Multimedia Tools and Applications (2024) 83:32613–32653
https://doi.org/10.1007/s11042-023-16890-w

Improved prairie dog optimization algorithm by dwarf mongoose optimization algorithm for optimization problems

Laith Abualigah[1,2,3,4,5,6,7] · Diego Oliva[8] · Heming Jia[9] · Faiza Gul[10] · Nima Khodadadi[11] · Abdelazim G Hussien[12,13] · Mohammad Al Shinwan[14] · Absalom E. Ezugwu[15] · Belal Abuhaija[16] · Raed Abu Zitar[17]

Received: 21 September 2022 / Revised: 21 June 2023 / Accepted: 4 September 2023 /
Published online: 22 September 2023
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract
Recently, optimization problems have been revised in many domains, and they need powerful search methods to address them. In this paper, a novel hybrid optimization algorithm is proposed to solve various benchmark functions, which is called IPDOA. The proposed method is based on enhancing the search process of the Prairie Dog Optimization Algorithm (PDOA) by using the primary updating mechanism of the Dwarf Mongoose Optimization Algorithm (DMOA). The main aim of the proposed IPDOA is to avoid the main weaknesses of the original methods; these weaknesses are poor convergence ability, the imbalance between the search process, and premature convergence. Experiments are conducted on 23 standard benchmark functions, and the results are compared with similar methods from the literature. The results are recorded in terms of the best, worst, and average fitness function, showing that the proposed method is more vital to deal with various problems than other methods.

Keywords Prairie dog optimization algorithm · Dwarf mongoose optimization algorithm · Meta-heuristics · Benchmark functions · Optimization problems

1 Introduction

Selecting the best configuration of parameters of a particular problem by either minimizing cost/time or maximizing the profit can be termed/defined as Optimization. Real-world global and engineering issues can be classified into continuous or discrete, unconstrained or constrained, multi-objective or single objective, and dynamic or static [1–5]. Mathemat-

✉ Laith Abualigah
aligah.2020@gmail.com

✉ Belal Abuhaija
babuhaij@kean.edu

Extended author information available on the last page of the article

ical techniques are insufficient to solve real-world problems since they fall in local optima, consume more time, etc. Therefore, alternative approximation-based techniques such as heuristic/metaheuristic are good for handling current problems [6–8].

Metaheuristics algorithms are optimization techniques that can perform the search space iteratively by using the concept of natural selection "survival-for-the-best" using two approaches: exploration and exploitation [9, 10]. Exploration means the agents (solutions) will move globally to various and several search areas. In contrast, exploitation means that the agents (solutions) will alter themselves reasonably by moving in the promising areas [11–14]. Metaheuristic algorithms have become very popular/famous for many reasons: avoiding local optimal solutions, flexibility, ease-of-use, simplicity, etc [15–17]. In literature, an enormous number of metaheuristics optimizers have been raised which can be classified into three groups [18–21]: (1) swarm-inspired, (2) evolution-inspired, and (3) physics-based.

Swarm-inspired algorithms: contains optimizers that mimic species' social habits such as navigation, labor division, mating, and self-organization. Examples of this algorithm type includes Particle swarm optimization [22], Cuckoo Search [23], Virus Colony Search (VCS) [24], Grasshopper Optimization Algorithm [25], Social Spider Optimization [26], Ant Lion Optimizer [27], Snake Optimizer [28], Remora Optimization Algorithm [29], Wild Horse Optimizer [30], Lightning search algorithm [31], Crow search algorithm [32], Arithmetic Optimization Algorithm [33], Aquila Optimizer [34], Moth-flame Optimization [35], Reptile Search Algorithm (RSA) [36], and Grey Wolf Optimizer [37]. Evolution-inspired algorithms: contains optimizers that implement the biology evolution's ideas and concepts. Examples of this algorithm type include Genetic Algorithms (GA) [38], Bacterial Foraging Optimization [39], Differential Evolution (DE) [40], Artificial Algae Algorithm [41], and biogeography-based Optimization (BBO) [42], etc. Physics-based algorithms contain algorithms that implement physics laws and phenomena. Examples of this algorithm type include Simulated Annealing (SA) [43], Big-Bang-Big Crunch (BBBC) [44], Ray Optimization (RO), Black Hole (BH) [45], Gravitational Search Algorithm (GSA) [46], Multi-Verse Optimizer (MVO) [47], and Henry Gas Solubility Optimizer (HGSO) [48]. The dwarf mongoose optimization algorithm is a swarm-inspired population-based optimizer developed by Agushaka et al. [49], which simulates the habits of dwarf mongoose in foraging. The dwarf mongoose population is divided into alpha, scout, and babysitters groups. Prairie dog optimization algorithm [50] is another novel algorithm proposed by Ezugwu et al. in which the four different activities of prairie dogs are implemented, namely: prairie dogs burrow build, foraging, alert sound, and unique communication.

The prairie dog optimization algorithm has many disadvantages, such as stagnation in local optima, the imbalance between diversification & intensification, and a slow convergence curve. To overcome the PDO algorithm, IPDO is proposed by updating its individual using the DMOA mechanism.

The major contributions of this paper can be listed as follows:

- Enhanced version of PDO is introduced by hybridizing DMOA and PDO.
- Twenty-three classical mathematical functions are used to test IPDO performance.
- IPDO is compared with standard DMOA, standard PDO, and seven other algorithms.

In the rest of this paper, Section 2 shows some literature-related work. In contrast, Section 3 shows the mathematical equations for both algorithms: Dwarf mongoose optimization algorithm (DMOA) & Prairie dog optimization (PDO), and the developed algorithm IPDO.

Section 4 shows the results discussion over 23 functions, whereas the paper's conclusion is proposed in Section 5.

2 Related works

Due to the stochastic nature of the meta-heuristic algorithms' search process, the solutions may or may not be close to the known global optimum. It is advised to execute the algorithm independently numerous times and pick the average result for comparison because the results obtained constantly differ due to this randomization.

Current methods, such as the slime mould optimization algorithm (SMA), the arithmetic optimization algorithm (AOA), the salp swarm algorithm (SSA), and the marine predator's algorithm, were compared in [51]. The proposed (IANN-AOA) artificial neural network was boosted using the arithmetic optimization approach to help it solve the damage quantification challenge [52]. The major goal is to remove the healthy components from the numerical model using a better indicator. The damage level is output, and the damaged elements' data is gathered via an upgraded indicator's damage index. The IANN-AOA results demonstrated that the enhanced indicator is more accurate at predicting damaged parts. Damage quantification yields the same result. However, IANN-AOA yields more precise results than IANN-BCMO.

The multi-objective variant of the arithmetic optimization technique (MOAOA) was proposed by Premjumar et al. [53]. Real-world constrained multi-objective optimization problems (RWMOPs) in chemical engineering, mechanical engineering, syntheses and processes, and power electronics systems were solved using the approach. A set of 35 constrained RWMOPs and five unconstrained ZDT issues were used to assess the MOAOA's performance, and its results were compared to those of four other cutting-edge multi-objective algorithms. The MOAOA's great accuracy and coverage across all objectives prove its superiority to the other algorithms under consideration.

Ibraheem et al. [54] solved Feature Selection, a brand-new combined Meta-heuristics based method called Reptile Search Algorithm-Snake Optimizer (RSA-SO) is presented in the study. The RSA-SO technique, where both have collaborated in a parallel process, makes use of the best traits and abilities of both the RSA [36] and SO [28] algorithms to generate an ideal selection of valuable features.

