73E+00	5.68E+00	2.01E+00	7.56E+00	2.25E+00	1.93E+00	3.73E+00	1.49E+00	3.87E+00	2.00E+00	1.98E+00	1.43E+00
	Rank	10	11	4	12	7	6	9	1	8	5	3	2
F4	Ave	8.24E+01	8.14E+01	1.90E+00	1.18E+02	1.44E+00	4.27E+00	5.14E+00	5.43E+00	1.87E+00	3.14E+00	2.21E+00	**1.40E+00**
	Std	3.91E+00	3.73E+00	5.83E−02	5.48E+00	1.02E−01	1.36E−01	2.33E−01	2.76E−01	1.05E−01	9.28E−02	8.73E−02	1.12E−01
	Rank	11	10	4	12	2	7	8	9	3	6	5	1
F5	Ave	8.33E+01	8.16E+01	2.04E+00	1.17E+02	**1.65E+00**	4.46E+00	5.49E+00	5.61E+00	2.23E+00	3.31E+00	2.38E+00	1.70E+00
	Std	4.36E+00	4.13E+00	7.79E−02	5.91E+00	1.16E−01	1.39E−01	2.74E−01	3.01E−01	1.09E−01	1.27E−01	1.30E−01	1.12E−01
	Rank	11	10	3	12	1	7	8	9	4	6	5	2
F6	Ave	8.26E+01	8.08E+01	1.88E+00	1.17E+02	1.47E+00	4.29E+00	5.17E+00	5.51E+00	1.89E+00	3.13E+00	2.19E+00	**1.44E+00**
	Std	3.95E+00	3.96E+00	4.98E−02	5.69E+00	1.03E−01	1.07E−01	2.35E−01	2.87E−01	9.33E−02	1.00E−01	1.02E−01	1.02E−01
	Rank	11	10	3	12	2	7	8	9	4	6	5	1
F7	Ave	8.52E+01	8.26E+01	4.79E+00	1.18E+02	4.22E+00	7.08E+00	1.08E+01	6.89E+00	7.23E+00	5.83E+00	4.95E+00	**3.99E+00**
	Std	3.94E+00	4.56E+00	1.02E−01	6.10E+00	8.98E−02	7.56E−02	3.86E−01	2.02E−01	1.31E−01	1.01E−01	1.43E−01	8.16E−02
	Rank	11	10	3	12	2	7	9	6	8	5	4	1
F8	Ave	8.36E+01	8.47E+01	3.18E+00	1.18E+02	**2.45E+00**	5.21E+00	7.69E+00	6.04E+00	3.84E+00	4.05E+00	3.23E+00	2.56E+00
	Std	3.80E+00	3.68E+00	4.73E−01	5.52E+00	2.88E−01	1.78E−01	3.86E−01	2.69E−01	4.12E−01	1.20E−01	8.69E−02	3.13E−01
	Rank	10	11	3	12	1	7	9	8	5	6	4	2
F9	Ave	8.33E+01	8.09E+01	2.84E+00	1.15E+02	2.33E+00	4.72E+00	6.90E+00	5.89E+00	2.70E+00	3.94E+00	3.20E+00	**2.30E+00**
	Std	3.88E+00	3.59E+00	4.30E−01	5.94E+00	2.88E−01	1.19E−01	3.34E−01	2.55E−01	4.71E−01	1.26E−01	5.50E−01	2.71E−01
	Rank	11	10	4	12	2	7	9	8	3	6	5	1
F10	Ave	8.36E+01	8.24E+01	2.96E+00	1.17E+02	2.46E+00	4.80E+00	6.56E+00	5.98E+00	2.84E+00	4.04E+00	3.41E+00	**2.42E+00**
	Std	3.99E+00	4.02E+00	3.74E−01	5.90E+00	4.67E−01	1.14E−01	3.51E−01	2.91E−01	5.39E−01	1.21E−01	3.01E−01	4.35E−01
	Rank	11	10	4	12	2	7	9	8	3	6	5	1
F11	Ave	8.38E+01	8.23E+01	3.16E+00	1.18E+02	**2.61E+00**	4.95E+00	6.43E+00	6.03E+00	3.03E+00	4.22E+00	3.38E+00	2.66E+00
	Std	3.97E+00	4.41E+00	5.50E−01	6.02E+00	3.95E−01	8.65E−02	3.01E−01	2.50E−01	3.95E−01	1.20E−01	9.95E−02	4.05E−01
	Rank	11	10	4	12	1	7	9	8	3	6	5	2
F12	Ave	8.85E+01	8.64E+01	9.09E+00	1.23E+02	8.66E+00	1.06E+01	1.90E+01	9.17E+00	1.53E+01	9.67E+00	9.14E+00	**8.56E+00**
	Std	4.42E+00	4.47E+00	1.39E+00	6.20E+00	1.47E+00	4.33E−01	3.53E+00	3.62E−01	2.54E+00	4.04E−01	1.14E+00	1.36E+00
	Rank	11	10	3	12	2	7	9	5	8	6	4	1
F13	Ave	8.90E+01	8.64E+01	9.28E+00	1.23E+02	**8.74E+00**	1.05E+01	1.83E+01	9.24E+00	1.46E+01	9.66E+00	9.34E+00	8.71E+00
	Std	4.20E+00	4.40E+00	1.50E+00	6.29E+00	1.38E+00	4.56E−01	7.75E−01	3.94E−01	2.24E+00	3.91E−01	1.24E+00	1.33E+00
	Rank	11	10	4	12	2	7	9	3	8	6	5	1
Mean Rank		10.769	10.230	3.538	12.000	2.153	7.000	8.692	6.846	4.923	5.923	4.615	1.307
Final Ranking		11	10	3	12	2	8	9	7	5	6	4	1

Table 15
Results of the test functions (F14-F29).

