



Hybrid arithmetic optimization algorithm with hunger games search for global optimization

Shubham Mahajan¹ · Laith Abualigah^{2,3} · Amit Kant Pandit¹

Received: 19 March 2021 / Revised: 1 March 2022 / Accepted: 9 March 2022 /

Published online: 31 March 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Recently, many population-dependent methods have been proposed. Despite their acceptance in many applications, we are still exploring suggested methods to solve actual problems. Consequently, researchers need to change and refine their procedures significantly based on the major evolutionary processes to achieve quicker convergence, more consistent equilibrium with high-quality performance and optimization. Therefore, a new hybrid method using Hunger Games Search (HGS) and Arithmetic Optimization Algorithm (AOA) is proposed in this paper. HGS is a recently proposed population-dependent optimization method that stabilizes the features and efficiently performs unconstrained and constrained problems. In contrast, AOA is a modern meta-heuristic optimization method. They can be applied to different problems, including image processing, machine learning, wireless networks, power systems, engineering design etc. The proposed method is analyzed in context with HGS and AOA. Each method is tested on the same parameters like population size and no. of iteration to evaluate the performance. The proposed method (AOA-HGS) is assessed by varying the dimensions on 23 functions (F1-F23). The impact of varying dimensions is a standard test utilized in previous studies for optimizing test functions that show the effect of varying dimensions on the efficiency of AOA-HGS. From this, it is noted that it works efficiently for both high and low dimensional problems. In high dimensional problem population, dependent methods give efficient search results. The AOA-HGS is very competitive and superior compared with

✉ Shubham Mahajan
mahajanshubham2232579@gmail.com

Laith Abualigah
aligah.2020@gmail.com

Amit Kant Pandit
amitkantpandit@gmail.com

¹ School of Electronics & Communication, Shri Mata Vaishno Devi University, Katra, J&K 182320, India

² Faculty of Computer Science and Informatics Amman Arab University, Amman 11953, Jordan

³ School of Computer Sciences, Universiti Sains Malaysia, 11800 Pulau Pinang, Malaysia

others on test functions. No. of optimization methods obtained optimum results, but AOA-HGS has the best when compared with all. So, AOA-HGS is capable of getting optimum results.

Keywords Hunger games search · Arithmetic optimization algorithm · Engineering design problems · Metaheuristic · Swarm intelligence · Optimization methods

1 Introduction

Technologies in the present world have less feature space and depend on priorities and financial limits in information, knowledge-dependent, and expert systems. A practicable solution and adequate information are to be found by researchers using various algorithms for different problems in areas like image segmentation [34], optimization [31], target tracking systems [50], QoS-aware and social recommendation [28–30], problems in scheduling [40], prediction of gold price [48], etc.

The method of determining appropriate values for variables of a given problem to minimize and optimize the objective function is termed optimization. There are difficulties of optimization in various fields of analysis. Various steps should be taken to solve an optimization problem. Next, the parameters of the problem need to be defined. Issues are categorized as continuous or discrete, depending on the type of parameters. Secondly, it is important to consider the restrictions applied to the parameters [42].

Constraints split optimization problems into confined and unregulated ones. The purposes of the issue should be examined and addressed [10, 36]. Mathematical optimization depends primarily on gradient-based knowledge of functions to obtain an optimum solution. While these methods are already being utilized by various researchers, and have few drawbacks. Mathematical optimization methods are influenced by local optima entrapment. This applies to a method that believes that the local solution is a global solution and fails to reach the optimum goal. They are most often unsuccessful in problems with uncertain or computationally costly derivatives [39]. Another form of an optimization algorithm that removes these major disadvantages is stochastic optimization [45].

Stochastic approaches rely on arbitrary operators to prevent local optimization. It begins the optimization by generating one or a series of arbitrary solutions to problems. Compared to mathematical optimization methods, it is not important to determine the solution's gradient to test the solutions using an objective function (s). Decisions on how to enhance outcomes are taken based on the realistic principles measured. Thus, the problem is assumed to be a black box, and it's a very effective tool for solving actual issues of undefined search spaces. Because of the above advantages, stochastic methods are commonly used [37]. Nature-inspired, population-dependent methods are the most common among stochastic optimization methods [51].

These approaches emulate normal problem-solving strategies, often those employed by species. Survival is the primary objective of all species. They have evolved and changed in various ways to accomplish this goal. It is also prudent to pursue guidance from nature as the greatest and oldest optimizing compiler on earth. These algorithms are categorized into two major groups: single-solution and multi-solution dependent. In the first stage single random solution for a specific problem is developed and improved and in the second multiple solutions for a specific issue are developed and improved. Multi-solution methods are more common

than single-solution approaches [38]. Multi-solution models have an inherently higher local optimum-avoidance owing to the improvement of multiple solutions during the optimization process. The solution stuck in a locally optimal is supported by certain methods to leaping from the locally optimal.

Existing solutions are searching a greater portion of search space relative to single-solution methods, so the likelihood of reaching global optimum is large. Commonly used methods dependent on single-solution are simulated annealing and hill-climbing [14, 27]. They both are ideal, but the avoidance of the local optimum of SA is high the stochastic's cooling factor. Some new methods dependent on single-solution are iterated local search and tabu search [18, 20, 32]. Some multi-solution dependent popular methods are particle swarm optimization [15], genetic algorithms (GA) [23], differential evolution (DE) [46] and ant optimization (ACO) [11]. Darwin's evolutionary theory influenced the GA method. Solutions are viewed as entities in this method, and solution parameters take over the place of genes. Survival of the fittest species is the key motivation of this method, where the strongest appears to be more interested in improving bad solutions.

Previous studies state different swarm intelligence optimization methods like firefly algorithm (FA) [52, 57], dolphin echolocation (DEL) [25, 26], grey wolf optimizer (GWO) [39] and bat algorithm (BA) [53]. BA and DEL copy echolocation of dolphins for finding prey and bats navigate. But FA copies the mating nature of fireflies. In Cuckoo Search (CS) [54, 56] method, the cuckoo's reproductive behavior is used in the optimization method. The hunting behavior of grey wolves is used in the swarm method technique called GWO. Other methods like the state of matter search (SMS) [12, 13] utilize different types of matter for problem optimization.

In contrast, the flower pollination algorithm (FPA) [55] utilizes the pollination and survival behavior of flowers for pollination. Now, the question is, what is the need for new methods when there are many pre-existing. This query's solution is in the No Free Lunch (NFL) [49] theorem, which has scientifically demonstrated no optimization strategy to solve all optimization problems. In other terms, techniques in this area work similarly on average when considering all optimization problems. This theorem has inspired the fast-growing algorithms that have been proposed in the last decades. It is also one of the reasons behind this paper.

HGS is influenced by animals' social behavior, whose search for food is related to their extent of hunger. This method is applied and implemented based on the common characteristics of animals and their food search. In this paper, we proposed a hybrid method using two meta-heuristic methods AOA and HGS. Because of gradient-free and simple structure, higher local optimum avoidance and letting issues as black-box nature-inspired methods are used as large in engineering and other problems [8, 21, 59] [34]. We are, therefore, still exploring the utilization of suggested methods to solve actual problems.

The main contributions of the paper are:

1. The application of global optimization utilizing AOA-HGS gives better results when experimental results are compared with HGSO, AOA, GOA, GWO, SCA.
2. AOA-HGS reduced the computational complexity and it also works efficiently for both high and low dimensional problems.

The paper is structured as: in section 2 expounds on HGS. AOA is discussed in section 3. Proposed work is discussed in section 4. Results are discussed in section 5. Conclusion and future scope are discussed in section 6.

2 Hunger games search (HGS) optimization

2.1 Approach food

To express its approaching behavior in mathematical formulas, the following formulas are proposed to imitate the contraction mode [58]:

$$\overrightarrow{X}(t+1) = \begin{cases} \overrightarrow{X}(t) \cdot (1 + \text{randn}(1)), & r_1 < l \\ \overrightarrow{W}_1 \cdot \overrightarrow{X}_b + \overrightarrow{R} \cdot \overrightarrow{W}_2 \cdot |\overrightarrow{X}_b - \overrightarrow{X}(t)|, & r_1 > l, r_2 > E \\ \overrightarrow{W}_1 \cdot \overrightarrow{X}_b - \overrightarrow{R} \cdot \overrightarrow{W}_2 \cdot |\overrightarrow{X}_b - \overrightarrow{X}(t)|, & r_1 > l, r_2 < E \end{cases} \quad (1)$$

where \overrightarrow{R} is in the range of $[-a, a]$; r_1 and r_2 respectively represent random numbers, which are in the range of $[0, 1]$; $\text{randn}(1)$ is a random number satisfying normal distribution; t indicates that the current iterations; \overrightarrow{W}_1 and \overrightarrow{W}_2 represent the weights of hunger; \overrightarrow{X}_b represents the location information of a random individual in all the optimal individuals; $\overrightarrow{X}(t)$ represents each individual's location, and the value of l has been discussed in the parameter setting experiment.

The formula of E is as follows:

$$E = \text{sech}(|F(i) - BF|) \quad (2)$$

where $i \in 1, 2, \dots, n$, $F(i)$ represents the fitness value of each individual, and BF is the best fitness obtained in the current iteration process. sech is a hyperbolic function ($\text{sech}(x) = \frac{2}{e^x + e^{-x}}$).

The formula of \overrightarrow{R} is as follows:

$$\overrightarrow{R} = 2 \times a \times \text{rand} - a \quad (3)$$

$$a = 2 \times \left(1 - \frac{t}{\text{Max_iter}}\right) \quad (4)$$

where rand is a random number in the range of $[0, 1]$; and Max_iter stands for the largest number of iterations.

2.2 Hunger role

The starvation characteristics of individuals in search are simulated mathematically.