The research [55] states that the current approaches can produce positive results in some situations but not in others. As a result, a fresh approach that can handle various complex problems is urgently needed. The "no free lunch theorem" compelled us to seek out and create a fresh strategy for addressing such complex problems.

In order to address optimization problems, the paper offers a novel optimization approach. The proposed method, known as GNDDMOA, is based on the application of the essential techniques of Generalized Normal Distribution Optimization (GND) and Dwarf Mongoose Optimization Algorithm (DMOA), followed by the Opposition-based Learning Mechanism (OBL). The proposed methods follow the transition techniques by defining a condition that decides which technique will be employed. It is advised to use this design in order to avoid the issue of quick convergence and preserve the diversity of viable solutions. Then, in response to a transition technique scenario, the Opposition-based Learning (OBL) Mechanism is engaged. In order to avoid becoming trapped in the local search zone, this phase is utilized to look for a new search area. Two sets of experiments-23 benchmark functions and 8 data clustering problems-are utilized to verify the effectiveness of the proposed technique.

Refracted Salp Swarm Algorithm (RSSA) was proposed by Fan et al. in [56]. The refracted opposition-based learning technique, the multi-leader mechanism, and the adaptive conversion parameter strategy all enhance the fundamental Salp Swarm Algorithm. For applications in the area of structural parameter identification, the modified RSSA performs noticeably better than the fundamental SSA method.

In order to address the drawbacks of the Harris Hawks Optimizer (HHO) in the exploration phase, where population diversity is limited and quickly slips into local optimum, Zhang et al. in [57] offer an improved Harris hawks optimization based on adaptive cooperative foraging and dispersed foraging tactics. Additionally, some researchers/authors believe combining more than two algorithms will improve the algorithm's performance. For instance, Kamboj et al. [58] used the sine cosine algorithm to create a hybrid version of the Harris Hawks optimizer known as the hybrid Harris Hawks-Sine Cosine Algorithm in order to speed up the global search phase of the existing Harris hawks Optimization and escape from local search space (hHHO-SCA).

In order to solve benchmark optimization functions and engineering challenges, this work suggests an enhanced version of the grasshopper optimization algorithm (GOA) based on the opposition-based learning (OBL) method. This algorithm is referred to as OBLGOA [59]. The proposed OBLGOA algorithm comprises two stages: the first stage employs the OBL approach to generate an initial population and its opposite, and the second stage adds the OBL as a second phase to update the GOA population after each iteration.

We need to take into account both the characteristics and methodologies of both IPDOA and DMOA, as well as the unique enhancements and alterations performed in the hybridization process, in order to examine the distinctions between this hybrid method and earlier approaches that combine optimization algorithms. I will give a broad description of how such hybridization can result in improvements as you haven't given any specifics regarding the IPDOA and DMOA algorithms or their adjustments.

The hybrid technique seeks to utilize the complimentary capabilities of two different optimization algorithms. Different search operators, techniques, or processes may be used by each algorithm to explore and take advantage of the search space. The process of hybridization enables a wider range of exploration and exploitation, which may improve convergence and prevent local maxima.

It's vital to remember that a hybrid method's success depends on several variables, including the algorithms' compatibility, the appropriate choice and integration of each component, and careful parameter tweaking. On a wide range of optimization issues, the success of the hybrid technique should be assessed by thorough testing and comparison with other approaches already in use. It is difficult to talk about the special distinctive differentiating features of this particular hybrid approach without precise information on the IPDOA, DMOA, or the changes brought in their merging. The broad advantages mentioned above, however, establish a framework for comprehending the possible benefits of such hybridization in solving optimization problems.

3 The proposed method

In this section, the original methods used in the proposed method are presented, and the main procedure of the proposed method is presented as well.

3.1 Prairie dog optimization algorithm

The prairie dog optimization (PDO) is a new metaheuristic proposed by Ezugwu et al. [50] in 2022 inspired by the natural behavior of prairie dogs. This algorithm has four parameters called n, m, p, ϵ that should be set at the start of the algorithm. Which is the specialized food source alarm, and ϵ is a small number that represents the food source quality. Next, like other metaheuristics, the PDO method begins with a random population (n) of solutions (number of prairie dogs). In addition, each of these populations sets to m group (coterie). The objective value of the solutions is determined at each iteration. The suggested method accomplishes exploration and exploitation via four prairie dog activities to switch between exploration and exploitation.

Foraging and burrowing activities belong to the exploration phase, and the following formulas are used:

a) For $iter < \frac{Max_{iter}}{4}$, the foraging activities will be active as follows:

$$PD_{i+1,j+1} = GBest_{i,j} - eCBest_{i,j} \times \rho - CPD_{i,j} \times Levy(n) \tag{1}$$

where $PD_{i+1,j+1}$ denotes to position updating, $GBest_{i,j}$ is global best and ρ is fixed to 0.1. $Levy(n)$ is a $Levy$ distribuation. In addition, $eCBest_{i,j}$ examines the impact of the most recent optimal solution that was obtained as follows:

$$eCBest_{i,j} = GBest_{i,j} \times \triangle + \frac{PD_{i,j} \times mean(PD_{n,m})}{GBest_{i,j} \times (UB_i - LB_j) + \triangle}$$

where UB_i and LB_j are upper and lower bounds, respectively. to position updating, In spite of the fact that the prairie dogs are distinct from one another, the PDO implementation treated them as though they were identical. This is represented by the small number $\triangle$. $PD_{i,j}$ is the jth dimension of the ith prairie dog in a coterie, and the value $CPD_{i,j}$ represents the random cumulative effect of all prairie dogs in the colony. $PD_{i,j}$ and $CPD_{i,j}$ equations are defined as followings:

$$PD_{i,j} = U(0,1) \times (\frac{UB_j}{m} - \frac{LB_j}{m}) + \frac{LB_j}{m} \tag{2}$$

where $U(0, 1)$ creates the random number between 0 and 1 with a uniform distribution.

$$CPD_{i,j} = \frac{GBest_{i,j} - rPD_{i,j}}{GBest_{i,j} + \triangle} \tag{3}$$

where $rPD_{i,j}$ denoted to random solution's position.

b) For $\frac{Max_{iter}}{4} \leq iter < \frac{Max_{iter}}{2}$, burrowing activities will be active as follows:

$$PD_{i+1,j+1} = GBest_{i,j} \times eCBest_{i,j} \times DS \times Levy(n) \tag{4}$$

where DS is the coterie's digging strength, obtained by the following:

$$DS = 1.5 \times r \times (1 - \frac{iter}{Max_{iter}})^{(\frac{2 \times iter}{Max_{iter}})} \tag{5}$$

Food and antipredation alarms belong to the exploitation phase, and the following formulas are used:

a) For $\frac{Max_{iter}}{2} \leq iter < \frac{3Max_{iter}}{4}$, the food alarm will be active as follows:

$$PD_{i+1,j+1} = GBest_{i,j} - eCBest_{i,j} \times \epsilon - CPD_{i,j} \times rand \tag{6}$$

where ϵ is an insignificant number that indicates the overall quality of the food source and $rand$ is the random number between 0 and 1.

b) $\frac{3Max_{iter}}{4} \leq iter \leq Max_{iter}$, the antipredation alarm will be active as follows:

$$PD_{i+1,j+1} = GBest_{i,j} \times PE \times \times rand \tag{7}$$

where PE is the predator effect, obtained by the following:

$$PE = r \times (1 - \frac{iter}{Max_{iter}})^{(\frac{2\times iter}{Max_{iter}})} \tag{8}$$

The algorithm uses its established processes in a repeating manner so that it can investigate all of the potential positions of near-optimal solutions. Each time the algorithm completes a loop, it will have found a better answer by that point and replace the one it had previously found following the rule. Finally, In this given study, the PDO method is terminated based on the maximum possible iterations.