F	Metric	PSO	GA	FPA	BBO	BAT	GWO	CS	FA	TLBO	MFO	DE	AOA
F14	Ave	1.39E+00	9.98E−01	9.98E−01	9.98E−01	1.27E+01	4.17E+00	1.27E+01	3.51E+00	9.98E−01	2.74E+00	1.23E+00	9.98E−01
	Rank	7	1	1	1	11	10	11	9	1	8	6	1
	Std	4.60E−01	4.52E−16	2.00E−04	4.52E−16	6.96E+00	3.61E+00	1.81E−15	2.16E+00	4.52E−16	1.82E+00	9.23E−01	5.54E−01
F15	Ave	1.61E−03	3.33E−02	6.88E−04	1.66E−02	3.00E−02	6.24E−03	3.13E−04	1.01E−03	1.03E−03	2.35E−03	5.63E−04	3.12E−04
	Rank	7	12	4	10	11	9	2	5	6	8	3	1
	Std	4.60E−04	2.70E−02	1.55E−04	8.60E−03	3.33E−02	1.25E−02	2.99E−05	4.01E−04	3.66E−03	4.92E−03	2.81E−04	2.64E−04
F16	Ave	−1.03E+00	−3.78E−01	−1.03E+00	−8.30E−01	−6.87E−01	−1.03E+00	−1.03E+00	−1.03E+00	−1.03E+00	−1.03E+00	−1.03E+00	−1.03E+00
	Rank	1	12	1	10	11	1	1	1	1	1	1	1
	Std	2.95E−03	3.42E−01	6.78E−16	3.16E−01	8.18E−01	6.78E−16	6.78E−16	6.78E−16	6.78E−16	6.78E−16	6.78E−16	5.48E−05
F17	Ave	4.00E−01	5.24E−01	3.98E−01	5.49E−01	3.98E−01	3.98E−01	3.98E−01	3.98E−01	3.98E−01	3.98E−01	3.98E−01	3.98E−01
	Rank	10	11	1	12	1	1	1	1	1	1	1	1
	Std	1.39E−03	6.06E−02	1.69E−16	6.05E−02	1.58E−03	1.69E−16	1.69E−16	1.69E−16	1.69E−16	1.69E−16	1.69E−16	2.54E−06
F18	Ave	3.10E+00	3.00E+00	3.00E+00	3.00E+00	1.47E+01	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00
	Rank	11	1	1	1	12	1	1	1	1	1	1	1
	Std	7.60E−02	0.00E+00	0.00E+00	0.00E+00	2.21E+01	4.07E−05	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.00E−02
F19	Ave	−3.86E+00	−3.42E+00	−3.86E+00	−3.78E+00	−3.84E+00	−3.86E+00	−3.86E+00	−3.86E+00	−3.86E+00	−3.86E+00	−3.86E+00	−3.86E+00
	Rank	1	12	1	11	10	1	1	1	1	1	1	1
	Std	1.24E−03	3.03E−01	3.16E−15	1.26E−01	1.41E−01	3.14E−03	3.16E−15	3.16E−15	3.16E−15	1.44E−03	3.16E−15	4.29E−04
F20	Ave	−3.11E+00	−1.61E+00	−3.30E+00	−2.71E+00	−3.25E+00	−3.26E+00	−3.32E+00	−3.28E+00	−3.24E+00	−3.24E+00	−3.27E+00	−3.32E+00
	Rank	10	12	3	11	7	6	1	4	8	9	5	1
	Std	2.91E−02	4.60E−01	1.95E−02	3.58E−01	5.89E−02	6.43E−02	1.78E−15	6.36E−02	1.51E−01	6.42E−02	5.89E−02	1.25E+01
F21	Ave	−4.15E+00	−6.66E+00	−5.22E+00	−8.32E+00	−4.27E+00	−8.64E+00	−5.06E+00	−7.67E+00	−8.65E+00	−6.89E+00	−8.65E+00	−8.85E+00
	Rank	12	8	9	5	11	4	10	6	3	7	2	1
	Std	9.20E−01	3.73E+00	8.15E−03	2.88E+00	2.55E+00	2.56E+00	1.78E−15	3.51E+00	1.77E+00	3.18E+00	1.52E+00	1.25E+00
F22	Ave	−6.01E+00	−5.58E+00	−5.34E+00	−9.38E+00	−5.61E+00	−1.04E+01	−5.09E+00	−9.64E+00	−1.02E+01	−8.26E+00	−9.75E+00	−1.04E+01
	Rank	9	11	12	7	10	1	12	6	4	8	4	1
	Std	1.96E+00	2.61E+00	5.37E−02	2.60E+00	3.02E+00	6.78E−04	8.88E−16	2.29E+00	7.27E−03	3.08E+00	1.99E+00	2.21E+00
F23	Ave	−4.72E+00	−4.70E+00	−5.29E+00	−6.24E+00	−3.97E+00	−1.01E+01	−5.13E+00	−9.75E+00	−1.01E+01	−7.66E+00	−1.05E+01	−1.05E+01
	Rank	10	11	8	7	12	3	9	5	4	6	1	1
	Std	1.74E+00	3.26E+00	3.56E−01	3.78E+00	3.01E+00	1.72E+00	1.78E−15	2.35E+00	1.70E+00	3.58E+00	8.88E−15	1.02E+00
F24	Ave	7.68E+02	6.27E+02	5.19E+02	4.93E+02	1.29E+03	4.87E+02	4.69E+02	4.72E+02	6.13E+02	4.12E+02	4.31E+02	4.07E+02
	Rank	11	10	8	7	12	6	4	5	9	2	3	1
	Std	7.61E+01	1.01E+02	4.78E+01	1.03E+02	1.50E+02	1.43E+02	6.06E+01	2.52E+02	1.23E+02	6.84E+01	6.42E+01	8.13E+01
F25	Ave	1.18E+03	9.99E+02	1.02E+03	9.35E+02	1.46E+03	9.85E+02	9.10E+02	9.54E+02	9.67E+02	9.48E+02	9.18E+02	9.10E+02
	Rank	11	9	10	4	12	8	2	6	7	5	3	1
	Std	3.30E+01	2.94E+01	3.19E+01	9.61E+00	6.84E+01	3.00E+01	3.67E−02	1.17E+01	2.74E+01	2.71E+01	1.05E+00	1.21E+01
F26	Ave	1.18E+03	9.99E+02	1.02E+03	9.34E+02	1.48E+03	9.74E+02	9.10E+02	9.54E+02	9.84E+02	9.40E+02	9.17E+02	9.12E+02
	Rank	11	9	10	4	12	7	1	6	8	5	3	2
	Std	3.52E+01	2.53E+01	3.49E+01	8.25E+00	4.56E+01	2.25E+01	4.72E−02	1.41E+01	4.53E+01	2.17E+01	8.98E−01	3.14E+00
F27	Ave	1.20E+03	1.00E+03	1.01E+03	9.40E+02	1.48E+03	9.70E+02	9.10E+02	9.48E+02	9.79E+02	9.45E+02	9.17E+02	9.10E+02
	Rank	11	9	10	4	12	7	2	6	8	5	3	1
	Std	2.40E+01	2.67E+01	3.15E+01	2.31E+01	6.06E+01	1.95E+01	4.97E−02	1.12E+01	3.82E+01	2.68E+01	8.62E−01	3.43E+01

(*continued on next page*)

Table 15 (*continued*).