The formula of \overrightarrow{W}_1 in Eq. (5) is as follows:

$$\overrightarrow{W}_1(i) = \begin{cases} \text{hungry}(i) \cdot \frac{N}{SHungry} \times r_4, & r_3 < l \\ 1 & r_3 > l \end{cases} \quad (5)$$

The formula of \overrightarrow{W}_2 in Eq. (6) is shown as follows:

$$\overrightarrow{W}_2(i) = (1 - \exp(-|\text{hungry}(i) - SHungry|)) \times r_5 \times 2 \quad (6)$$

where *hungry* represents the hunger of each individual; *N* represents the number of individuals, and *SHungry* is the sum of hungry feelings of all individuals, that is $sum(hungry)$. r_3 , r_4 and r_5 are random numbers in the range of [0, 1].

The formula for $hungry(i)$ is provided below:

$$hungry(i) = \begin{cases} 0, & AllFitness(i) = BF \\ hungry(i) + H, & AllFitness(i) = BF \end{cases} \quad (7)$$

where $AllFitness(i)$ preserves the fitness of each individual in the current iteration.

The formula for *H* can be seen as follows:

$$TH = \frac{F(i)-BF}{WF-BF} \times r_6 \times 2 \times (UB-LB) \quad (8)$$

$$H = \begin{cases} LH \times (1 + r), & TH < LH \\ TH, & TH \geq LH \end{cases} \quad (9)$$

where r_6 is a random number in the range of [0, 1]; $F(i)$ represents each individual's fitness value of individual; BF is the best fitness obtained in the current iteration process; WF stands for the worst fitness obtained in the current iteration process; and UB and LB indicate the upper and lower bounds of the search space, respectively. The hunger sensation *H* is limited to a lower bound, LH .

Algorithm 1. Pseudo Code of HGS

Initialize the parameters N, Max_iter, l, D, SHungry

Initialize the positions of Individuals $X_i (i = 1, 2, \dots, N)$

While ($t \leq Max_iter$)

Calculate the fitness of all Individuals

Update BF, WF, X_b , BI

Calculate the Hungry by Eq. (7)

Calculate the W_1 by Eq. (5)

Calculate the W_2 by Eq. (6)

For each Individuals

Calculate E by Eq. (2)eq

Update R by Eq. (3)

Update positions by Eq. (1)

End For

$t = t + 1$

End While

Return BF, X_b

3 Arithmetic optimization algorithm

AOA [2] is a new meta-heuristic method that uses common mathematical operations such as Division (D), Addition (A), Multiplication (M), Subtraction (S), as shown in Fig. 1, which is applied and modeled to execute optimization in a wide variety of search fields [8]. Commonly, population-based algorithms (PBA) launch their improvement processes by randomly selecting several candidate strategies. This defined solution is enhanced incrementally by a set of optimization standards and analyzed sequentially by a particular objective function, and that's the basis of optimization techniques. Although PBA is stochastically trying to find some efficient strategy to optimization problems, a single-run solution is not guaranteed. However, the chance of an optimum global solution to the problem is enhanced by a large set of possible solutions and optimization simulations [21]. Considering variations among meta-heuristic methods [43, 44] in PBA approaches, the optimization process comprises two cycles: exploitation vs. exploration. The previous examples for extensive coverage are search fields through search agents to bypass local solutions. Above is an increase in the performance of solutions achieved during the exploration process.

3.1 Motivation

Arithmetic is a key component of mathematics and its most important components of modern math, including analysis, geometry and algebra. Arithmetic operators (AO) [2] are traditionally used for the study of numbers [59]. These basic math functions are used for optimization for finding ideal elements, particularly with selected solutions. Optimization [1, 3, 4, 16, 17, 22, 41] challenges have appeared in all mathematical fields, such as engineering [5–7, 9, 19, 24, 47], economics and computer science to organizational analysis and technology, and the advancement of optimization methods has drawn mathematics [33] attention from time to

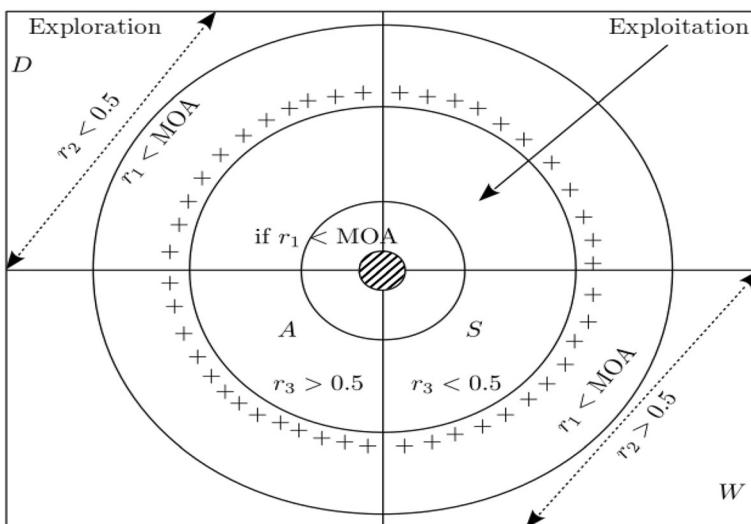


Fig. 1 Shows AOA search phases [2]

time. The key motivation of the new AOA is the use of AO to solve problems. The behavior of AO and their effect mostly on existing algorithms, the arrangement of AO and their superiority is shown in Fig. 2. AOA is then proposed based on a statistical model. [35]

3.2 Initial stage

The method of optimization starts with selected sets denoted by A as in Eq. 10. The ideal set in every iteration is created randomly and is taken as the optimum solution [2].

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & \dots & a_{1,j} & a_{1,1} & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & \dots & a_{2,j} & \dots & a_{2,n} \\ a_{3,1} & a_{3,2} & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ a_{N-1,1} & \dots & \dots & \dots & a_{N-1,j} & \dots & a_{N-1,n} \\ a_{N,1} & \dots & \dots & \dots & a_{N,j} & a_{N,n-1} & a_{N,n} \end{bmatrix} \tag{10}$$

Exploitation/Exploration should be carefully chosen at the start of AOA. The coefficient of math optimizer accelerated (MOA) is defined in Eq. 11.

$$MOA (C_{iter}) = Min + C_{iter} \times \left(\frac{Max-Min}{M_{iter}} \right) \tag{11}$$

where

$MOA (C_{iter}) = i^{th}$ iteration function value.

M_{iter} = Max. no. of iteration.

Max & Min = Accelerated Function of Max. and Min. Values.

C_{iter} = current iteration (within 1 and M_{iter}).

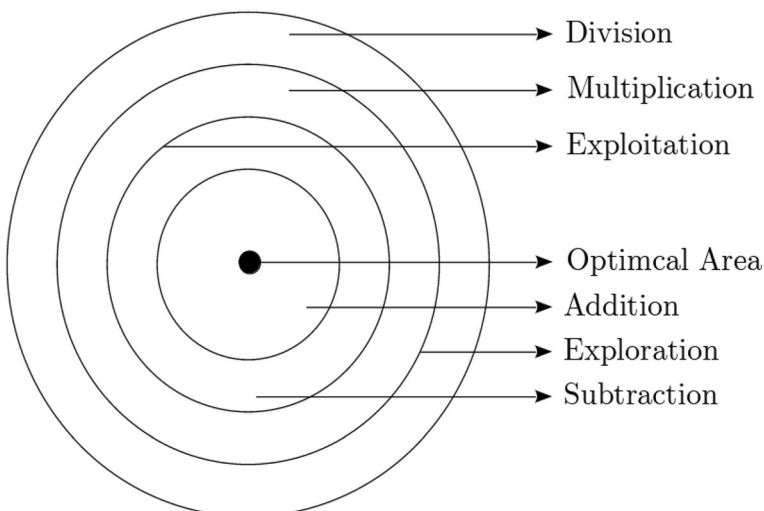


Fig. 2 Shows arithmetic operators according to superiority

3.3 Exploration stage

The exploratory nature of AOA is discussed, as per the AO, mathematical calculations whether using Division (D) or Multiplication (M) operator have obtained high distribution values or decisions that contribute to an exploration search method. However, as opposed to other operators, these D and M operators never easily reach the objective due to the high distribution of S and A operators. AOA exploration operators exploit the search field arbitrarily through many regions and seek a better alternative dependent on two key search techniques M and D search techniques as shown in Eq.12 [58].

$$a_{i,j}(C_{iter} + 1) = \begin{cases} besta_j \div (MOP \div \varepsilon) \times ((UB_j - LB_j) \times \mu + LB_j), r_2 < 0.5 \\ besta_j \times MOP \times ((UB_j - LB_j) \times \mu + LB_j), otherwise \end{cases} \quad (12)$$

where,

$a_i(C_{iter} + 1) = i^{th}$ solution of next iteration.

$a_{i,j}(C_{iter} + 1) = j^{th}$ position in current iteration.

$\mu =$ control parameter ≤ 0.5 .

LB_j & $UB_j =$ Lower & Upper bound limit

$\varepsilon =$ smallest integer no.