3.2 Dwarf mongoose optimization algorithm

The novel Dwarf Mongoose Optimization algorithm (DMOA) model is introduced in this paper [49]. This algorithm mimics the dwarf mongoose's foraging habit. The starting value for a group of solutions is often set by DMOA using the following formulas.

3.3 Alpha group

Once the population has been started, the fitness of each solution is calculated. The likelihood value for each population's fitness and the alpha female (α) is selected based on the following probability.

$$\alpha = \frac{fit_i}{\sum_{i=1}^{n} fit_i} \tag{9}$$

The n corresponds to mongooses in the (α) group. Where the bs refer to the babysitters, the peep is the dominant female's vocalization that keeps the family on track. Equation (10) expresses the solutions updating approach.

$$X_{i+1} = X_i + ph_i \times peep \tag{10}$$

where the ϕ (random integer between 1 and -1) donates the mongoose sleeping mound as a set of the sleeping mound, (11) represents the sleeping mound for each repeat.

$$si_m = \frac{fit_{i+1} - fit_i}{max\left[|fit_{i+1} - fit_i|\right]} \tag{11}$$

The sleeping mound's average weight express in (12)

$$\varphi = \frac{\sum_{i=1}^{n} si_m}{n} \tag{12}$$

The algorithm accelerates to the scouting stage when the next food source or sleeping mound is considered after the babysitting exchange requirement is satisfied.

3.4 Scout group

Scouts look for the next sleeping mound to ensure exploration to avoid the previous sleeping mound. This movement is based on locating a new sleeping mound, whether successful or unsuccessful. Furthermore, the mongooses' migration depends on how well they do overall. The justification is that the group will discover a fresh sleeping mound if they forage sufficiently. Equation (13) simulated the scout mongoose.

$$X_{i+1} = \begin{cases} X - CF \times phi \times rand\left[X - \overrightarrow{M}\right], if\varphi_{i+1} > \varphi_i \\ X + CF \times phi \times rand\left[X - \overrightarrow{M}\right], otherwise \end{cases} \tag{13}$$

The rand is a random value between 0 and 1. Conversely, the CF is computed using (14), and (15) computes the M value.

$$CF = \left(1 - \frac{iter}{Max_{iter}}\right)^{\left(2 \times \frac{iter}{Max_{iter}}\right)} \tag{14}$$

$$\overrightarrow{M} = \sum_{n}^{i=1} \frac{X_i \times sm_i}{Xi} \tag{15}$$

Babysitters are often lesser group members who look after the children and are rotated frequently so the mother, the α female, can lead the rest of the squad on daily hunting expeditions.

3.5 The procedure of the proposed IPDOA

In this section, the proposed method is presented and its procedure.

As mentioned before, the proposed method (IPDOA) aims to address the weaknesses of the original methods (i.e., Prairie Dog Optimization Algorithm (PDOA) and Dwarf Mongoose Optimization Algorithm (DMOA)); these weaknesses are poor convergence ability, the imbalance between the search process, and premature convergence. The proposed method is based on enhancing the search process of the Prairie Dog Optimization Algorithm (PDOA) by using the primary updating mechanism of the Dwarf Mongoose Optimization Algorithm (DMOA).

The proposed method's procedure is presented in Algorithm 2, mainly based on two leading search operators. The operators of the PDOA and MDOA are used according to a condition if $rand < 0.5$ to execute one of them.

Usually, when one of the optimization methods works to update the candidate solutions, the low diversity is recognized. This problem affects the performance of the search operators by reducing their performance. Moreover, when the low diversity of the used solutions happens, the imbalance between the search process will happen as the operators focus on one side of the updating mechanisms, local search or global search. Thus, the changing between the search mechanisms in the proposed method improves searchability and performance and guide to more robust solutions. This change caused to avoid premature convergence and imbalance between the search operators. The Pseudo-code of the proposed hybrid IPDOA method is presented in Algorithm 2.

Time Complexity of the proposed method: The amount of time that it takes for an optimization method to find a solution for a given issue size is referred to as the time complexity of the algorithm. It is based on a number of variables, such as the dimensionality of the search space,

Algorithm 1 The sudo code of the DMOA algorithm.

Input: Choose initial parameters and conditions for the algorithm.
Generate the parameters and population.

while $iter < Max_{iter}$ **do**
for $iter = 1 : Solutions$ **do**
Compute the Fitness value of Mongoose.

Assigned time counter C

Determine the value of alpha using (9).

Obtain a candidate solution by using (10).

Estimate the sleeping mound by using (11).

Specify the average value of the sleeping mound by using (12).

Choose the movement vector by using (15).

Simulate the following solution of scout Mongoose by using (13)

end for

end while
t=t+1

return The best solution.

end

the difficulty of the issue, the rate of convergence, and the effectiveness of the algorithm's operations (such as fitness evaluation, solution updating, exploration, and exploitation). The precise adjustments made to IPDOA and DMOA throughout the hybridization process may have an impact on their time complexity and, as a result, the hybrid algorithm's overall complexity.

4 Experiments and results

In this section, the results of the comparative methods are presented. The proposed method is compared with other comparative methods, which are Aquila Optimizer (AO) [34], Salp Swarm Algorithm (SSA) [60], Whale Optimization Algorithm (WOA) [61], Sine Cosine Algorithm (SCA) [62], Arithmetic Optimization Algorithm (AOA) [33], Grey Wolf Optimizer (GWO) [63], Particle Swarm Optimizer (PSO) [64], Prairie Dog Optimization Algorithm (PDOA) [50], and Dwarf Mongoose Optimization Algorithm (DMOA) [49].

To handle these test functions, the IPDOA is tested for 500 iterations and 30 candidate solutions. The suggested IPDOA algorithm was performed 20 times independently to test the IPDOA 's dependability. The results were reported in terms of the best function value, the average of the results, the worst function value, the standard deviation (STD), the p-value, and the Friedman test and Wilcoxon signed-rank test. The following sections compare the proposed algorithms to 10 well-known state-of-the-art algorithms. All algorithms employ

Algorithm 2 Pseudo-code of the proposed IPDOA.

Input: Set the parameters and solutions for the algorithm.