F	Metric	PSO	GA	FPA	BBO	BAT	GWO	CS	FA	TLBO	MFO	DE	AOA
F28	Ave	1.71E+03	1.51E+03	1.54E+03	1.07E+03	1.96E+03	1.34E+03	1.34E+03	1.02E+03	1.47E+03	1.46E+03	1.55E+03	9.81E+02
	Rank	11	8	9	3	12	4	5	2	7	6	10	1
	Std	3.52E+01	9.46E+01	4.29E+01	2.02E+02	5.85E+01	1.91E+02	1.34E+02	2.71E+02	2.69E+02	3.61E+01	9.64E+01	4.73E+01
F29	Ave	2.10E+03	1.94E+03	2.03E+03	1.90E+03	2.22E+03	1.91E+03	1.90E+03	1.99E+03	1.88E+03	1.88E+03	1.90E+03	1.62E+03
	Rank	11	8	10	5	12	7	6	9	3	2	4	1
	Std	2.97E+01	1.13E+01	3.03E+01	8.82E+00	3.55E+01	6.57E+00	1.86E+02	1.89E+01	3.49E+00	6.53E+00	4.20E+00	4.25E+00
Mean	Rank	8.9375	8.9375	6.0625	6.3125	10.4375	4.75	4.3125	4.5	4.375	4.625	3.125	1.125
Final	Ranking	10	10	8	9	12	7	3	5	4	6	2	1

Table 16
Ranking-based Wilcoxon signed rank test between AOA and other algorithms using the test functions (F1–F29).

F	GWO			BAT			FA			CS			MFO			GSA			DE		
	p-value	H	S	*p*-value	H	S	*p*-value	H	S	*p*-value	H	S	*p*-value	H	S	*p*-value	H	S	*p*-value	H	S
F1	6.24E−02	0	−	3.25E−03	1	+	6.24E+05	1	+	5.66E−05	1	+	6.32E−04	1	+	7.65E−07	1	+	3.54E−08	1	+
F2	3.65E−04	1	+	5.64E−06	1	+	4.67E−06	1	+	7.39E−07	1	+	2.64E−07	1	+	4.45E−05	1	+	8.45E−05	1	+
F3	2.34E−08	1	+	3.64E−08	1	+	4.56E−07	1	+	6.45E−08	1	+	3.95E−09	1	+	6.32E−09	1	+	3.69E−08	1	+
F4	4.25E−05	1	+	4.25E−08	1	+	5.65E−08	1	+	2.54E−05	1	+	5.65E−08	1	+	7.25E−05	1	+	2.85E−05	1	+
F5	1.00E+00	0	=	4.34E−06	1	+	4.36E−03	1	+	8.65E−04	1	+	6.59E−04	1	+	3.25E−06	1	+	4.65E−05	1	+
F6	6.28E−01	0	−	4.75E−07	1	+	6.65E−02	0	−	2.64E−01	0	−	2.42E−06	1	+	6.65E−02	0	−	3.12E−06	1	+
F7	1.00E+00	0	=	4.15E−05	1	+	3.02E−04	1	+	6.58E−05	1	+	1.68E−05	1	+	6.32E−05	1	+	4.65E−06	1	+
F8	2.65E−01	0	−	6.32E−04	1	+	7.45E−02	0	−	5.14E−02	0	−	2.78E−01	0	−	8.27E−01	0	−	4.01E−05	1	+
F9	6.25E−04	1	+	6.32E−04	1	+	7.65-E05	1	+	2.35E−05	1	+	3.14E−09	1	+	8.61E−05	1	+	6.14E−05	1	+
F10	4.65E−02	0	−	4.34E−05	1	+	6.12E−05	1	+	4.19E−09	1	+	6.32E−06	1	+	3.32E−04	1	+	3.21-E03	1	+
F11	1.34E−02	0	−	6.34E−05	1	+	4.64E−02	0	−	9.47E−01	0	−	5.37E−05	0	+	2.58E−02	0	−	6.24E−05	1	+
F12	3.25E−06	1	+	5.15E−05	1	+	6.45E−03	1	+	4.56E−05	1	+	6.55E−05	1	+	7.46E−04	1	+	4.64E−05	1	+
F13	4.56E−05	1	+	6.67E−09	1	+	4.45E−07	1	+	6.47E−06	1	+	6.46E−06	1	+	4.25E−06	1	+	5.78E−06	1	+
F14	2.35E−06	1	+	4.54E−04	1	+	5.54E−04	1	+	8.58E−04	1	+	6.45E−06	1	+	4.86E−07	1	+	7.65E−09	1	+
F15	7.35E−06	1	+	5.25E−08	1	+	5.45E−06	1	+	4.64E−03	1	+	2.18E−04	1	+	6.47E−04	1	+	4.64E−04	1	+
F16	6.75E−06	1	+	5.48E−06	1	+	5.51E−06	1	+	4.67E−03	1	+	6.48E−04	1	+	2.22E−05	1	+	4.97E−08	1	+
F17	4.28E−05	1	+	8.54E−06	1	+	1.97E−03	1	+	2.75E−07	1	+	3.52E−06	1	+	5.46E−06	1	+	4.54E−09	1	+
F18	6.48E−06	1	+	8.65E−05	1	+	4.65E−05	1	+	6.45E−04	1	+	4.64E−07	1	+	4.54E−03	1	+	4.54E−05	1	+
F19	3.64E−05	1	+	2.48E−06	1	+	7.38E−08	1	+	8.65E−04	1	+	2.23E−09	1	+	7.12E−06	1	+	6.27E−05	1	+
F20	3.07E−08	1	+	2.14E−09	1	+	3.27E−07	1	+	6.45E−06	1	+	5.79E−06	1	+	6.46E−08	1	+	6.43E−06	1	+
F21	4.54E−03	1	+	3.14E−04	1	+	5.24E−05	1	+	7.54E−02	0	−	3.95E−03	1	+	4.02E−02	1	+	5.34E−05	1	+
F22	5.55E−08	1	+	3.15E−06	1	+	4.64E−09	1	+	6.55E−10	1	+	7.45E−05	1	+	3.55E−06	1	+	6.63E−06	1	+
F23	5.37E−03	1	+	3.87E−06	1	+	7.45E−05	1	+	4.45E−06	1	+	5.46E−05	1	+	3.64E−08	1	+	4.44E−06	1	+
F24	6.48E−03	1	+	6.38E−05	1	+	8.33E−06	1	+	4.68E−09	1	+	5.98E−06	1	+	1.90E−07	1	+	6.16E−06	1	+
F25	3.25E−08	1	+	7.16E−05	1	+	6.77E−06	1	+	3.64E−04	1	+	2.18E−04	1	+	6.50E−08	1	+	3.14E−07	1	+
F26	5.41E−06	1	+	3.91E−06	1	+	4.84E−06	1	+	2.62E−06	1	+	3.41E−05	1	+	3.14E−06	1	+	7.32E−07	1	+
F27	6.25E−07	1	+	3.46E−07	1	+	3.25E−06	1	+	6.12E−08	1	+	2.71E−06	1	+	1.23E−06	1	+	6.18E−07	1	+
F28	4.61E−06	1	+	5.23E−06	1	+	7.27E−06	1	+	6.32E−05	1	+	5.49E−07	1	+	1.46E−09	1	+	2.34E−08	1	+
F29	1.64E−06	1	+	3.33E−10	1	+	6.48E−09	1	+	6.38E−08	1	+	7.26E−08	1	+	1.85E−07	1	+	5.64E−08	1	+
(W|L|T)	(22|5|2)			(29|0|0)			(27|2|0)			(25|4|0)			(28|1|0)			(26|3|0)			(29|0|0)		