$besta_j = j^{th}$ position of optimum solution till now

$$MOP(C_{iter}) = 1 - \frac{C_{iter}^{\frac{1}{\alpha}}}{M_{iter}^{\frac{1}{\alpha}}} \quad (13)$$

where,

Math Optimizer Probability (MOP) = coefficient

$MOP(C_{iter}) = i^{th}$ iteration function value

$C_{iter} =$ current iteration

$M_{iter} =$ Max. iterations ≤ 5

3.4 Exploitation stage

The exploitation nature of AOA is discussed, as per AO mathematical formulas, whether using addition (A) or subtraction (S) as they provided high-density results. AOA exploitation operators exploit the search field deeply through many regions and seek a better alternative dependent on two key search techniques A and S search techniques as shown in Eq.14 [2].

$$a_{i,j}(C_{iter} + 1) = \begin{cases} besta_j - MOP \times ((UB_j - LB_j) \times \mu + LB_j), r_3 < 0.5 \\ besta_j + MOP \times ((UB_j - LB_j) \times \mu + LB_j), otherwise \end{cases} \quad (14)$$

4 Proposed work

Several population-dependent approaches have recently been proposed. Despite their adoption in a wide area of engineering applications, we are still examining suggested methods to solve real problems. As a result, researchers need to substantially modify and improve their methods, often based on major evolutionary processes, to achieve faster integration, a more consistent

balance with high-quality efficiency and optimization. Therefore, a new hybrid method using Hunger Games Search (HGS) and Arithmetic Optimization Algorithm (AOA) is proposed in this paper. HGS is a recently proposed population-dependent optimization method that stabilizes the features and efficiently performs unconstrained and constrained problems.

In contrast, AOA is a modern meta-heuristic optimization method. They can be applied to different problems, including image processing, machine learning, wireless networks, power systems, engineering design etc. The method proposed is analyzed in context with HGS and AOA. To evaluate the performance, each method is tested on the same parameters like population size and no. of iteration. The proposed method (AOA-HGS) is evaluated by varying the dimensions. The impact of varying dimensions is a standard test utilized in previous studies for optimizing test functions that show the effect of varying dimensions on the efficiency of AOA-HGS. From this, it is noted that it works efficiently for both high and low dimensional problems. In high dimensional problem population, dependent methods give efficient search results. Figure 3 below demonstrates the step-by-step flow and implementation of proposed model.

From Fig. 3, the different steps utilized in the proposed method according to their implementation are discussed. The 1st step is defining parameters in this different parameter to be used are defined. The 2nd step is to generate the solution by means of the defined parameters. The 3rd step is to estimate the fitness function followed by choosing the best solution in 4th step. In 5th step, there is a condition, if the value of the random number (*rand*) is larger than 0.5, then HGS is not used, and if the value is smaller than 0.5, AOA is used. Then in 6th step, if the desired criteria are met, then the best solution is returned in 7th step; else, by feedback lope, it is again fed to 3rd step for estimating the fitness function.

The complexity of the proposed AOAHGS is based on the complexity of original AOA and HGS and it is given as follows:

$$O(AOAHGS) = (N) \times O(AOA) \times O(HGS) \tag{15}$$

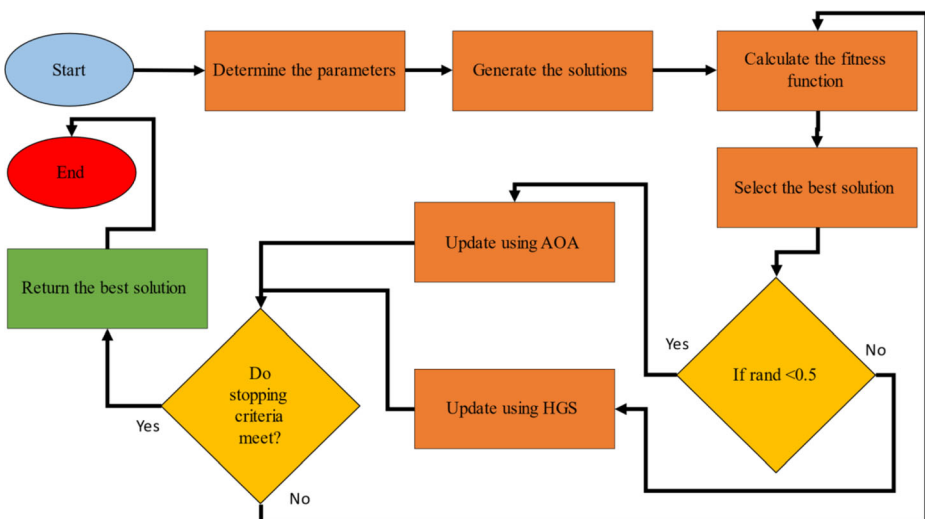


Fig. 3 Demonstrates the step-by-step flow and implementation of proposed model

$$O(AOA) = O(N \times (t \times Dim + 1)) \tag{16}$$

$$O(HGS) = O(N \times (t \times Dim + 1)) \tag{17}$$

Therefore, the total complexity of the proposed AOAHGS is given as follows:

$$O(AOAHGS) = O(t \times N \times (Dim + N)) \tag{18}$$

where, N presents the number of solutions, Dim presents the solution size, and t is the number of iterations.

5 Results & discussion

The proposed approach is examined, and the proposed system’s efficiency is correlated with the performance of existing methods. The implementation and testing are done in $i5 - 1.70GHz$ processor using MATLAB software. The proposed method’s performance (i.e., AOA + HGS) is examined on 23 test functions as described in Fig. 4. The results are further compared with HGSO, AOA, GOA, GWO, SCA.

To evaluate the performance, each method is tested on the same parameters like population size and no. of iteration. The proposed AOA-HGS is evaluated by varying the dimensions. The impact of varying dimensions is a standard test used in previous studies for optimizing test functions that show the effect of varying dimensions on the efficiency of AOA-HGS. From

Function	Description	Dimensions	Range	f_{min}
F1	$f(x) = \sum_{i=1}^n x_i^2$	10, 100,	[-100,100]	0
F2	$f(x) = \sum_{i=0}^n x_i + \prod_{i=0}^n x_i $	10, 100,	[-10,10]	0
F3	$f(x) = \sum_{i=1}^n (\sum_{j=1}^n x_j)^2$	10, 100,	[-100,100]	0
F4	$f(x) = \max_i \{ x_i , 1 \leq i \leq n\}$	10, 100,	[-100,100]	0
F5	$f(x) = \sum_{i=1}^{n-1} [100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2]$	10, 100,	[-30,30]	0
F6	$f(x) = \sum_{i=1}^n (x_i + 0.5)^2$	10, 100,	[-100,100]	0
F7	$f(x) = \sum_{i=0}^n ix_i^2 + \text{random}(0, 1)$	10, 100,	[-128,128]	0
F8	$f(x) = \sum_{i=1}^n (-x_i \sin(\sqrt{ x_i }))$	10, 100,	[-500,500]	-418.9829 × n
F9	$f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	10, 100,	[-5,12.5,12]	0
F10	$f(x) = -20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	10, 100,	[-32,32]	0
F11	$f(x) = 1 + \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{2}})$	10, 100,	[-600,600]	0
F12	$f(x) = \frac{\pi}{4} \{10 \sin(\pi y_1)\} + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1}) + \sum_{i=1}^n u(x_i, 10, 100, 4)]$, where $y_i = 1 + \frac{x_i + 1}{4} u(x_i, a, k, m) \begin{cases} K(x_i - a)^m & \text{if } x_i > a \\ 0 & -a \leq x_i \leq a \\ K(-x_i - a)^m & -a \leq x_i \end{cases}$	10, 100,	[-50,50]	0
F13	$f(x) = 0.1(\sin^2(3\pi x_1) + \sum_{i=1}^n (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)]) + (x_n - 1)^2 + 1 + \sin^2(2\pi x_n) + \sum_{i=1}^n u(x_i, 5, 100, 4)$	10, 100,	[-50,50]	0
F14	$f(x) = (\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^j (x_i - a_{ij})})^{-1}$	2	[-65,65]	1
F15	$f(x) = \sum_{i=1}^{11} [a_i - \frac{x_i(b_i^2 + b_i x_i)}{b_i^2 + b_i x_i + x_i^2}]^2$	4	[-5,5]	0.00030
F16	$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^3$	2	[-5,5]	-1.0316
F17	$f(x) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{97}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 10$	2	[-5,5]	0.398
F18	$f(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	[-2,2]	3
F19	$f(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{j=1}^3 a_{ij}(x_j - p_{ij})^2)$	3	[-1,2]	-3.86
F20	$f(x) = -\sum_{i=1}^4 c_i \exp(-\sum_{j=1}^6 a_{ij}(x_j - p_{ij})^2)$	6	[0,1]	-32
F21	$f(x) = -\sum_{i=1}^5 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,1]	-10.1532
F22	$f(x) = -\sum_{i=1}^7 [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,1]	-10.4028
F23	$f(x) = -\sum_{i=1}^{10} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,1]	-10.5363

Fig. 4 The description of the tested functions

this, it is noted that it works efficiently for both high and low dimensional problems. In high dimensional problem population, dependent methods give efficient search results. In this work, AOA-HGS is used to test scalable multi, and unimodal test functions (F1-F13) with two different dimensions (10 & 1000) and results of comparative methods are tested using ten benchmark functions (F14-F23). The AOA-HGS is dealing with 13 functions (F1-F13) with two different dimensions is compared using standard deviation (SD) and average fitness value (Avg).

AOA-HGS gives the most optimum results in 10, which shows that any optimization methods give optimum results in low dimensions. Now, AOA-HGS is testing using high dimensions (1000), which also gave the optimum results. To verify, the proposed AOA-HGS is compared with previously state-of-art algorithms using the same dimensions (10 & 100) and test function. When the results were analyzed, it showed that AOA-HGS is the most optimum in different cases.

It is compared with other methods to evaluate the convergence behavior of AOA-HGS, as shown in Fig. 5. The curves of convergence with test functions (F1-F13) can be seen in Fig. 5. After observing the Fig. 5, it becomes clear that AOA-HGS gives low and stable convergence compared with other methods. AOA-HGS has more optimum global search capability and fast convergence and attained optimum results than other methods on the same test functions concerning convergence speed and global search capability.

The average runtime of the proposed AOA-HGS algorithm is compared with other pre-existing methods with 10 dimensions is shown in Table 1, and with 1000 dimensions is shown in Table 2. As AOA-HGS depends on the population method, there is no need for optimization so, the running time needed by AOA-HGS is less in terms of seconds when compared with other methods. So, the computational efficiency of the proposed AOA-HGS is much optimum than other methods. Observing the results shows that the AOA-HGS is on top, followed by other methods.