Initialize the parameters and solutions.

while (iter $< Max_{iter}$) **do**

Calculate the Fitness Function (FF) values.

for (i=1 to $Solutions$) **do**
if (rand < 0.5) **then**

Update the solutions using PDOA

Apply the main operators in PDOA

else
Update the solutions using MDOA

Apply the main operators in MDOA

end if

end for

iter=iter+1

end while

Output: Return the best solution.

the same population size and iteration counts of 30 and 500, respectively, to allow for a fair comparison. The code is run on a Personal computer with 16 GB of RAM and an Intel(R)

Table 1 Parameters values of the tested algorithms

No.	Algorithm	Parameter	Value
1	AO	α	0.1
		δ	0.1
2	SSA	v_0	0
3	WOA	α	Decreased from 2 to 0
		b	2
4	SCA	α	0.05
5	AOA	α	5
		μ	0.5
6	GWO	Convergence parameter (a)	Linear reduction from 2 to 0
7	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
8	PDOA	????	????
8	DMOA	CF	$(1 - \frac{iter}{Max_{iter}})^{(2*\frac{iter}{Max_{iter}})}$

Table 2 Details of the tested benchmark functions

Function	Description	Dimensions	Range	f_{min}
F1	$f(x) = \sum_{i=1}^{n} x_i^2$	30	[-100,100]	0
F2	$f(x) = \sum_{i=0}^{n} \lvert x_i \rvert + \prod_{i=0}^{n} \lvert x_i \rvert$	30	[-10,10]	0
F3	$f(x) = \sum_{i=1}^{d} (\sum_{j=1}^{i} x_j)^2$	30	[-100,100]	0
F4	$f(x) = max_i\{\lvert x_i \rvert, 1 \le i \le n\}$	30	[-100,100]	0
F5	$f(x) = \sum_{i=1}^{n-1} [100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2]$	30	[-30,30]	0
F6	$f(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	30	[-100,100]	0
F7	$f(x) = \sum_{i=0}^{n} i x_i^4 + \text{random}[0, 1)$	30	[-128,128]	0
F8	$f(x) = \sum_{i=1}^{n} (-x_i sin(\sqrt{\lvert x_i \rvert}))$	30	[-500,500]	-418.9829× n
F9	$f(x) = \sum_{i=1}^{n} [x_i^2 - 10cos(2\pi x_i) + 10]$	30	[-5.12,5.12]	0
F10	$f(x) = -20exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - exp(\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)) + 20 + e$	10, 50, 100, 500	[-32,32]	0
F11	$f(x) = 1 + \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} cos(\frac{x_i}{\sqrt{i}})$	10, 50, 100, 500	[-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 $y_i = 1 + \frac{x_i + 1}{4}, u(x_i, a, k, m) \begin{cases} K(x_i - a)^m & \text{if } x_i > a \\ 0 & -a \le x_i \ge a \\ K(-x_i - a)^m & -a \le x_i \end{cases}$	10, 50, 100, 500	[-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, 50, 100, 500	[-50,50]	0
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	$f(x) = \sum_{i=1}^{11} \left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2$	4	[-5,5]	0.00030

Table 2 continued

Function	Description	Dimensions	Range	f_{min}
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	$\mathrm{f(x)}=\left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10(1 - \frac{1}{8\pi})cos\ x_1 + 10$	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

Core(TM) i7 CPU using the MATLAB R2015a environment. The main parameters of the used methods are given in Table 1 with their values.

The details of the tested benchmark function problems are given in Table 2. They include various problem classes: multimodal, fixed-dimension multimodal, and unimodal functions.

4.1 Results and discussion

The experimental results are discussed in this subsection. First of all, the behavior and performance of the proposed IPDOA. A comparison between the PDOA, the DMOA, and the IPDOA is made based on the distribution of the candidate solutions. After the statistical com-

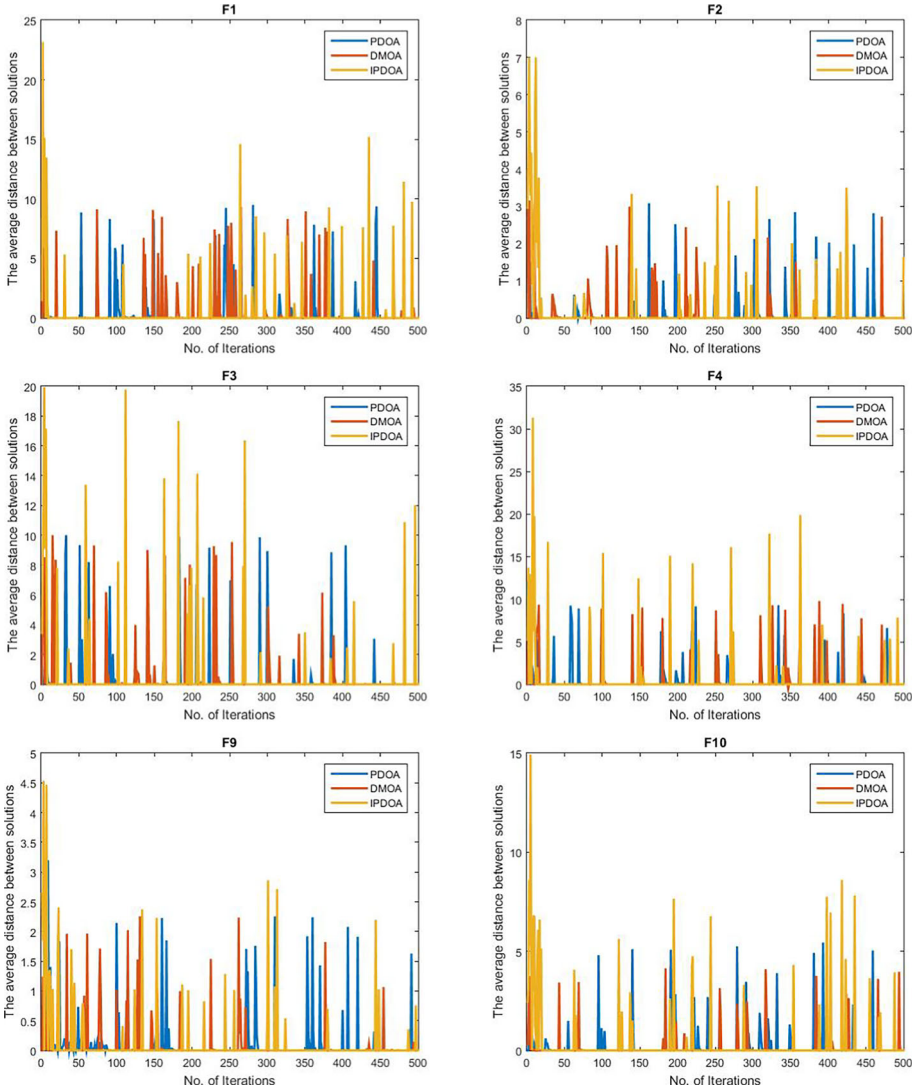


Fig. 1 The distributions of the candidate solutions through the optimization process