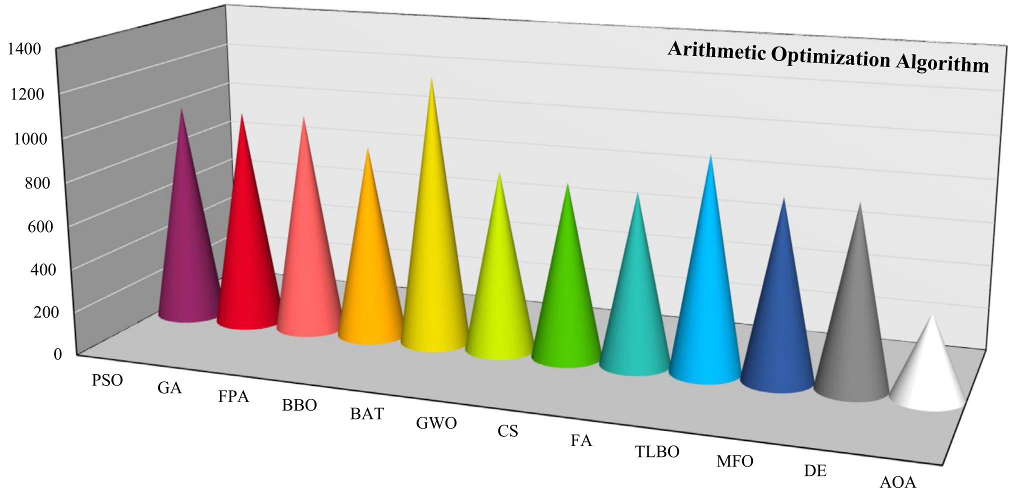


Fig. 12. Average fitness values of the test functions (F14–F29).

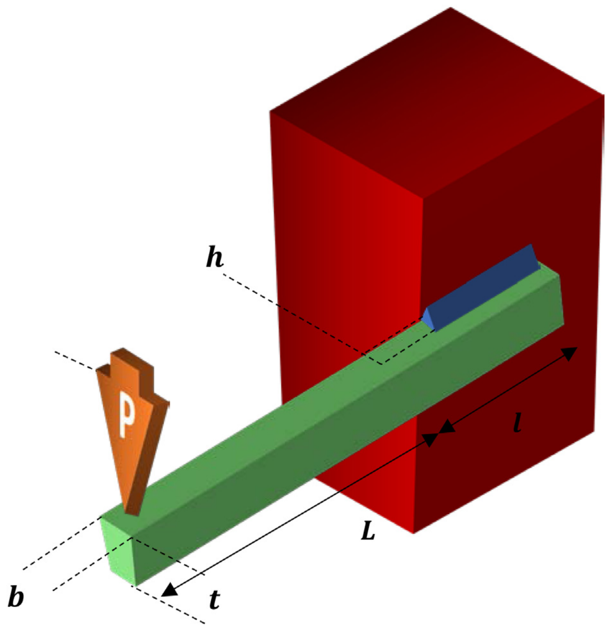


Fig. 13. Welded beam design problem.

Taken together, the results of this work evidently showed that the proposed AOA algorithm can be considered as a reliable alternative to the existing optimization algorithms. The mechanisms proposed allow this algorithm to show exploratory and exploitative behaviors when solving a wide range of problems.

4. Conclusion and potential future researches

In this paper, from the behavior of Arithmetic operators in mathematical calculations, a novel meta-heuristic optimization algorithm, the Arithmetic Optimization Algorithm (AOA), is proposed. Counter to most of the well-known optimization algorithms, AOA has an easy and straightforward implementation, according to its mathematical presentation, to adapt to tackle new optimization problems. It does not need to adjust many parameters except the

Table 17
Results of the comparative algorithms for solving the welded beam design problem.

Algorithm	Optimal values for variables				Optimal cost
	h	l	t	b	
SIMPLEX [49]	0.2792	5.6256	7.7512	0.2796	2.5307
DAVID [49]	0.2434	6.2552	8.2915	0.2444	2.3841
APPROX [49]	0.2444	6.2189	8.2915	0.2444	2.3815
GA [50]	0.2489	6.1730	8.1789	0.2533	2.4300
HS [51]	0.2442	6.2231	8.2915	0.2400	2.3807
CSCA [52]	0.203137	3.542998	9.033498	0.206179	1.733461
CPSO [53]	0.202369	3.544214	9.04821	0.205723	1.72802
RO [54]	0.203687	3.528467	9.004233	0.207241	1.735344
WOA [55]	0.205396	3.484293	9.037426	0.206276	1.730499
GSA [26]	0.182129	3.856979	10.000	0.202376	1.87995
MVO [26]	0.205463	3.473193	9.044502	0.205695	1.72645
OBSCA [56]	0.230824	3.069152	8.988479	0.208795	1.722315
AOA	0.194475	2.57092	10.000	0.201827	1.7164

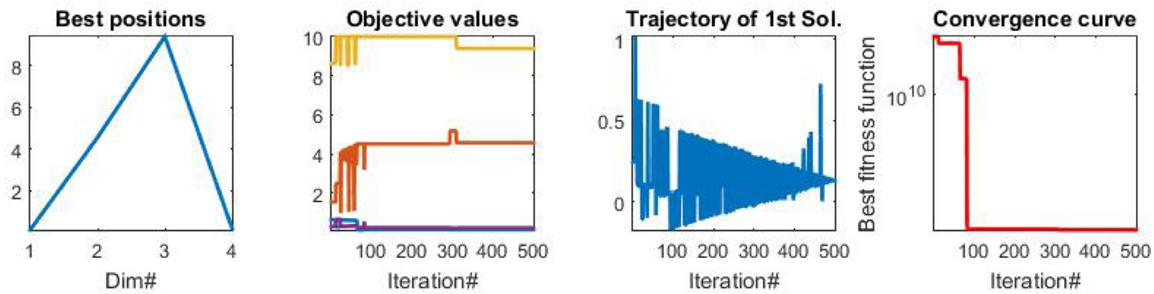


Fig. 14. Qualitative results for the welded beam design problem.

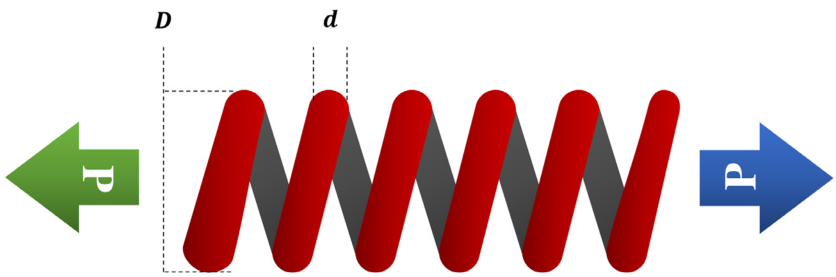


Fig. 15. Tension/compression spring design problem:.

population size and stopping criterion, which are standard parameters in all optimization algorithms. The random and adaptive parameters also expedited the divergence and convergence of the search solutions in the AOA.