If we notice the values in Table 3, the AOA-HGS is very competitive and superior compared with others on test functions (F14-F23). No. of optimization methods obtained optimum results, but AOA-HGS has the best when compared with all. So, AOA-HGS is capable of obtaining optimum results. It is that the ability of the proposed method gains the advantages of two powerful methods together. This proposal can also tackle the weaknesses of the signal search method, as clearly stated in the used tables and figures.

6 Conclusion & future scope

This work introduces a hybrid method (AOA-HGS) dependent on the population-based model to address optimization problems depending on the types of social animals in search of food (HGS) along with a meta-heuristic optimization method (AOA). The proposed AOA-HGS method is validated on an exhaustive collection of 23 functions (F1-F23). Results achieved were compared with other state-of-the-art methods like HGSO, AOA, GOA, GWO, SCA. Observing the experimental results, it became clear that AOA-HGS has more fast convergence with optimum global search capability. The AOA-HGS is very competitive and superior compared with others on test functions. No. of optimization methods obtained optimum results, but AOA-HGS has the best when compared with all. So, AOA-HGS is capable of obtaining optimum results. The main advantages of the proposed method it can find better

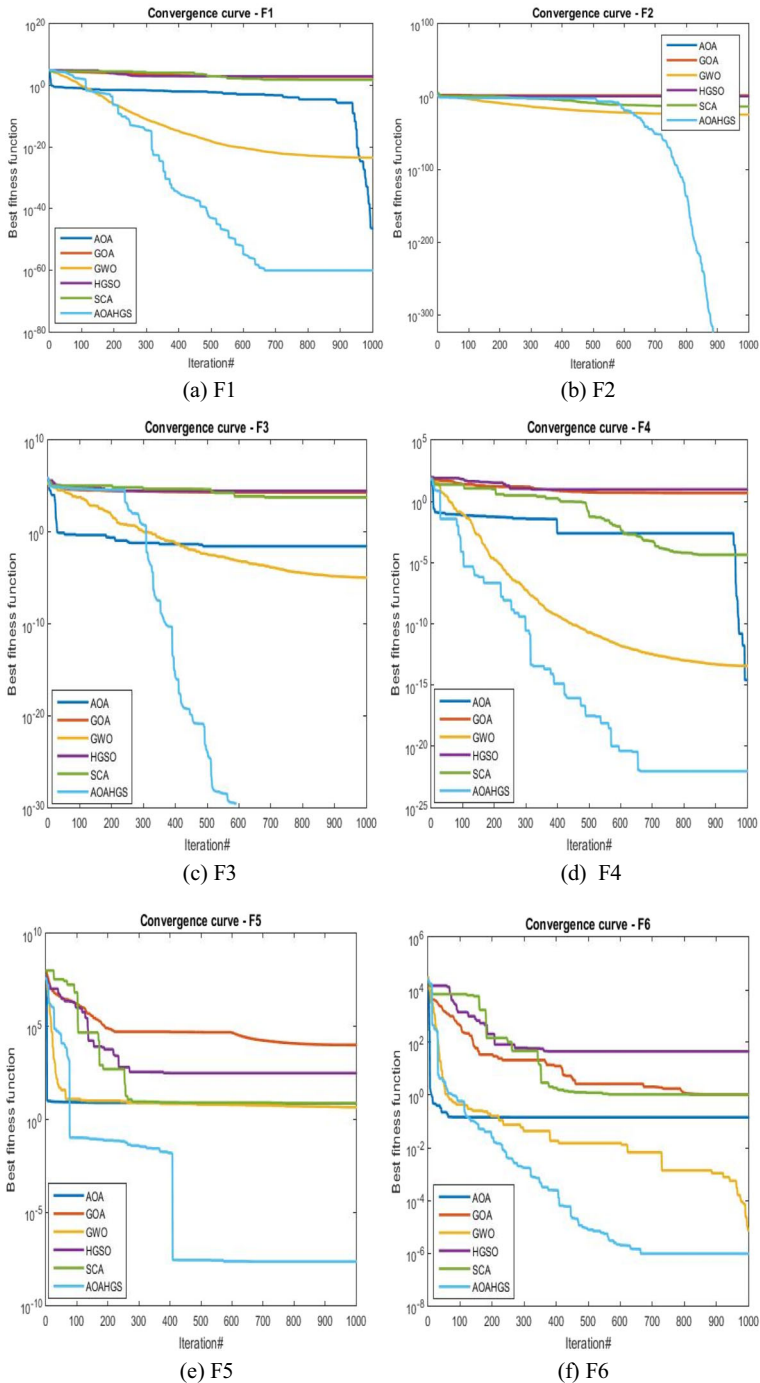


Fig. 5 Convergence behavior of AOA-HGS along with other methods (a) F1 (b) F2 (c) F3 (d) F4 (e) F5 (f) F6 (g) F7 (h) F8 (i) F9 (j) F10 (k) F11 (l) F12 (m) F13 (n) F14 (o) F15 (p) F16 (q) F17 (r) F18 (s) F19 (t) F20 (u) F21 (v) F22 (w) F23

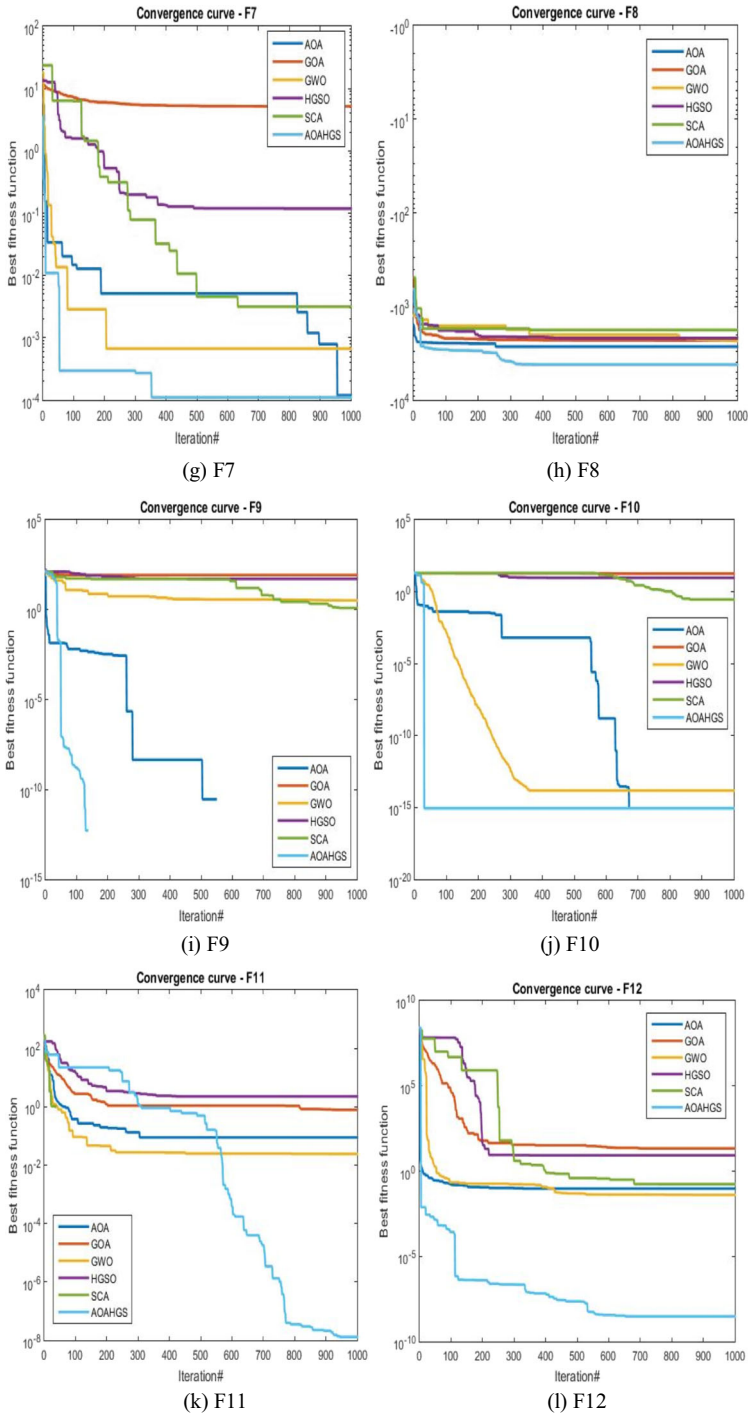


Fig. 5 continued.