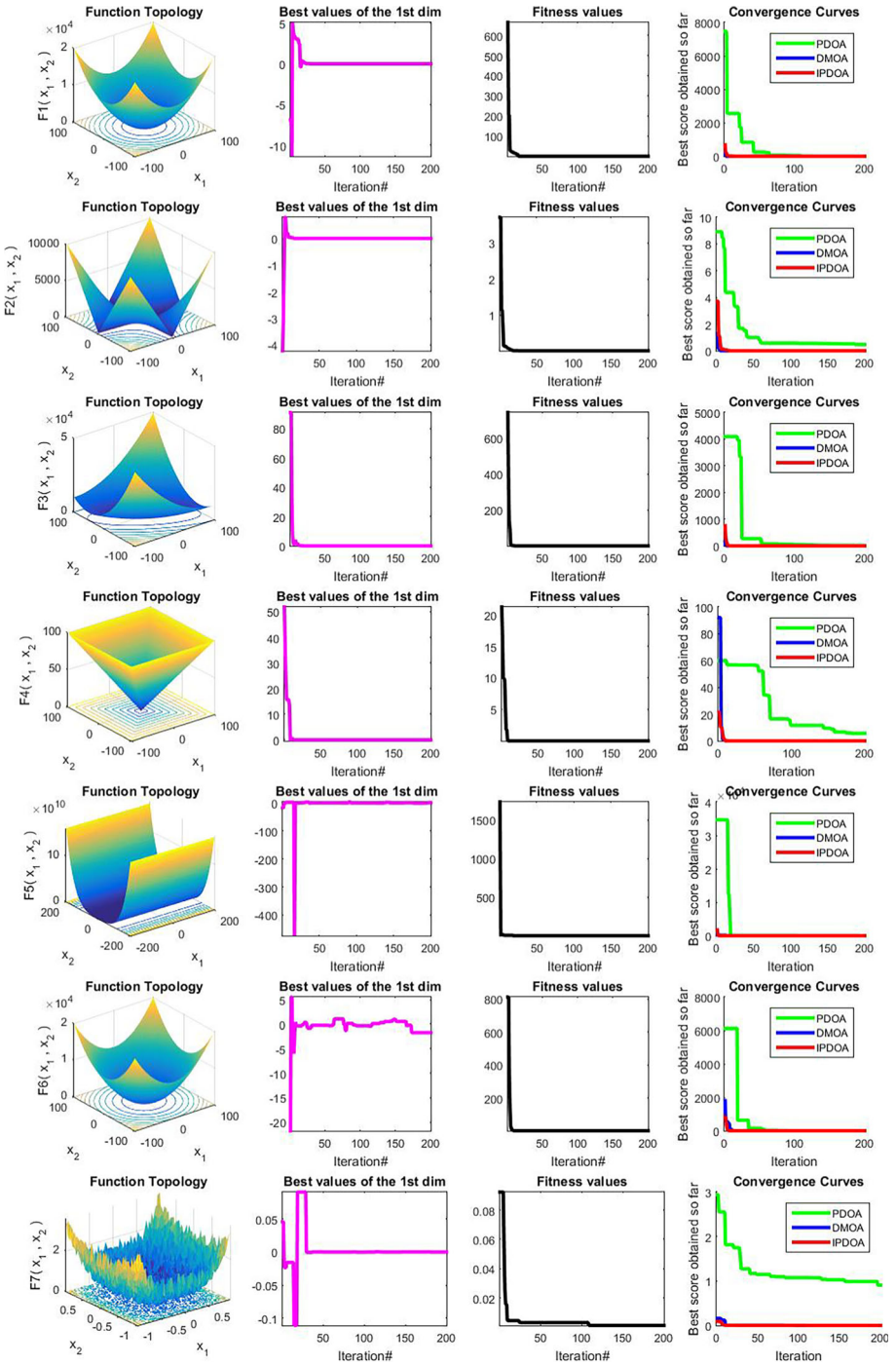


Fig. 2 The obtained qualitative results for the 13 tested problems (F1-F13)

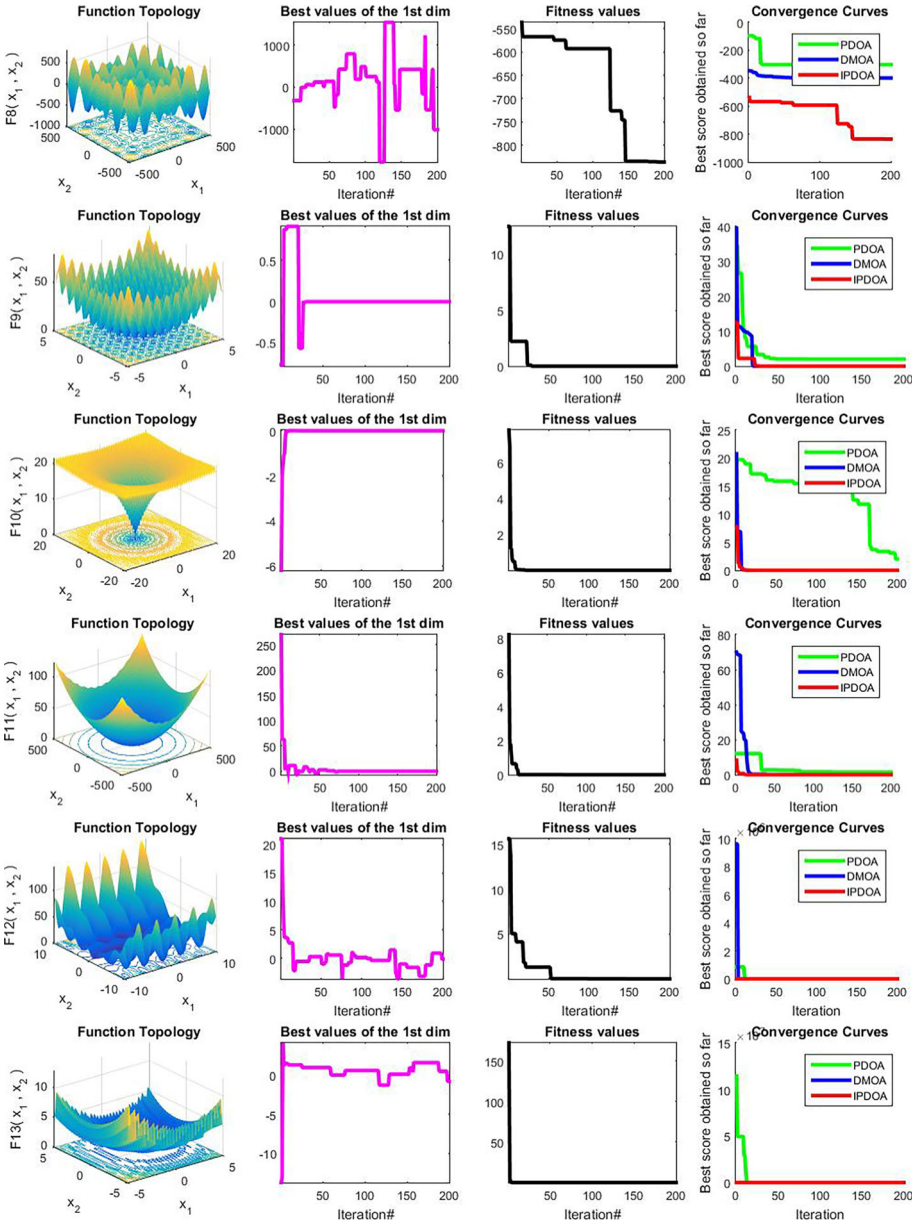


Fig. 2 continued

parison of the results is performed considering the other nine swarm-based algorithms. Here are two experiments performed, one for 10 dimensions and a second for 100 dimensions.