Comprehensive experiments are conducted to validate the performance of the proposed AOA. Firstly, a set of twenty-nine well-known benchmark test functions, including unimodal, multimodal, composite functions, and hybrid composition functions are used to examine exploration, exploitation, local optima escape, and convergence behavior of the proposed AOA. Secondly, AOA is tested to solve some benchmark test functions with two-dimensional space. Various performance metrics (search story, trajectory, the average of fitness values, and the best-obtained solution during the optimization process) are applied to observe and confirm the performance of the proposed AOA qualitatively. Statistical ranking tests have been conducted to confirm the significant improvement of the proposed AOA over benchmark test functions statistically. Finally, the proposed AOA is also used to solve five real-life engineering design problems (welded beam design problem, tension/compression spring design problem,

Table 18
Results of the comparative algorithms for solving the tension/compression spring design problem.

Algorithm	Optimal values for variables			Optimal weight
	d	*D*	*N*	
CC [57]	70.050000	0.315900	14.250000	0.0128334
GA [58]	0.051480	0.351661	11.632201	0.01270478
HS [59]	0.051154	0.349871	12.076432	0.0126706
CSCA [52]	0.051609	0.354714	11.410831	0.0126702
PSO [53]	0.051728	0.357644	11.244543	0.0126747
CPSO [53]	0.051728	0.357644	11.244543	0.0126747
ES [60]	0.051643	0.355360	11.397926	0.012698
RO [54]	0.051370	0.349096	11.76279	0.0126788
WOA [55]	0.051207	0.345215	12.004032	0.0126763
GSA [26]	0.050276	0.323680	13.525410	0.0127022
MVO [26]	0.05251	0.37602	10.33513	0.012790
OBSCA [56]	0.05230	0.31728	12.54854	0.012625
AOA	0.0500	0.349809	11.8637	0.012124

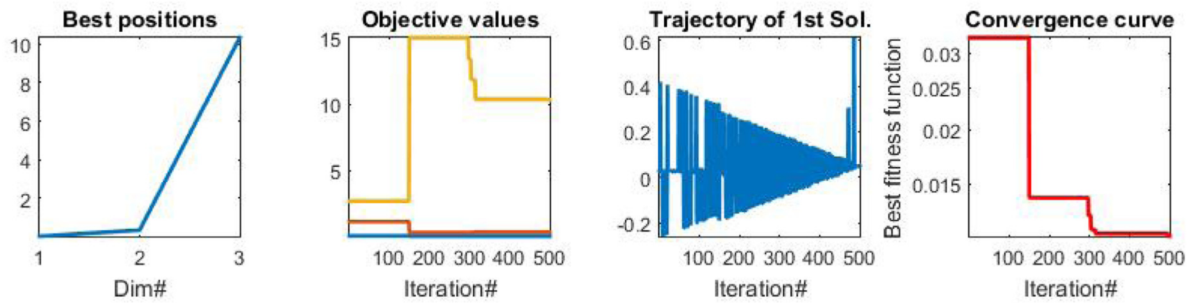


Fig. 16. Qualitative results for the tension/compression spring design.

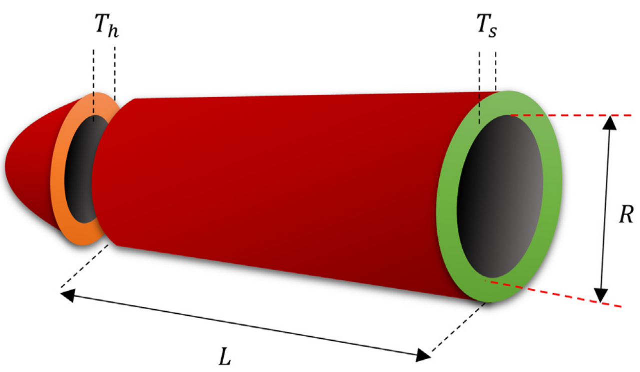


Fig. 17. Pressure vessel design problem.

pressure vessel design problem, 3-bar truss design problem, and speed reducer design problem) to test and confirm its performance. According to the obtained results, the proposed AOA can find better solutions for most of the examined problems compared to other well-known optimization algorithms in regard to solutions' quality and computational performance. Moreover, the results of the proposed AOA also sufficiently prove its superiority on the ability to avert trapping of the local optima. Consequently, we achieved the intended goals of proposing a new algorithm in this paper.

We have proposed the AOA algorithm with a simple yet effective framework and a minimum number operators to build the foundations of this algorithm. We will leave exploring other arithmetic and evolutionary operators (e.g.

Table 19
Results of the comparative algorithms for solving the pressure vessel design problem.

Algorithm	Optimal values for variables				Optimal cost
	T_s	T_h	R	L	
Branch-bound [61]	1.125	0.625	48.97	106.72	7982.5
GA [58]	0.81250	0.43750	42.097398	176.65405	6059.94634
HS [59]	1.125000	0.625000	58.29015	43.69268	7197.730
CSCA [52]	0.8125	0.4375	42.098411	176.63769	6059.7340
PSO-SCA [62]	0.8125	0.4375	42.098446	176.6366	6059.71433
CPSO [53]	0.8125	0.4375	42.091266	176.7465	6061.0777
HPSO [63]	0.8125	0.4375	42.0984	176.6366	6059.7143
ES [60]	0.8125	0.4375	42.098087	176.640518	6059.74560
ACO [64]	0.812500	0.437500	42.098353	176.637751	6059.7258
WOA [55]	0.812500	0.437500	42.0982699	176.638998	6059.7410
GSA [25]	1.125	0.625	55.9886598	84.4542025	8538.8359
MVO [26]	0.8125	0.4375	42.090738	176.73869	6060.8066
AOA	0.8303737	0.4162057	42.75127	169.3454	6048.7844

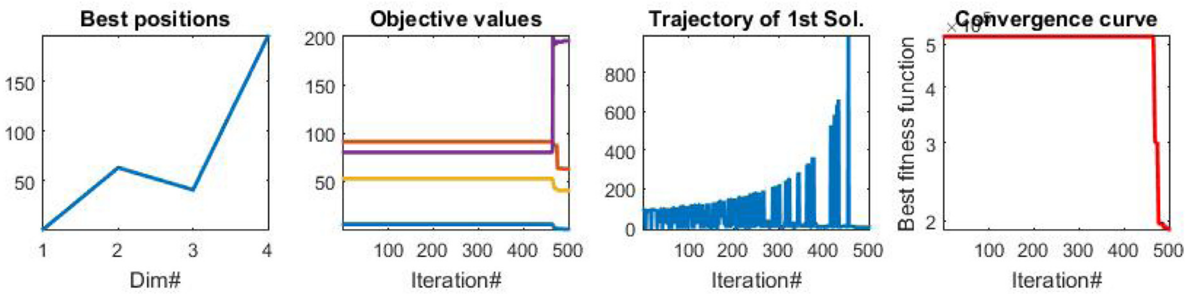


Fig. 18. Qualitative results for the pressure vessel design problem.