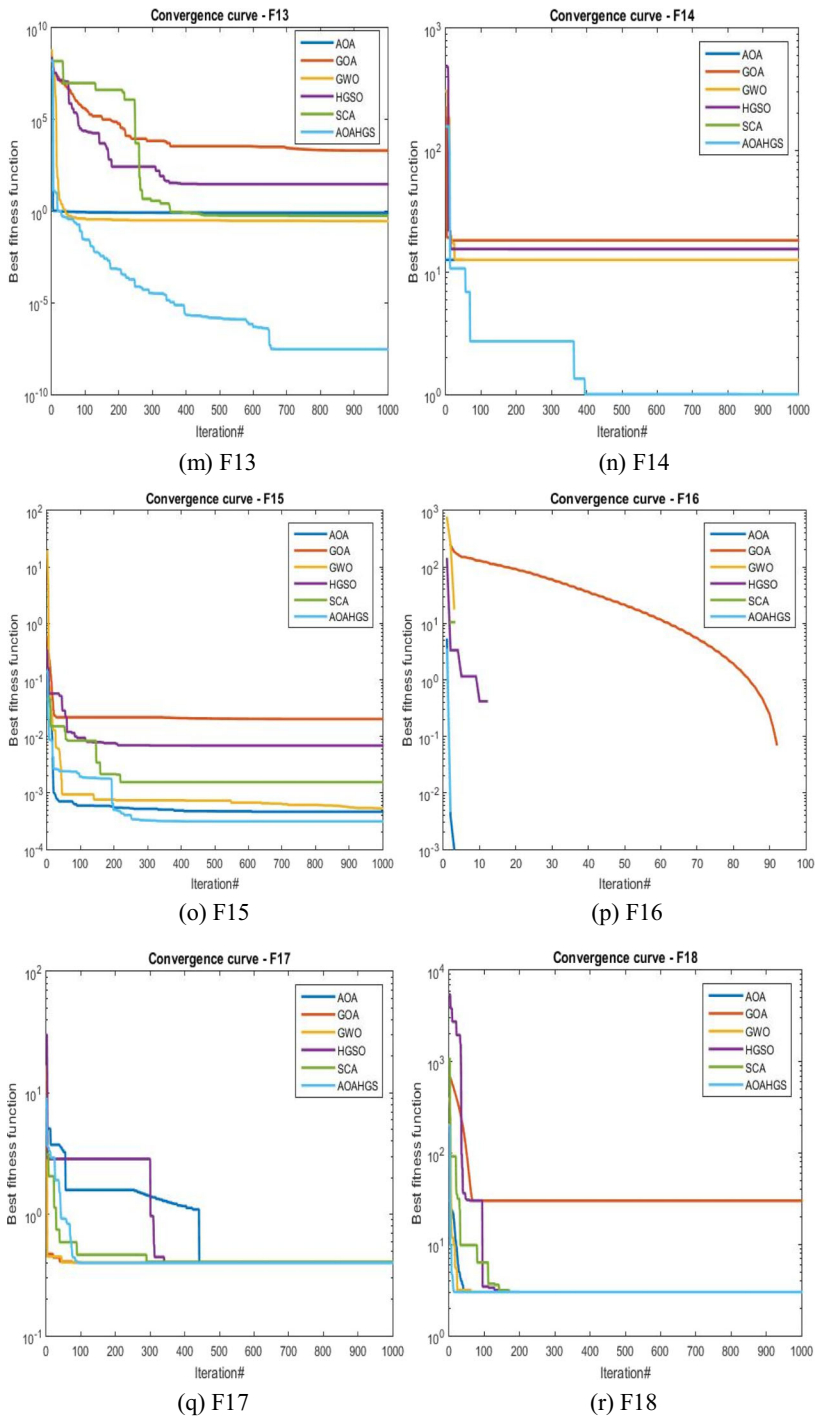


Fig. 5 continued.

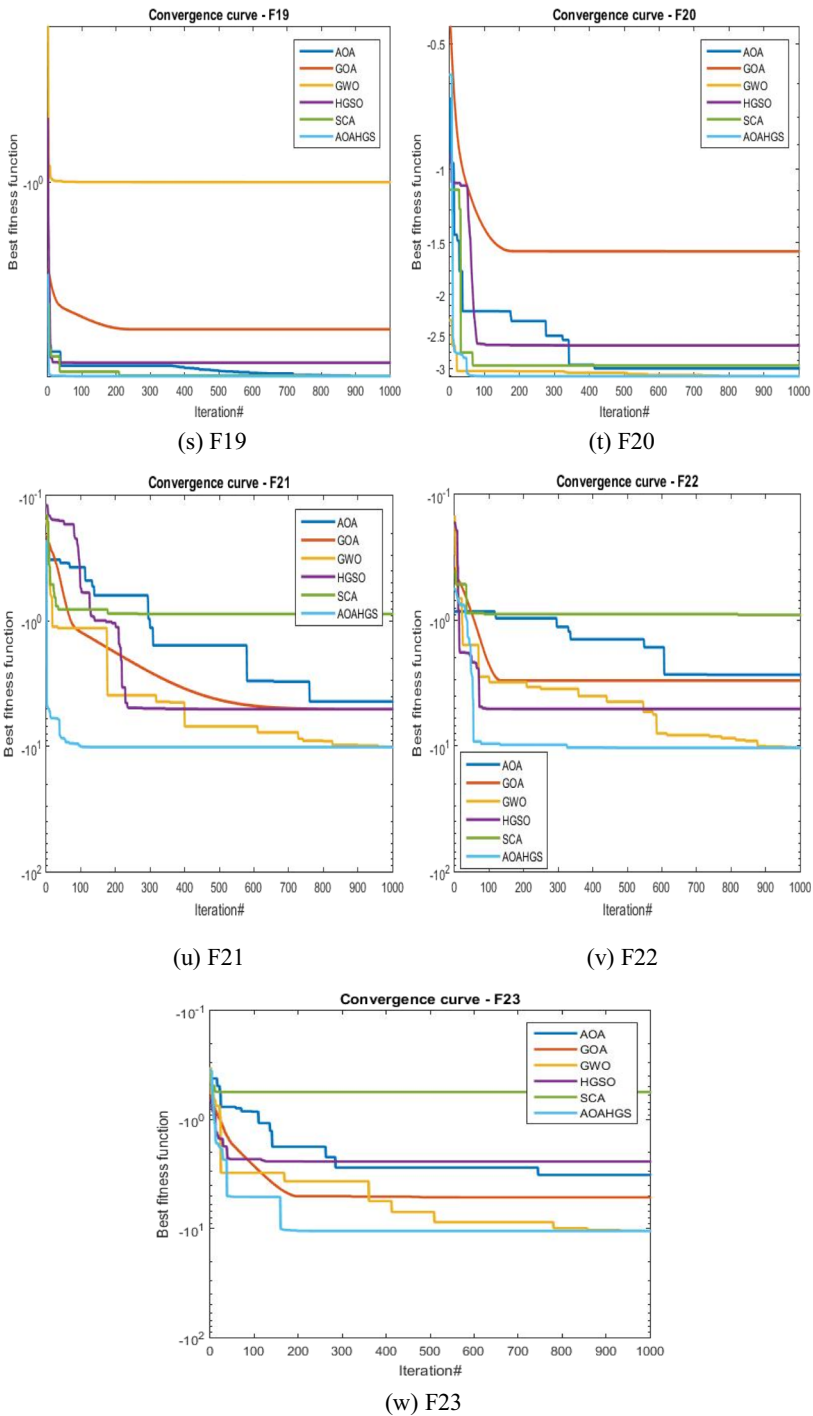


Fig. 5 continued.

Table 1 The comparative methods result using thirteen benchmark functions (F1–F13), where the dimension is fixed to 10

Function	Measure	AOA	GOA	GWO	HGSO	SCA	AOAHGS
F1	Worst	1.6962E-98	2.1633E+00	1.8080E-41	3.0790E+02	2.5092E-14	3.4438E-171
	Average	3.39323E-99	9.08386E-01	5.74541E-42	1.38634E+02	5.29214E-15	6.88772E-172
	Best	0.00000E+00	2.44329E-01	6.26418E-43	2.66400E+01	4.36916E-18	5.62916E-253
	STD	7.58750E-99	7.94368E-01	7.33839E-42	1.06292E+02	1.10741E-14	0.00000E+00
	Rank	2	5	3	6	4	1
F2	Worst	4.81661E-62	4.97770E+01	1.77949E-23	3.97461E+01	7.20482E-12	0.00000E+00
	Average	9.63322E-63	2.95455E+01	7.63370E-24	1.34336E+01	1.63909E-12	0.00000E+00
	Best	0.00000E+00	7.76904E+00	1.78339E-25	3.90569E+00	6.70035E-17	0.00000E+00
	STD	2.15405E-62	1.92063E+01	8.41160E-24	1.49161E+01	3.12765E-12	0.00000E+00
	Rank	2	6	3	5	4	1
F3	Worst	2.19888E-52	1.23105E+02	8.47430E-19	1.62897E+03	1.44781E-01	1.29724E-169
	Average	4.39776E-53	4.27695E+01	3.64560E-19	9.48485E+02	2.90367E-02	2.59449E-170
	Best	0.00000E+00	1.49695E+01	7.59451E-21	3.83770E+01	1.58926E-08	6.62064E-288
	STD	9.83370E-53	4.56831E+01	4.03013E-19	6.67745E+02	6.47034E-02	0.00000E+00
	Rank	2	5	3	6	4	1
F4	Worst	1.40329E-24	2.92010E+01	2.17382E-14	4.98319E+01	1.27511E-01	2.09886E-55
	Average	2.80659E-25	2.00943E+01	1.01008E-14	1.81041E+01	2.68876E-02	4.19772E-56
	Best	0.00000E+00	8.91033E+00	2.16768E-15	3.54249E+00	5.04201E-05	1.93994E-130
	STD	6.27571E-25	9.07753E+00	9.45559E-15	1.92850E+01	5.62732E-02	9.38640E-56
	Rank	2	6	3	5	4	1
F5	Worst	8.203020191	88.745.85516	8.098905995	7200.970177	8.947893658	6.666888869
	Average	7.933506989	35.859.91681	7.487928825	2955.840505	7.885069328	6.480260277
	Best	7.656774113	1146.714828	6.957962743	601.101513	7.295008906	6.263593086
	STD	0.24838742	33.652.1378	0.552551442	2628.501128	0.655674993	0.14361069
	Rank	4	6	2	5	3	1
F6	Worst	1.74987E-01	1.34040E+00	9.80894E-01	2.10631E+03	9.75030E-01	4.85330E-07
	Average	1.08599E-01	1.04051E+00	2.40980E-01	5.74616E+02	8.65734E-01	2.78673E-07
	Best	4.45436E-02	6.03520E-01	1.16176E-05	3.23015E+01	7.33881E-01	1.08222E-07
	STD	5.47507E-02	3.06613E-01	4.24840E-01	8.68745E+02	1.11336E-01	1.55272E-07
	Rank	2	5	3	6	4	1
F7	Worst	2.35022E-04	1.25188E+01	1.38601E-03	4.36449E-01	1.90989E-02	6.91670E-04
	Average	1.26235E-04	6.42633E+00	1.11708E-03	1.78992E-01	7.38369E-03	3.41723E-04
	Best	1.35304E-05	2.28875E+00	8.39506E-04	3.52766E-02	1.15813E-03	8.33871E-05
	STD	9.12877E-05	4.50898E+00	2.50083E-04	1.63285E-01	7.49709E-03	2.90804E-04
	Rank	2	5	3	6	4	1