Table 3 The results of the tested algorithms for functions F1-F13 in 10 dimensions

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F1	Best	1.73664E-87	3.60963E-04	2.08421E-30	1.07884E-03	1.19252E+04	2.32547E-18	8.95127E-04	2.10377E-04	6.01631E-93	6.14555E-103
	Average	4.35491E-88	9.04699E-05	5.26459E-31	2.71197E-04	3.57341E+03	7.36732E-19	2.56662E-04	1.16310E-04	1.50408E-93	1.53639E-103
	Worst	1.00979E-92	9.52265E-08	4.33159E-33	3.99341E-10	3.49658E+02	2.80246E-20	5.52567E-06	1.57750E-05	2.55010E-152	1.22880E-115
	STD	8.67435E-88	1.80329E-04	1.03851E-30	5.38434E-04	5.58150E+03	1.07568E-18	4.28296E-04	8.61320E-05	3.00816E-93	3.07277E-103
	p-value	3.54096E-01	3.54408E-01	3.49767E-01	3.52648E-01	2.47657E-01	2.19797E-01	2.75899E-01	3.55415E-02	3.55918E-01	NaN
	h	0	0	0	0	0	0	0	1	0	NaN
	Rank	3	6	4	9	10	5	8	7	2	1
F2	Best	1.14958E-41	3.29050E+01	3.03672E-24	1.23968E-04	1.17187E+01	4.56660E-11	1.47263E-01	3.06198E+01	0.00000E+00	5.64149E-49
	Average	2.87426E-42	1.13577E+01	7.65518E-25	3.20404E-05	8.26764E+00	1.90097E-11	5.48415E-02	1.69409E+01	0.00000E+00	1.49838E-49
	Worst	6.50926E-47	1.19176E-01	3.72810E-31	3.95007E-08	5.52373E+00	6.39520E-12	8.53882E-03	5.34773E+00	0.00000E+00	1.12814E-53
	STD	5.74770E-42	1.47889E+01	1.51418E-24	6.13053E-05	2.99939E+00	1.84446E-11	6.26060E-02	1.25188E+01	0.00000E+00	2.76645E-49
	p-value	3.55855E-01	1.75451E-01	3.50977E-01	3.36168E-01	1.49680E-03	8.49061E-02	1.30337E-01	3.52715E-02	3.20296E-01	NaN
	h	0	0	0	0	1	0	0	1	0	NaN
	Rank	3	9	4	6	8	5	7	10	1	2
F3	Best	4.27981E-85	5.93875E+02	2.19076E+03	5.58769E+00	3.66827E+03	1.10984E-07	3.84547E+00	6.79320E+03	6.12890E-99	1.49696E-95
	Average	1.15739E-85	3.01704E+02	1.12870E+03	1.89824E+00	2.16586E+03	3.56774E-08	1.75394E+00	3.00691E+03	1.53223E-99	3.79467E-96
	Worst	2.26615E-98	5.87297E+01	3.38004E+01	3.22344E-03	1.62025E+03	1.73261E-09	1.29268E-01	8.57511E+02	1.78009E-109	2.31604E-126
	STD	2.08324E-85	2.47430E+02	9.11355E+02	2.63327E+00	1.00219E+03	5.11298E-08	1.68321E+00	2.62752E+03	3.06445E-99	7.45061E-96
	p-value	3.09037E-01	5.05609E-02	4.80017E-02	1.99464E-01	4.97052E-03	2.12307E-01	8.22743E-02	6.20473E-02	3.47865E-01	NaN
	h	0	0	1	0	1	0	0	0	0	NaN
	Rank	3	7	8	6	9	4	5	10	1	2

Table 3 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F4	Best	1.93637E-44	7.04721E+00	4.80758E+01	1.26936E+00	2.20613E+01	2.80902E-05	7.63857E-01	2.66999E+01	6.15916E-57	4.50933E-50
	Average	9.86336E-45	4.45221E+00	2.19534E+01	3.43418E-01	1.13558E+01	8.21802E-06	4.53871E-01	1.64776E+01	1.53979E-57	1.12735E-50
	Worst	1.21666E-45	9.19191E-01	4.45683E+00	1.64933E-03	5.50450E+00	2.00328E-07	1.41636E-01	3.08261E+00	0.00000E+00	1.75473E-60
	STD	8.45287E-45	3.06455E+00	1.99896E+01	6.17831E-01	7.45654E+00	1.33016E-05	3.00986E-01	1.00690E+01	3.07958E-57	2.25465E-50
	p-value	5.83424E-02	2.71349E-02	7.04419E-02	3.08822E-01	2.26319E-02	2.62772E-01	2.35213E-02	1.69716E-02	3.55909E-01	NaN
	h	0	1	0	0	1	0	1	1	0	NaN
	Rank	3	7	10	5	8	4	6	9	1	2
F5	Best	4.23669E-02	5.58716E+02	8.93183E+00	6.78183E+01	1.58132E+05	8.07246E+00	1.04776E+03	1.87904E+04	8.56765E+00	2.17808E+00
	Average	1.50316E-02	2.75029E+02	8.59505E+00	2.33421E+01	4.51811E+04	7.75286E+00	3.14295E+02	5.91770E+03	8.33615E+00	7.96336E-01
	Worst	1.11823E-04	1.38158E+01	8.19771E+00	7.92287E+00	4.37007E+02	7.09299E+00	1.97399E+01	9.14896E+00	8.17891E+00	8.09104E-02
	STD	1.99635E-02	2.33255E+02	3.67214E-01	2.96538E+01	7.55251E+04	4.61754E-01	4.90475E+02	8.66188E+03	1.80851E-01	9.88034E-01
	p-value	1.64914E-01	5.69548E-02	5.98918E-06	1.79383E-01	2.76656E-01	1.42394E-05	2.48350E-01	2.20865E-01	5.50259E-06	NaN
	h	0	0	1	0	0	1	0	0	1	NaN
	Rank	1	7	5	6	10	3	8	9	4	2
F6	Best	1.31188E-03	1.97898E-03	9.41488E-01	1.26973E+00	3.07335E+03	9.00606E-01	8.03322E-04	1.30436E-04	3.30658E-01	9.27636E-01
	Average	8.76273E-04	6.60890E-04	8.49283E-01	1.09379E+00	1.04941E+03	6.68329E-01	2.18686E-04	8.29008E-05	1.75430E-01	2.49548E-01
	Worst	7.00269E-05	1.24159E-05	6.31603E-01	9.05320E-01	1.58825E+02	5.04429E-01	1.13184E-05	2.22680E-05	3.25580E-02	8.50748E-03
	STD	5.80574E-04	9.17907E-04	1.46072E-01	1.67854E-01	1.37152E+03	1.96062E-01	3.89863E-04	5.05918E-05	1.25180E-01	4.52183E-01
	p-value	3.13547E-01	3.13164E-01	4.50320E-02	1.28160E-02	1.76910E-01	1.40157E-01	3.12378E-01	3.12137E-01	7.62748E-01	NaN
	h	0	0	1	1	0	0	0	0	0	NaN
	Rank	4	3	8	9	10	7	2	1	5	6