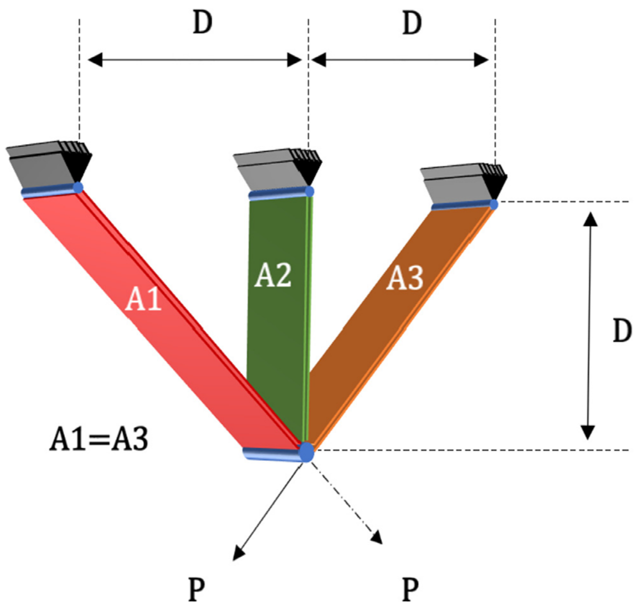


Fig. 19. 3-bar truss design problem.

Table 20
Results of the comparative algorithms for solving the 3-bar truss design problem.

Algorithm	Optimal values for variables		Optimal weight
	x_1	x_2	
DEDS [65]	0.78867513	0.40824828	263.89584
SSA [20]	0.78866541	0.408275784	263.89584
MBA [47]	0.7885650	0.4085597	263.89585
PSO-DE [62]	0.7886751	0.4082482	263.89584
Tsa [66]	0.788	0.408	263.68
Ray and Sain [67]	0.795	0.395	264.3
CS [39]	0.78867	0.40902	263.9716
AOA	0.79369	0.39426	263.9154

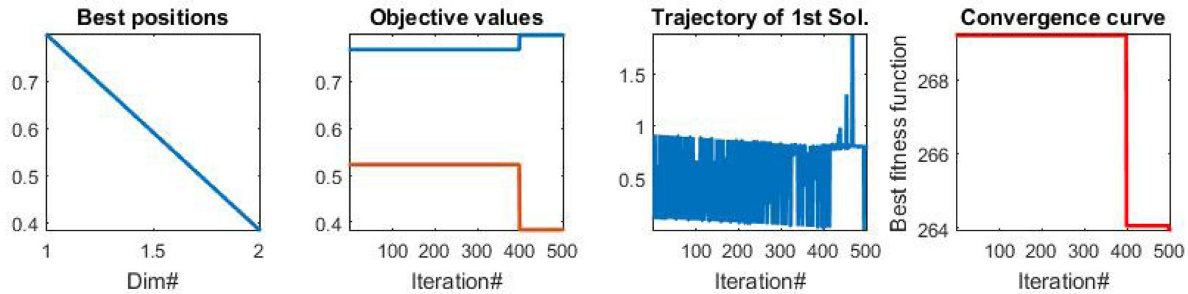


Fig. 20. Qualitative results for the 3-bar truss design problem.

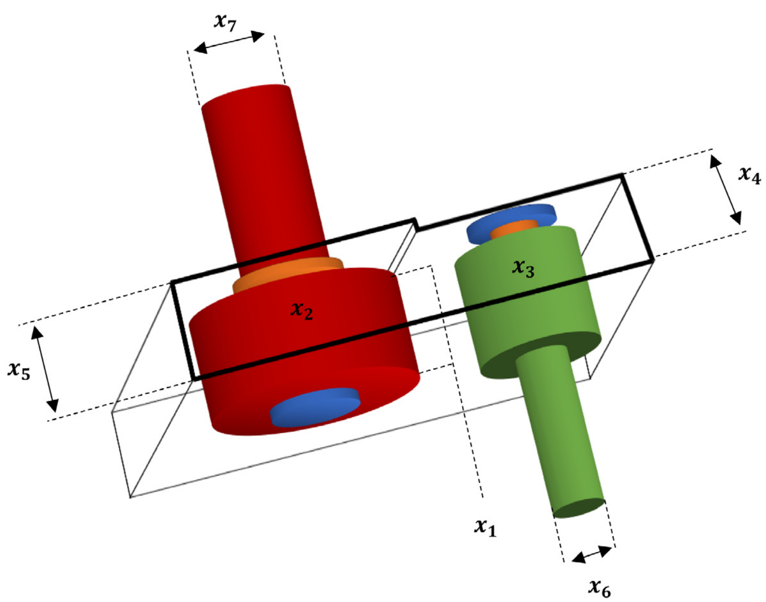


Fig. 21. Construction of a speed reducer.

mutation and crossover, multi-swarm composition, evolutionary updating composition, and chaotic maps) to future works.

In addition, other improved versions of the proposed AOA can be proposed to solve optimization problems with binary, discrete, and multiple objectives, respectively. Levy flight, disruption, mutation, and opposition-based learning can be combined with AOA for enhancing its performance. The AOA algorithm can be hybridized with

Table 21
Results of the comparative algorithms for solving the speed reducer design problem.

Algorithm	Optimal values for variables							Optimal weight
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
MFO [40]	3.497455	0.700	17	7.82775	7.712457	3.351787	5.286352	2998.94083
WSA [48]	3.500	0.7	17	7.3	7.8	3.350215	5.286683	2996.348225
LGSI4 [48]	3.501	0.7	17	7.3	7.8	3.350214	5.286683	2996.348205
AAO [68]	3.499	0.6999	17	7.3	7.8	3.3502	5.2872	2996.783
PSO-DE [62]	3.5	0.7	17	7.3	7.8	3.35021	5.28668	2996.3481
LGSI2 [48]	3.5	0.7	17	7.3	7.8	3.350215	5.286683	2996.348166
GWO [36]	3.501	0.7	17	7.3	7.811013	3.350704	5.287411	2997.81965
APSO [69]	3.501313	0.7	18	8.127814	8.042121	3.352446	5.287076	3187.630486
AAO [68]	3.4999	0.7	17	7.3	7.8	3.3502	5.2877	2997.058
CS [39]	3.5015	0.7000	17	7.6050	7.8181	3.3520	5.2875	3000.9810
SCA [22]	3.521	0.7	17	8.3	7.923351	3.355911	5.300734	3026.83772
FA [70]	3.507495	0.7001	17	7.719674	8.080854	3.351512	5.287051	3010.137492
AOA	3.50384	0.7	17	7.3	7.72933	3.35649	5.2867	2997.9157