Table 1 (continued)

Function	Measure	AOA	GOA	GWO	HGSO	SCA	AOAHGS
F8	Rank	1	6	3	5	4	2
	Worst	-2.18677E+03	-1.83890E+03	-1.79780E+03	-1.98754E+03	-1.91076E+03	-3.00544E+03
	Average	-2.40069E+03	-2.26483E+03	-2.25110E+03	-2.40345E+03	-1.97698E+03	-3.73963E+03
	Best	-2.81135E+03	-2.43279E+03	-2.67754E+03	-2.79551E+03	-2.05739E+03	-4.18983E+03
	STD	2.511460E+02	2.45972E+02	4.17228E+02	3.02232E+02	5.38981E+01	6.17593E+02
F9	Rank	3	4	5	2	6	1
	Worst	0.00000E+00	1.25931E+02	3.21492E+00	8.96384E+01	1.30282E+00	0.00000E+00
	Average	0.00000E+00	7.35914E+01	8.49761E-01	5.73490E+01	2.60605E-01	0.00000E+00
	Best	0.00000E+00	4.37674E+01	0.00000E+00	3.50060E+01	1.42109E-14	0.00000E+00
	STD	0.00000E+00	3.20877E+01	1.39590E+00	2.10296E+01	5.82617E-01	0.00000E+00
F10	Rank	1	6	4	5	3	1
	Worst	8.88178E-16	1.96342E+01	2.22045E-14	1.26893E+01	1.12689E-01	8.88178E-16
	Average	8.88178E-16	1.92970E+01	1.65201E-14	8.22839E+00	2.25428E-02	8.88178E-16
	Best	8.88178E-16	1.84945E+01	7.99361E-15	4.46026E+00	1.72466E-08	8.88178E-16
	STD	0.00000E+00	4.59479E-01	5.94483E-15	3.05926E+00	5.03933E-02	0.00000E+00
F11	Rank	1	6	3	5	4	1
	Worst	6.82787E-14	9.04040E-01	8.63303E-02	2.63019E+00	1.61302E-01	0.00000E+00
	Average	1.37224E-14	6.74002E-01	2.97356E-02	1.76213E+00	7.22649E-02	0.00000E+00
	Best	0.00000E+00	5.30865E-01	0.00000E+00	1.16468E+00	1.78746E-14	0.00000E+00
	STD	3.04983E-14	1.90531E-01	3.33870E-02	5.91856E-01	7.12913E-02	0.00000E+00
F12	Rank	2	5	3	6	4	1
	Worst	1.50286E-01	6.27173E+02	1.50464E-01	2.83698E+01	3.33426E-01	3.49009E-07
	Average	1.05198E-01	1.52369E+02	6.71897E-02	9.13396E+00	2.28425E-01	9.15641E-08
	Best	7.97403E-02	1.43005E+01	1.91291E-02	2.56415E+00	1.62252E-01	2.03968E-09
	STD	2.83448E-02	2.65957E+02	5.24643E-02	1.08762E+01	6.39320E-02	1.45004E-07
F13	Rank	3	6	2	5	4	1
	Worst	9.95596E-01	1.34764E+02	5.06146E-01	2.23621E+01	5.81808E-01	1.05542E-06
	Average	8.77911E-01	3.64575E+01	3.24561E-01	7.98428E+00	4.97203E-01	3.06028E-07
	Best	7.65564E-01	2.80247E+00	6.13289E-02	8.08835E-01	4.54402E-01	2.39348E-09
	STD	9.02201E-02	5.56133E+01	1.65522E-01	8.44449E+00	5.24461E-02	4.34423E-07
Rank	4	6	2	5	3	1	
Summation	29	72	39	66	51	14	
Mean Ranking	2.23	5.54	3.00	5.08	3.92	1.08	
Final Ranking	2	6	3	5	4	1	

Table 2 The comparative methods result using thirteen benchmark functions (F1–F13), where the dimension is fixed to 100

Function	Measure	AOA	GOA	GWO	HGSO	SCA	AOAHGS
F1	Worst	4.53685E-31	5.54643E+02	1.64849E-22	1.44893E+04	1.20841E+02	7.36361E-69
	Average	9.07371E-32	4.26866E+02	4.07113E-23	6.46612E+03	6.04652E+01	1.47272E-69
	Best	8.37161E-54	3.81499E+02	2.79572E-24	8.68307E+02	2.76365E-01	0.00000E+00
	STD	2.02894E-31	7.26815E+01	6.97436E-23	6.16896E+03	5.58187E+01	3.29310E-69
	Rank	2	5	3	6	4	1
F2	Worst	1.18712E-145	1.26743E+06	1.13483E-13	1.00084E+02	1.39557E-02	0.00000E+00
	Average	2.37423E-146	2.54965E+05	6.46130E-14	5.3596E+01	3.53181E-03	0.00000E+00
	Best	2.29622E-180	1.41039E+02	1.61224E-14	1.18691E+01	3.51964E-05	0.00000E+00
	STD	5.30895E-146	5.65987E+05	4.10810E-14	3.94908E+01	5.98522E-03	0.00000E+00
	Rank	2	6	3	5	4	1
F3	Worst	2.15068E-02	1.67634E+04	4.53837E-02	8.54061E+04	4.04734E+04	6.76655E-28
	Average	5.37564E-03	1.11774E+04	1.27010E-02	3.77977E+04	2.38611E+04	1.35331E-28
	Best	8.82101E-22	7.37221E+03	2.31636E-04	1.29245E+04	4.58495E+03	0.00000E+00
	STD	9.25000E-03	3.70117E+03	1.87375E-02	2.85952E+04	1.39829E+04	3.02609E-28
	Rank	2	4	3	6	5	1
F4	Worst	5.26751E-02	4.52209E+01	5.23957E-05	5.25673E+01	4.61991E+01	2.86463E-13
	Average	4.70464E-02	3.94517E+01	2.41575E-05	3.43672E+01	3.37732E+01	5.72927E-14
	Best	4.20117E-02	2.90696E+01	2.55393E-06	2.43239E+01	2.00748E+01	2.11266E-26
	STD	5.19427E-03	6.16211E+00	1.89609E-05	1.07891E+01	1.18049E+01	1.28110E-13
	Rank	3	6	2	5	4	1
F5	Worst	2.89515E+01	1.42474E+07	2.88605E+01	1.37731E+07	3.61917E+05	2.87580E+01
	Average	2.86669E+01	6.98177E+06	2.82582E+01	3.03818E+06	7.33209E+04	1.67028E+01
	Best	2.80068E+01	3.21291E+06	2.71791E+01	1.04830E+04	5.33974E+01	1.38879E-02
	STD	3.75581E-01	4.24138E+06	6.73657E-01	6.00455E+06	1.61339E+05	1.52115E+01
	Rank	3	6	2	5	4	1
F6	Worst	4.37310E+00	5.07071E+02	4.00890E+00	1.46797E+04	2.29306E+01	1.02448E-03
	Average	3.90684E+00	3.32430E+02	3.35487E+00	5.44202E+03	1.70536E+01	4.14930E-04
	Best	3.38327E+00	1.69824E+02	2.25824E+00	9.05175E+02	5.22152E+00	3.18659E-06
	STD	3.57361E-01	1.28240E+02	7.33696E-01	6.06300E+03	7.63722E+00	4.14761E-04
	Rank	3	5	2	6	4	1
F7	Worst	1.24357E-03	1.24563E+02	9.00609E-03	8.00005E+00	4.24051E-01	1.07044E-03
	Average	4.79589E-04	8.80772E+01	6.51046E-03	3.11823E+00	1.49364E-01	4.49396E-04
	Best	1.42432E-05	4.85634E+01	3.78870E-03	4.83134E-01	4.26948E-02	1.41917E-04
	STD	5.43068E-04	3.45892E+01	2.06748E-03	3.43379E+00	1.63801E-01	3.63183E-04
	Rank	3	5	2	6	4	1

Table 2 (continued)