Table 3 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F7	Best	3.73345E-03	1.31394E-01	2.67264E-02	2.06854E-02	4.64685E-01	3.64085E-03	7.79097E-02	8.37349E-01	7.95265E-04	1.86354E-03
	Average	2.37116E-03	7.15616E-02	1.14801E-02	8.64632E-03	2.16592E-01	2.58133E-03	5.30633E-02	6.13117E-01	4.71981E-04	7.67950E-04
	Worst	1.18249E-03	3.69901E-02	2.10317E-03	3.73463E-03	4.00012E-02	1.64353E-03	3.01381E-02	1.97911E-01	3.46147E-04	1.65664E-04
	STD	1.15296E-03	4.14429E-02	1.07612E-02	8.09411E-03	1.96984E-01	9.89991E-04	2.40893E-02	2.95190E-01	2.16555E-04	7.71562E-04
	p-value	6.01654E-02	1.42160E-02	9.42647E-02	1.00734E-01	7.09505E-02	2.77114E-02	4.87757E-03	6.01894E-03	4.87994E-01	NaN
	h	0	1	0	0	0	1	1	1	0	NaN
	Rank	3	8	6	5	9	4	7	10	1	2
F8	Best	-1.78435E+03	-2.32235E+03	-1.75240E+03	-1.65972E+03	-1.97746E+03	-1.40616E+03	-1.10898E+03	-1.80589E+03	-1.48241E+03	-2.05516E+03
	Average	-1.97280E+03	-2.69533E+03	-2.31685E+03	-1.86532E+03	-2.36880E+03	-2.27607E+03	-1.29764E+03	-2.06132E+03	-2.08186E+03	-3.33130E+03
	Worst	-2.46139E+03	-3.12377E+03	-2.92257E+03	-2.16106E+03	-3.00606E+03	-2.65763E+03	-1.43162E+03	-2.49198E+03	-2.58678E+03	-4.14437E+03
	STD	3.26470E+02	3.41249E+02	5.09900E+02	2.22470E+02	4.43314E+02	5.85644E+02	1.35607E+02	3.00377E+02	4.54419E+02	1.00752E+03
	p-value	4.25978E-02	2.76909E-01	1.22500E-01	2.95037E-02	1.30901E-01	1.20110E-01	7.11168E-03	5.21521E-02	6.44672E-02	NaN
	h	1	0	0	1	0	0	1	0	0	NaN
	Rank	8	2	4	9	3	5	10	7	6	1
F9	Best	7.81811E-03	3.78088E+01	1.42109E-14	2.89915E+00	5.40105E+01	1.56952E+01	1.40099E+01	4.77579E+01	0.00000E+00	0.00000E+00
	Average	1.95675E-03	2.88550E+01	3.55271E-15	7.51509E-01	4.64143E+01	7.40885E+00	1.18869E+01	3.75596E+01	0.00000E+00	0.00000E+00
	Worst	0.00000E+00	2.18891E+01	0.00000E+00	2.82506E-03	2.65296E+01	9.80549E-13	1.00843E+01	2.88538E+01	0.00000E+00	0.00000E+00
	STD	3.90757E-03	7.35558E+00	7.10543E-15	1.43195E+00	1.32759E+01	7.44291E+00	1.85708E+00	9.62499E+00	0.00000E+00	0.00000E+00
	p-value	3.55240E-01	2.26642E-04	3.55918E-01	3.34317E-01	4.26028E-04	9.36057E-02	1.39537E-05	2.33329E-04		NaN
	h	0	1	0	0	1	0	1	1		NaN
	Rank	4	8	3	5	10	6	7	9	1	1

Table 3 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F10	Best	8.88178E-16	1.99345E+01	7.99361E-15	1.99670E+01	1.64445E+01	7.74318E-10	1.69041E-01	1.63442E+01	8.88178E-16	8.88178E-16
	Average	8.88178E-16	6.97002E+00	6.21725E-15	4.99859E+00	1.06706E+01	4.64659E-10	5.44874E-02	9.95625E+00	8.88178E-16	8.88178E-16
	Worst	8.88178E-16	2.31685E+00	8.88178E-16	1.46353E-04	5.40582E+00	1.13642E-10	2.84373E-03	8.88178E-16	8.88178E-16	8.88178E-16
	STD	0.00000E+00	8.64619E+00	3.55271E-15	9.97895E+00	5.77380E+00	3.19175E-10	7.81741E-02	7.66582E+00	0.00000E+00	0.00000E+00
	p-value	NaN	1.58028E-01	2.40082E-02	3.55104E-01	1.01338E-02	2.69234E-02	2.12755E-01	4.07952E-02		NaN
	h	NaN	0	1	0	1	1	0	1		NaN
	Rank	1	8	4	7	10	5	6	9	1	1
F11	Best	0.00000E+00	3.95867E-01	6.62402E-01	6.00996E-01	4.51837E+00	5.12781E-02	2.55626E+00	2.32472E-01	4.94958E-08	0.00000E+00
	Average	0.00000E+00	2.43188E-01	3.30076E-01	3.19548E-01	3.90673E+00	2.41550E-02	1.30674E+00	1.52736E-01	1.23859E-08	0.00000E+00
	Worst	0.00000E+00	1.61086E-01	0.00000E+00	6.80540E-02	3.13172E+00	2.22045E-16	5.04710E-01	7.90203E-02	0.00000E+00	0.00000E+00
	STD	0.00000E+00	1.04373E-01	3.81143E-01	2.18404E-01	7.02767E-01	2.79969E-02	8.99514E-01	6.87464E-02	2.47400E-08	0.00000E+00
	p-value	NaN	3.46752E-03	1.33978E-01	2.64170E-02	3.15533E-05	1.35184E-01	2.71415E-02	4.35959E-03	3.55345E-01	NaN
	h	NaN	1	0	1	1	0	1	1	0	NaN
	Rank	1	6	8	7	10	4	9	5	3	1
F12	Best	2.66031E-04	2.66938E+01	5.96606E-01	4.59887E-01	3.37591E+01	5.70852E-01	2.86746E-04	1.91122E+01	3.78814E-01	1.58719E-02
	Average	1.62632E-04	1.18485E+01	2.34248E-01	2.66668E-01	1.37864E+01	2.49843E-01	7.25222E-05	1.48006E+01	3.07440E-01	4.27198E-03
	Worst	1.68656E-05	2.29270E+00	5.52909E-02	5.62473E-02	6.38742E+00	4.24069E-02	2.26205E-07	8.61561E+00	1.96544E-01	2.08082E-05
	STD	1.15262E-04	1.10860E+01	2.54224E-01	1.66491E-01	1.33293E+01	2.42185E-01	1.42820E-04	4.68502E+00	7.95673E-02	7.74808E-03
	p-value	3.29677E-01	7.64872E-02	1.20542E-01	1.98482E-02	8.41255E-02	8.90428E-02	3.20065E-01	7.35314E-04	2.73201E-04	NaN
	h	0	0	0	1	0	0	0	1	1	NaN
	Rank	2	8	4	6	9	5	1	10	7	3

Table 3 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F13	Best	5.51749E-03	2.03801E-02	6.59101E-01	5.37703E-01	4.27191E+05	5.33513E-01	2.22186E-04	6.59232E+00	9.95668E-01	3.24823E-03
	Average	1.56847E-03	1.14029E-02	5.58846E-01	4.74519E-01	1.19521E+05	4.84749E-01	9.19156E-05	2.42296E+00	8.71890E-01	1.36645E-03
	Worst	3.66733E-05	2.08182E-03	5.16464E-01	3.77815E-01	2.10906E+00	4.11215E-01	9.38533E-06	4.69406E-04	7.67585E-01	1.60008E-05
	STD	2.63910E-03	9.11645E-03	6.71410E-02	7.16776E-02	2.05565E+05	5.20488E-02	9.18563E-05	3.13471E+00	9.62153E-02	1.35595E-03
	p-value	8.96138E-01	7.22733E-02	3.04549E-06	1.16754E-05	2.89044E-01	1.57418E-06	1.09822E-01	1.73296E-01	1.83423E-06	NaN
	h	0	0	1	1	0	1	0	0	1	NaN
	Rank	3	4	7	5	10	6	1	9	8	2
	Mean ranking	3.0000	6.3846	5.7692	6.5385	8.9231	4.8462	5.9231	8.0769	3.1538	2.0000
	Final ranking	2	7	5	8	10	4	6	9	3	1