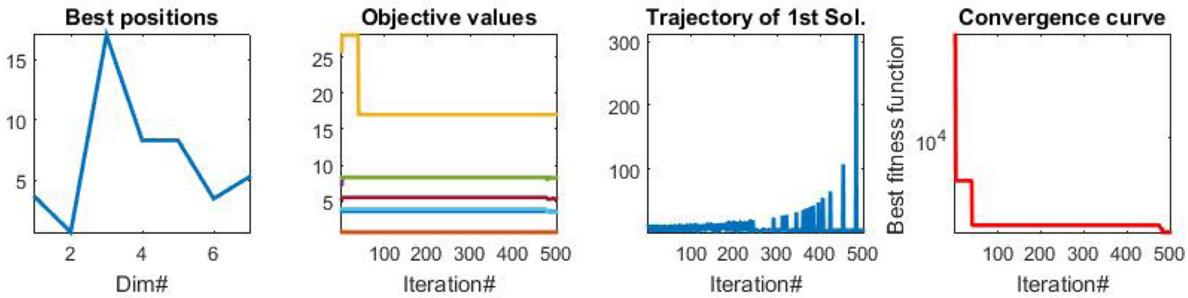


Fig. 22. Qualitative results for the speed reducer design problem.

other stochastic components, including local search or global search methods, in the area of optimization to enhance its performance. Finally, the investigation of the utilization of AOA in other various disciplines would be a valuable contribution, such as in neural networks, image processing applications, feature selection, task scheduling in cloud computing, text and data mining applications, big data applications, signal denoising, recource management applications, smart home applications, network applications, industry and engineering applications, other benchmark test functions, other real-world problems.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] V. Kumar, J.K. Chhabra, D. Kumar, Parameter adaptive harmony search algorithm for unimodal and multimodal optimization problems, J. Comput. Sci. 5 (2) (2014) 144–155.
[2] W. Chao, Y. Jin Ming Koh, G.X. Neng, H.C. Kang, Material and shape optimization of bi-directional functionally graded plates by giga and an improved multi-objective particle swarm optimization algorithm, Comput. Methods Appl. Mech. Engrg. 366 (2020).
[3] J. Zhang, M. Xiao, L. Gao, Q. Pan, Queuing search algorithm: A novel metaheuristic algorithm for solving engineering optimization problems, Appl. Math. Model. 63 (2018) 464–490.
[4] W. Zhao, C. Du, S. Jiang, An adaptive multiscale approach for identifying multiple flaws based on xfem and a discrete artificial fish swarm algorithm, Comput. Methods Appl. Mech. Engrg. 339 (2018) 341–357.
[5] L. Abualigah, Multi-verse optimizer algorithm: A comprehensive survey of its results, variants and applications, Neural Comput. Appl. (2020) 1–26.
[6] V.V. de Melo, W. Banzhaf, Drone squadron optimization: a novel self-adaptive algorithm for global numerical optimization, Neural Comput. Appl. 30 (10) (2018) 3117–3144.

[7] L. Abualigah, A. Diabat, A comprehensive survey of the grasshopper optimization algorithm: results, variants, and applications, Neural Comput. Appl. (2020) 1–24.
[8] L. Abualigah, A. Diabat, Z.W. Geem, A comprehensive survey of the harmony search algorithm in clustering applications, Appl. Sci. 10 (11) (2020) 3827.
[9] L. Abualigah, Group search optimizer: a nature-inspired meta-heuristic optimization algorithm with its results, variants, and applications, Neural Comput. Appl. (2020) 1–24.
[10] A. Faramarzi, M. Heidarinejad, B. Stephens, S. Mirjalili, Equilibrium optimizer: A novel optimization algorithm, Knowl.-Based Syst. 191 (2020) 105190.
[11] A. Sadollah, H. Sayyaadi, H.M. Lee, J.H. Kim, et al., Mine blast harmony search: a new hybrid optimization method for improving exploration and exploitation capabilities, Appl. Soft Comput. 68 (2018) 548–564.
[12] S. Gholizadeh, M. Danesh, C. Gheyratmand, A new newton metaheuristic algorithm for discrete performance-based design optimization of steel moment frames, Comput. Struct. 234 (2020) 106250.
[13] N.A. Kallioras, N.D. Lagaros, D.N. Avtzis, Pity beetle algorithm–a new metaheuristic inspired by the behavior of bark beetles, Adv. Eng. Softw. 121 (2018) 147–166.
[14] L. Abualigah, M. Shehab, M. Alshinwan, H. Alabool, Salp swarm algorithm: a comprehensive survey, Neural Comput. Appl. (2019) 1–21.
[15] L.J. Fogel, A.J. Owens, M.J. Walsh, Artificial Intelligence Through Simulated Evolution.
[16] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, J. Glob. Optim. 11 (4) (1997) 341–359.
[17] N. Hansen, S.D. Müller, P. Koumoutsakos, Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es), Evol. Comput. 11 (1) (2003) 1–18.
[18] A.H. Gandomi, A.H. Alavi, Krill herd: a new bio-inspired optimization algorithm, Commun. Nonlinear Sci. Numer. Simul. 17 (12) (2012) 4831–4845.
[19] L. Abualigah, M. Shehab, M. Alshinwan, S. Mirjalili, M. Abd Elaziz, Ant lion optimizer: A comprehensive survey of its variants and applications, Arch. Comput. Methods Eng. (2020).
[20] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp swarm algorithm: A bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (2017) 163–191.
[21] M.-Y. Cheng, D. Prayogo, Symbiotic organisms search: a new metaheuristic optimization algorithm, Comput. Struct. 139 (2014) 98–112.
[22] S. Mirjalili, Sca: a sine cosine algorithm for solving optimization problems, Knowl.-based Syst. 96 (2016) 120–133.
[23] A. Kaveh, N. Farhoudi, A new optimization method: Dolphin echolocation, Adv. Eng. Softw. 59 (2013) 53–70.
[24] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, science 220 (4598) (1983) 671–680.
[25] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, Gsa: a gravitational search algorithm, Inf. Sci. 179 (13) (2009) 2232–2248.
[26] S. Mirjalili, S.M. Mirjalili, A. Hatamlou, Multi-verse optimizer: a nature-inspired algorithm for global optimization, Neural Comput. Appl. 27 (2) (2016) 495–513.
[27] A. Kaveh, S. Talatahari, A novel heuristic optimization method: charged system search, Acta Mech. 213 (3–4) (2010) 267–289.
[28] E. Atashpaz-Gargari, C. Lucas, Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition, in: 2007 IEEE Congress on Evolutionary Computation, Ieee, 2007, pp. 4661–4667.
[29] R.V. Rao, V.J. Savsani, D. Vakharia, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, Comput. Aided Des. 43 (3) (2011) 303–315.
[30] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1) (1997) 67–82.
[31] M.K. Habib, A.K. Cherri, Parallel quaternary signed-digit arithmetic operations: addition, subtraction, multiplication and division, Opt. Laser Technol. 30 (8) (1998) 515–525.
[32] E. Bonabeau, M. Dorigo, D.d.R.D.F. Marco, G. Theraulaz, G. Théraulaz, et al., Swarm Intelligence: from Natural to Artificial Systems, Vol. 1, Oxford university press, 1999.
[33] R. Eberhart, J. Kennedy, Particle swarm optimization, in: Proceedings of the IEEE International Conference on Neural Networks, Vol. 4, Citeseer, 1995, pp. 1942–1948.
[34] D. Simon, Biogeography-based optimization, IEEE Trans. Evol. Comput. 12 (6) (2008) 702–713.
[35] X.-S. Yang, M. Karamanoglu, X. He, Flower pollination algorithm: a novel approach for multiobjective optimization, Eng. Optim. 46 (9) (2014) 1222–1237.
[36] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61.
[37] X.-S. Yang, A.H. Gandomi, Bat algorithm: a novel approach for global engineering optimization, Eng. Comput. (2012).
[38] A.H. Gandomi, X.-S. Yang, A.H. Alavi, Mixed variable structural optimization using firefly algorithm, Comput. Struct. 89 (23–24) (2011) 2325–2336.
[39] A.H. Gandomi, X.-S. Yang, A.H. Alavi, Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems, Eng. Comput. 29 (1) (2013) 17–35.
[40] S. Mirjalili, Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm, Knowl.-based Syst. 89 (2015) 228–249.
[41] K. Deb, An efficient constraint handling method for genetic algorithms, Comput. Methods Appl. Mech. Engrg. 186 (2–4) (2000) 311–338.
[42] L. Xia, L. Zhang, Q. Xia, T. Shi, Stress-based topology optimization using bi-directional evolutionary structural optimization method, Comput. Methods Appl. Mech. Engrg. 333 (2018) 356–370.
[43] A.H. Gandomi, K. Deb, Implicit constraints handling for efficient search of feasible solutions, Comput. Methods Appl. Mech. Engrg. 363 (2020) 112917.