Function	Measure	AOA	GOA	GWO	HGSO	SCA	AOAHGS
F8	Rank	2	6	3	5	4	1
	Worst	-3.84211E+03	-5.89745E+03	-3.80514E+03	-2.94427E+03	-3.10445E+03	-7.57789E+03
	Average	-4.53830E+03	-6.50268E+03	-5.22717E+03	-3.66619E+03	-3.45544E+03	-1.08023E+04
	Best	-4.81338E+03	-6.96869E+03	-5.82431E+03	-4.18454E+03	-4.24500E+03	-1.25336E+04
	STD	4.14772E+02	4.76701E+02	8.14274E+02	4.87805E+02	4.52774E+02	2.34338E+03
F9	Rank	4	2	3	5	6	1
	Worst	0.00000E+00	3.14614E+02	9.97831E+00	2.70109E+02	7.18926E+01	0.00000E+00
	Average	0.00000E+00	2.87806E+02	5.71892E+00	2.54266E+02	3.85261E+01	0.00000E+00
	Best	0.00000E+00	2.51490E+02	1.38795E+06	2.44391E+02	1.77451E+00	0.00000E+00
	STD	0.00000E+00	3.22269E+01	3.86433E+00	9.63256E+00	3.16678E+01	0.00000E+00
F10	Rank	1	6	3	5	4	1
	Worst	8.88178E-16	2.02681E+01	4.30322E-12	1.85262E+01	2.03266E+01	8.88178E-16
	Average	8.88178E-16	1.89989E+01	2.55387E-12	1.50245E+01	1.50569E+01	8.88178E-16
	Best	8.88178E-16	1.71464E+01	9.67226E-13	1.29034E+01	8.35861E-01	8.88178E-16
	STD	0.00000E+00	1.52075E+00	1.43515E-12	2.13803E+00	8.20850E+00	0.00000E+00
F11	Rank	1	6	3	4	5	1
	Worst	8.29231E-01	1.98704E+00	1.59251E-02	1.31167E+02	9.24854E-01	0.00000E+00
	Average	6.94886E-01	1.52159E+00	3.18502E-03	5.84504E+01	6.72642E-01	0.00000E+00
	Best	3.92307E-01	1.21310E+00	0.00000E+00	2.99281E+01	7.54713E-02	0.00000E+00
	STD	1.81393E-01	3.06350E-01	7.12193E-03	4.20904E+01	3.45552E-01	0.00000E+00
F12	Rank	4	5	2	6	3	1
	Worst	7.38370E-01	1.10622E+06	3.73157E-01	6.17878E+05	1.57243E+01	2.55430E-04
	Average	6.99570E-01	3.59284E+05	2.05408E-01	1.60276E+05	4.96976E+00	9.66020E-05
	Best	6.67213E-01	4.90887E+03	5.92649E-02	2.69753E+01	9.39148E-01	3.41405E-06
	STD	3.28205E-02	4.68265E+05	1.13686E-01	2.67594E+05	6.12608E+00	1.16868E-04
F13	Rank	3	6	2	5	4	1
	Worst	2.93956E+00	2.66856E+06	1.95530E+00	3.94376E+07	2.04846E+04	1.98923E+00
	Average	2.91929E+00	1.07974E+06	1.73510E+00	1.40827E+07	6.10793E+03	7.96744E-01
	Best	2.90512E+00	4.42330E+05	1.39915E+00	3.20541E+01	2.10042E+01	1.39833E-04
	STD	1.73442E-02	9.34940E+05	2.14695E-01	1.57714E+07	8.81996E+03	8.32603E-01
Summation	33	68	33	69	55	13	
Mean Ranking	2.54	5.23	2.54	5.31	4.23	1.00	
Final Ranking	2	5	2	6	4	1	

Table 3 The results of the comparative methods using ten benchmark functions (F14-F23)

Function	Measure	AOA	GOA	GWO	HGSO	SCA	AOAHGS
F14	Worst	1.26705E+01	1.45631E+01	1.26705E+01	1.55040E+01	1.26705E+01	1.07632E+01
	Average	1.12759E+01	9.96477E+00	6.87046E+00	7.04191E+00	5.66700E+00	3.54865E+00
	Best	9.37532E+00	3.96825E+00	2.98211E+00	1.99203E+00	9.98004E-01	9.98098E-01
	STD	1.40589E+00	4.38384E+00	4.49329E+00	5.84989E+00	6.39329E+00	4.11596E+00
	Rank	6	5	3	4	2	1
F15	Worst	3.59937E-02	1.35205E-01	2.09531E-02	2.04092E-02	1.09868E-02	1.50667E-03
	Average	1.21701E-02	6.98171E-02	8.70898E-03	9.03057E-03	2.62936E-03	1.24419E-03
	Best	4.43290E-04	1.98066E-02	5.15881E-04	1.13419E-03	3.08481E-04	6.75591E-04
	STD	1.65913E-02	5.18184E-02	1.10772E-02	1.03683E-02	4.68868E-03	3.51707E-04
	Rank	5	6	3	4	2	1
F16	Worst	-1.03163E+00	-2.15464E-01	-1.03163E+00	-1.03143E+00	-1.03152E+00	-1.03163E+00
	Average	-1.03163E+00	-8.68396E-01	-1.03163E+00	-1.03157E+00	-1.03158E+00	-1.03163E+00
	Best	-1.03163E+00	-1.03163E+00	-1.03163E+00	-1.03163E+00	-1.03163E+00	-1.03163E+00
	STD	5.16806E-07	3.65000E-01	4.29778E-08	8.31902E-05	4.78788E-05	1.11022E-16
	Rank	3	6	2	5	4	1
F17	Worst	4.42045E-01	3.97901E-01	3.97891E-01	3.98011E-01	5.04011E+00	3.97887E-01
	Average	4.22839E-01	3.97893E-01	3.97889E-01	3.97918E-01	2.26222E+00	3.97887E-01
	Best	4.13694E-01	3.97888E-01	3.97888E-01	3.97887E-01	3.98343E-01	3.97887E-01
	STD	1.13534E-02	5.30993E-06	1.67891E-06	5.29294E-05	2.53588E+00	7.54690E-14
	Rank	5	3	2	4	6	1
F18	Worst	1.94299E+02	3.00000E+01	8.40001E+01	8.40003E+01	3.00228E+00	3.00000E+00
	Average	8.86975E+01	1.38000E+01	1.92002E+01	1.92002E+01	3.00047E+00	3.00000E+00
	Best	3.00000E+00	3.00000E+00	3.00000E+00	3.00000E+00	3.00000E+00	3.00000E+00
	STD	8.62048E+01	1.47885E+01	3.62243E+01	3.62244E+01	1.01220E-03	9.74470E-15
	Rank	6	3	4	5	2	1
F19	Worst	-3.81139E+00	-8.71991E-01	-3.85584E+00	-3.67848E+00	-3.83161E+00	-3.86278E+00
	Average	-3.84151E+00	-2.64950E+00	-3.85946E+00	-3.80166E+00	-3.84918E+00	-3.86278E+00
	Best	-3.85230E+00	-3.83495E+00	-3.86254E+00	-3.86252E+00	-3.85487E+00	-3.86278E+00
	STD	1.70039E-02	1.16419E+00	3.36128E-03	8.36826E-02	9.89532E-03	3.02337E-09
	Rank	4	6	2	5	3	1
F20	Worst	-2.94365E+00	-1.90960E+00	-2.43176E+00	-2.95524E+00	-1.14725E+00	-3.01561E+00
	Average	-3.04388E+00	-2.77527E+00	-3.08637E+00	-3.21291E+00	-1.83837E+00	-3.17530E+00
	Best	-3.09189E+00	-3.19441E+00	-3.32199E+00	-3.31606E+00	-2.99379E+00	-3.32200E+00
	STD	5.84620E-02	5.37738E-01	3.73339E-01	1.57457E-01	7.23290E-01	1.12230E-01
	Rank	4	6	2	5	3	1

Table 3 (continued)

Function	Measure	AOA	GOA	GWO	HGSO	SCA	AOAHGS
F21	Rank	4	5	3	1	6	2
	Worst	-1.64517E+00	-3.48624E-01	-5.05517E+00	-3.06985E-01	-3.50654E-01	-5.10077E+00
	Average	-2.58539E+00	-2.73710E+00	-8.11946E+00	-4.62737E+00	-5.14612E-01	-9.14271E+00
	Best	-3.63920E+00	-5.10077E+00	-1.01509E+01	-1.01424E+01	-8.77978E-01	-1.01532E+01
	STD	8.00423E-01	2.31366E+00	2.77874E+00	3.65582E+00	2.15887E-01	2.25951E+00
F22	Rank	5	4	2	3	6	1
	Worst	-1.81126E+00	-7.20249E-01	-2.73694E+00	-3.74816E-01	-5.08767E+00	-1.03974E+01
	Average	-3.26096E+00	-1.58143E+00	-6.07973E+00	-2.13119E+00	-8.28506E+00	-1.04002E+01
	Best	-4.84638E+00	-3.72430E+00	-1.03438E+01	-4.87800E+00	-1.04029E+01	-1.04025E+01
	STD	1.21658E+00	1.22210E+00	3.74680E+00	2.27910E+00	2.90006E+00	2.17550E-03
F23	Rank	4	6	3	5	2	1
	Worst	-1.60567E+00	-1.19074E+00	-2.42159E+00	-1.67587E+00	-5.58458E-01	-5.12848E+00
	Average	-3.45031E+00	-2.53910E+00	-7.83085E+00	-2.81400E+00	-3.52547E+00	-7.30109E+00
	Best	-5.41125E+00	-5.12746E+00	-1.05360E+01	-4.36474E+00	-7.32496E+00	-1.05364E+01
	STD	1.52603E+00	1.52969E+00	3.82415E+00	9.81206E-01	2.60377E+00	2.95350E+00
Summation	46	50	25	41	36	12	
Mean Ranking	4.60	5.00	2.50	4.10	3.60	1.20	
Final Ranking	5	6	2	4	3	1	

results by combining the search process of two powerful methods and some limitations can be defined as other applications can be tested to further validation of the proposed method.

Further, AOA and HGS can be integrated with other existing state-of-the-art algorithms, improving the algorithm and giving more precise results with less computational time, which is needed in real-time applications and problems.

Data and code availability Not Applicable

Funding The authors received no specific funding for this study.

Declarations

Conflict of interest Authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Abualigah L, Dulaimi AJ (2021) A novel feature selection method for data mining tasks using hybrid Sine Cosine Algorithm and Genetic Algorithm. *Cluster Computing* : 1–16.
2. Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomi AH (2020) The arithmetic optimization algorithm. *Comput Methods Appl Mech Eng* 376:113609
3. Alshaer HN, Otair MA, Abualigah L, Alshinwan M, Khasawneh AM (2021) Feature selection method using improved CHI Square on Arabic text classifiers: analysis and application. *Multimed Tools Appl* 80(7):10373–10390
4. Altabeeb AM, et al. (2021) Solving capacitated vehicle routing problem using cooperative firefly algorithm. *Applied Soft Computing* 108 (2021): 107403
5. Bansal M, Kumar M, Kumar M, Kumar K (2021) An efficient technique for object recognition using Shi-Tomasi corner detection algorithm. *Soft Comput* 25:4423–4432. <https://doi.org/10.1007/s00500-020-05453-y>
6. Bansal M, Kumar M, Kumar M (2021) 2D object recognition: a comparative analysis of SIFT, SURF and ORB feature descriptors. *Multimed Tools Appl* 80:18839–18857. <https://doi.org/10.1007/s11042-021-10646-0>
7. Bansal M, Kumar M, Sachdeva M, Mittal A (2021) Transfer learning for image classification using VGG19: Caltech-101 image data set. *J Ambient Intell Human Comput*. <https://doi.org/10.1007/s12652-021-03488-z>
8. Boussaïd I, Lepagnot J, Siarry P (2013 Jul 10) A survey on optimization metaheuristics. *Inf Sci* 237:82–117
9. Chhabra P, Garg NK, Kumar M (2020) Content-based image retrieval system using ORB and SIFT features. *Neural Comput & Applic* 32:2725–2733. <https://doi.org/10.1007/s00521-018-3677-9>
10. Coello CA (2002 Jan 4) Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Comput Methods Appl Mech Eng* 191(11–12):1245–1287
11. Colomi A, Dorigo M, Maniezzo V (1991) Distributed optimization by ant colonies. In *Proceedings of the first European conference on artificial life* 142, 134–142
12. Cuevas E, Echavarría A, Zaldivar D, Pérez-Cisneros M (2013 Nov 15) A novel evolutionary algorithm inspired by the states of matter for template matching. *Expert Syst Appl* 40(16):6359–6373
13. Cuevas E, Echavarría A, Ramírez-Ortegón MA (2014 Mar 1) An optimization algorithm inspired by the states of matter that improves the balance between exploration and exploitation. *Appl Intell* 40(2):256–272
14. Davis L (1991) Bit-climbing, representational bias, and test suit design. In *Proc. Intl. Conf. Genetic algorithm*, (pp. 18–23)
15. Eberhat R, Kennedy J (1995) A new optimizer using particle swarm theory. In *Sixth international symposium on micro machine and human science, Piscataway* (pp. 39–43)
16. Eid A, Kamel S, Abualigah L (2021) Marine predators algorithm for optimal allocation of active and reactive power resources in distribution networks. *Neural Comput & Applic* 33:1–29
17. Elaziz A, Mohamed LA, Attiya A (2021) Advanced optimization technique for scheduling IoT tasks in cloud-fog computing environments. *Future Generation Computer Systems*

18. Fogel, David B (1998) Artificial intelligence through simulated evolution. Wiley-IEEE Press
19. Ghosh KK et al (2021) Theoretical and empirical analysis of filter ranking methods: Experimental study on benchmark DNA microarray data. *Expert Systems with Applications* 169:114485
20. Glover F (1989 Aug) Tabu search—part I. *ORSA J Comput* 1(3):190–206
21. Gogna A, Tayal A (2013 Dec 1) Metaheuristics: review and application. *J Exp Theoretic Artific Intell* 25(4): 503–526
22. Hassan, Mohamed H., et al. (2021) Development and application of slime mould algorithm for optimal economic emission dispatch. *Expert Systems with Applications*: 115205
23. Holland JH (1992) Genetic algorithms. *Sci Am* 267(1):66–73
24. Kaur H, Kumar M (2021) Performance evaluation of various feature selection techniques for offline handwritten Gurumukhi place name recognition. In: Singh T.P., Tomar R., Choudhury T., Perumal T., Mahdi H.F. (eds) data driven approach towards disruptive technologies. Studies in autonomic, data-driven and industrial computing. Springer, Singapore. https://doi.org/10.1007/978-981-15-9873-9_44
25. Kaveh A, Farhoudi N (2013 May 1) A new optimization method: dolphin echolocation. *Adv Eng Softw* 59: 53–70
26. Kaveh A, Farhoudi N (2016 Mar 1) Dolphin monitoring for enhancing metaheuristic algorithms: layout optimization of braced frames. *Comput Struct* 165:1–9
27. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) Optimization by simulated annealing science 220(4598):671–680
28. Li J, Lin J (2020 May 1) A probability distribution detection based hybrid ensemble QoS prediction approach. *Inf Sci* 519:289–305
29. Li J, Zheng XL, Chen ST, Song WW, Chen DR (2014 Jun 10) An efficient and reliable approach for quality-of-service-aware service composition. *Inf Sci* 269:238–254
30. Li J, Chen C, Chen H, Tong C (2017 Jul 1) Towards context-aware social recommendation via individual trust. *Knowl-Based Syst* 127:58–66
31. Liu E, Lv L, Yi Y, Xie P (2019 Jun 24) Research on the steady operation optimization model of natural gas pipeline considering the combined operation of air coolers and compressors. *IEEE Access* 7:83251–83265
32. Lourenço HR, Martin OC, Stützle T (2003) Iterated local search. In *Handbook of metaheuristics 2003* (pp. 320–353). Springer, Boston, MA
33. Mahajan S, Pandit AK (2021) Hybrid method to supervise feature selection using signal processing and complex algebra techniques. *Multimed Tools Appl*. <https://doi.org/10.1007/s11042-021-11474-y>
34. Mahajan S, Mittal N, Pandit AK (2021) Image segmentation using multilevel thresholding based on type II fuzzy entropy and marine predators algorithm. *Multimed Tools Appl* 26:1–25
35. Mahajan S, Abualigah L, Pandit AK, Altalhi M (2022) Hybrid Aquila optimizer with arithmetic optimization algorithm for global optimization tasks. *Soft Comput*. <https://doi.org/10.1007/s00500-022-06873-8>
36. Marler RT, Arora JS (2004 Apr 1) Survey of multi-objective optimization methods for engineering. *Struct Multidiscip Optim* 26(6):369–395
37. Michalewicz Z. Genetic algorithms+ data structures= evolution programs. Springer Science & Business Media; 2013.
38. Mirjalili S, Lewis A (2013 Apr 1) S-shaped versus V-shaped transfer functions for binary particle swarm optimization. *Swarm and Evolutionary Computation* 9:1–4
39. Mirjalili S, Mirjalili SM, Lewis A (2014 Mar 1) Grey wolf optimizer. *Adv Eng Softw* 69:46–61
40. Pang J, Zhou H, Tsai YC, Chou FD (2018 Sep 1) A scatter simulated annealing algorithm for the bi-objective scheduling problem for the wet station of semiconductor manufacturing. *Comput Ind Eng* 123:54–66
41. Şahin, Canan Batur, and Laith Abualigah (2021) A novel deep learning-based feature selection model for improving the static analysis of vulnerability detection. *Neural Comput Appl*: 1–19.
42. Saremi S, Mirjalili S, Lewis A (2017 Mar 1) Grasshopper optimisation algorithm: theory and application. *Adv Eng Softw* 105:30–47
43. Sharaff, Aakanksha, et al. (2020) personalized recommendation system with user interaction based on LMF and popularity model. 2020 International Conference on System, Computation, Automation and Networking (ICSCAN). IEEE,
44. Sharma, Dimple, and Aakanksha Sharaff. "Identifying Spam Patterns in SMS using Genetic Programming Approach." 2019 *International Conference on Intelligent Computing and Control Systems (ICCS)*. IEEE, 2019.
45. Spall JC (2005) Introduction to stochastic search and optimization: estimation, simulation, and control. John Wiley & Sons
46. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim* 11(4):341–359

47. Walia S, Kumar K, Kumar M, Gao X-Z (2021) Fusion of handcrafted and deep features for forgery detection in digital images. *IEEE Access* 9:99742–99755. <https://doi.org/10.1109/ACCESS.2021.3096240>
48. Wen F, Yang X, Gong X, Lai KK (2017 Jan 5) Multi-scale volatility feature analysis and prediction of gold price. *International Journal of Information Technology & Decision Making* 16(01):205–223
49. Wolpert DH, Macready WG (1997 Apr) No free lunch theorems for optimization. *IEEE Trans Evol Comput* 1(1):67–82
50. Yan J, Pu W, Zhou S, Liu H, Bao Z (2020 Mar 1) Collaborative detection and power allocation framework for target tracking in multiple radar system. *Information Fusion* 55:173–183
51. Yang XS (2010) Nature-inspired metaheuristic algorithms. Luniver press
52. Yang XS (2010 Jan 1) Firefly algorithm, stochastic test functions and design optimisation. *Int J Bio-Inspired Comput* 2(2):78–84
53. Yang XS (2010) A new metaheuristic bat-inspired algorithm. In *Nature inspired cooperative strategies for optimization (NICSO 2010) 2010* (pp. 65-74). Springer, Berlin, Heidelberg
54. Yang XS (2010) Nature-inspired metaheuristic algorithms. Luniver press
55. Yang XS (2012) Flower pollination algorithm for global optimization. In *International conference on unconventional computing and natural computation 2012 Sep 3* (pp. 240-249). Springer, Berlin, Heidelberg
56. Yang XS, Deb S (2010 Jan 1) Engineering optimisation by cuckoo search. *International Journal of Mathematical Modelling and Numerical Optimisation* 1(4):330–343
57. Yang XS, Bramer M, Ellis R, Petridis M (2010) *Research and development in intelligent systems XXVI. Development*, Springer
58. Yang Y, Chen H, Heidari AA, Gandomi AH (2021 Mar) Hunger games search: visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. *Expert Syst Appl* 10:114864
59. Zhou A, Qu BY, Li H, Zhao SZ, Suganthan PN, Zhang Q (2011 Mar 1) Multiobjective evolutionary algorithms: a survey of the state of the art. *Swarm and Evolutionary Computation*. 1(1):32–49

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.