[44] M. Fesanghary, M. Mahdavi, M. Minary-Jolandan, Y. Alizadeh, Hybridizing harmony search algorithm with sequential quadratic programming for engineering optimization problems, Comput. Methods Appl. Mech. Engrg. 197 (33–40) (2008) 3080–3091.
[45] S.S. Rao, Engineering Optimization: Theory and Practice, John Wiley & Sons, 2019.
[46] S. Gholizadeh, E. Salajegheh, Optimal design of structures subjected to time history loading by swarm intelligence and an advanced metamodel, Comput. Methods Appl. Mech. Engrg. 198 (37–40) (2009) 2936–2949.
[47] A. Sadollah, A. Bahreininejad, H. Eskandar, M. Hamdi, Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems, Appl. Soft Comput. 13 (5) (2013) 2592–2612.
[48] A. Baykasoğlu, Ş. Akpinar, Weighted superposition attraction (wsa): A swarm intelligence algorithm for optimization problems–part 2: Constrained optimization, Appl. Soft Comput. 37 (2015) 396–415.
[49] K. Ragsdell, D. Phillips, Optimal design of a class of welded structures using geometric programming.
[50] K. Deb, Optimal design of a welded beam via genetic algorithms, AIAA J. 29 (11) (1991) 2013–2015.
[51] K.S. Lee, Z.W. Geem, A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice, Comput. Methods Appl. Mech. Engrg. 194 (36–38) (2005) 3902–3933.
[52] F.-z. Huang, L. Wang, Q. He, An effective co-evolutionary differential evolution for constrained optimization, Appl. Math. Comput. 186 (1) (2007) 340–356.
[53] Q. He, L. Wang, An effective co-evolutionary particle swarm optimization for constrained engineering design problems, Eng. Appl. Artif. Intell. 20 (1) (2007) 89–99.
[54] A. Kaveh, M. Khayatazad, A new meta-heuristic method: ray optimization, Comput. Struct. 112 (2012) 283–294.
[55] S. Mirjalili, A. Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67.
[56] M.A. Elaziz, D. Oliva, S. Xiong, An improved opposition-based sine cosine algorithm for global optimization, Expert Syst. Appl. 90 (2017) 484–500.
[57] J.S. Arora, Introduction to Optimum Design, Elsevier, 2004.
[58] C.A.C. Coello, Use of a self-adaptive penalty approach for engineering optimization problems, Comput. Ind. 41 (2) (2000) 113–127.
[59] M. Mahdavi, M. Fesanghary, E. Damangir, An improved harmony search algorithm for solving optimization problems, Appl. Math. Comput. 188 (2) (2007) 1567–1579.
[60] E. Mezura-Montes, C.A.C. Coello, An empirical study about the usefulness of evolution strategies to solve constrained optimization problems, Int. J. Gen. Syst. 37 (4) (2008) 443–473.
[61] E. Sandgren, Nonlinear integer and discrete programming in mechanical design optimization, J. Mech. Des. 112 (2) (1990) 223–229.
[62] H. Liu, Z. Cai, Y. Wang, Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization, Appl. Soft Comput. 10 (2) (2010) 629–640.
[63] Q. He, L. Wang, A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization, Appl. Math. Comput. 186 (2) (2007) 1407–1422.
[64] A. Kaveh, S. Talatahari, An improved ant colony optimization for constrained engineering design problems, Eng. Comput. 27 (1) (2010) 155–182.
[65] M. Zhang, W. Luo, X. Wang, Differential evolution with dynamic stochastic selection for constrained optimization, Inform. Sci. 178 (15) (2008) 3043–3074.
[66] J.-F. Tsai, Global optimization of nonlinear fractional programming problems in engineering design, Eng. Optim. 37 (4) (2005) 399–409.
[67] T. Ray, P. Saini, Engineering design optimization using a swarm with an intelligent information sharing among individuals, Eng. Optim. 33 (6) (2001) 735–748.
[68] J.M. Czerniak, H. Zarzycki, D. Ewald, Aao as a new strategy in modeling and simulation of constructional problems optimization, Simul. Model. Pract. Theory 76 (2017) 22–33.
[69] N.B. Guedria, Improved accelerated pso algorithm for mechanical engineering optimization problems, Appl. Soft Comput. 40 (2016) 455–467.
[70] A. Baykasoğlu, F.B. Ozsoydan, Adaptive firefly algorithm with chaos for mechanical design optimization problems, Appl. Soft Comput. 36 (2015) 152–164.