Table 4 The results of the tested algorithms for functions F14-F23 in 10 dimensions

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F14	Best	1.26705E+01	1.26705E+01	1.08302E+01	3.97178E+00	1.55038E+01	1.26705E+01	1.36186E+01	1.17187E+01	1.26705E+01	2.98211E+00
	Average	6.37968E+00	9.75249E+00	4.45514E+00	3.22952E+00	7.06686E+00	1.12363E+01	6.85642E+00	6.61649E+00	1.26705E+01	2.23955E+00
	Worst	1.99203E+00	9.98462E-01	1.02553E+00	2.98211E+00	9.98004E-01	8.84084E+00	9.98004E-01	1.99203E+00	1.26705E+01	1.99203E+00
	STD	4.91944E+00	5.83602E+00	4.34901E+00	4.94838E-01	7.26758E+00	1.83266E+00	5.54760E+00	5.35409E+00	4.25869E-11	4.95037E-01
	p-value	4.10833E-01	4.25940E-02	1.95732E-01	6.75047E-02	5.85386E-01	6.44790E-01	4.98978E-01	4.58559E-01	3.55918E-01	NaN
	h	0	1	0	0	0	0	0	0	0	NaN
	Rank	4	8	3	2	7	9	6	5	10	1
F15	Best	9.87625E-04	2.09846E-02	1.20020E-03	1.55573E-03	1.67473E+00	5.01183E-04	3.72628E-03	8.42533E-03	9.20622E-02	1.52288E-03
	Average	8.12279E-04	1.14273E-02	8.78563E-04	1.25642E-03	4.25373E-01	4.41251E-04	1.80336E-03	4.26912E-03	3.84818E-02	1.03453E-03
	Worst	4.57691E-04	1.23389E-03	6.03971E-04	9.76905E-04	1.11539E-03	3.56827E-04	7.71831E-04	2.29785E-03	1.52740E-02	8.45345E-04
	STD	2.43457E-04	1.07154E-02	2.87927E-04	2.41066E-04	8.32953E-01	6.59302E-05	1.31804E-03	2.81931E-03	3.59678E-02	3.26161E-04
	p-value	3.16683E-01	1.00597E-01	5.00342E-01	3.15842E-01	3.47573E-01	1.18438E-02	3.00654E-01	6.28515E-02	8.24858E-02	NaN
	h	0	0	0	0	0	1	0	0	0	NaN
	Rank	2	8	3	5	10	1	6	7	9	4
F16	Best	-1.02296E+00	-1.03163E+00	-1.03163E+00	-1.03004E+00	-1.03122E+00	-1.03163E+00	-1.00199E+00	-2.15464E-01	-1.03163E+00	-1.03163E+00
	Average	-1.02651E+00	-1.03163E+00	-1.03163E+00	-1.03069E+00	-1.03142E+00	-1.03163E+00	-1.01389E+00	-8.27587E-01	-1.03163E+00	-1.03163E+00
	Worst	-1.03037E+00	-1.03163E+00	-1.03163E+00	-1.03138E+00	-1.03161E+00	-1.03163E+00	-1.03132E+00	-1.03163E+00	-1.03163E+00	-1.03163E+00
	STD	3.24108E-03	9.88867E-15	1.03551E-06	5.49607E-04	1.59890E-04	3.89570E-07	1.26126E-02	4.08082E-01	6.37105E-07	1.28198E-16
	p-value	1.00740E-01	3.06315E-02	3.06372E-02	3.73982E-02	3.20170E-02	3.06333E-02	3.06315E-02	3.96630E-01	3.06353E-02	NaN
	h	0	1	1	1	1	1	1	0	1	NaN
	Rank	8	2	5	7	6	3	9	10	4	1

Table 4 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F17	Best	4.01604E-01	3.97887E-01	4.27427E-01	4.39352E-01	2.70539E+00	3.97944E-01	3.97887E-01	3.97887E-01	9.62169E-01	4.12154E-01
	Average	3.99717E-01	3.97887E-01	4.13066E-01	4.10357E-01	1.55166E+00	3.97904E-01	3.97887E-01	3.97887E-01	6.46241E-01	4.03311E-01
	Worst	3.98412E-01	3.97887E-01	3.99962E-01	3.99797E-01	3.97893E-01	3.97889E-01	3.97887E-01	3.97887E-01	4.03766E-01	3.98111E-01
	STD	1.43550E-03	2.96000E-14	1.46348E-02	1.93415E-02	1.33221E+00	2.65936E-05	0.00000E+00	2.82987E-13	2.56177E-01	6.62113E-03
	p-value	3.29513E-01	1.52493E-01	2.70148E-01	5.16344E-01	1.35483E-01	1.53553E-01	1.52493E-01	1.52493E-01	1.06768E-01	NaN
	h	0	0	0	0	0	0	0	0	0	NaN
	Rank	5	2	8	7	10	4	1	3	9	6
F18	Best	3.44003E+00	3.00000E+00	4.30026E+00	3.00274E+00	8.40000E+01	8.40003E+01	3.00000E+00	3.73808E+00	3.75156E+02	3.00000E+00
	Average	3.13982E+00	3.00000E+00	3.32734E+00	3.00081E+00	2.32501E+01	4.35008E+01	3.00000E+00	3.26163E+00	2.33870E+02	3.00000E+00
	Worst	3.00779E+00	3.00000E+00	3.00087E+00	3.00007E+00	3.00007E+00	3.00136E+00	3.00000E+00	3.00256E+00	3.00000E+00	3.00000E+00
	STD	2.01488E-01	2.63488E-13	6.48617E-01	1.28754E-03	4.05000E+01	4.67647E+01	2.08296E-15	3.26414E-01	1.74221E+02	1.08545E-11
	p-value	5.48787E-01	1.60037E-01	8.62341E-01	1.61131E-01	3.61733E-01	1.36059E-01	1.60037E-01	1.60037E-01	3.81658E-02	NaN
	h	0	0	0	0	0	0	0	0	1	NaN
	Rank	5	2	7	4	8	9	1	6	10	3
F19	Best	-3.74942E+00	-3.86175E+00	-3.07646E+00	-3.82597E+00	-3.66646E+00	-3.85556E+00	-3.62908E+00	-3.62908E+00	-3.80778E+00	-3.86048E+00
	Average	-3.79907E+00	-3.86232E+00	-3.64182E+00	-3.83814E+00	-3.76489E+00	-3.86080E+00	-3.72296E+00	-3.72296E+00	-3.82901E+00	-3.86175E+00
	Worst	-3.83103E+00	-3.86277E+00	-3.86250E+00	-3.84797E+00	-3.85718E+00	-3.86278E+00	-3.83418E+00	-3.83418E+00	-3.85117E+00	-3.86270E+00
	STD	3.92717E-02	4.32141E-04	3.78040E-01	1.05562E-02	8.45799E-02	3.50120E-03	9.59308E-02	9.59308E-02	2.38909E-02	1.07462E-03
	p-value	1.92348E-01	2.71442E-02	6.91820E-01	5.42632E-02	5.36384E-01	2.83604E-02	2.68055E-02	2.75718E-02	7.55958E-02	NaN
	h	0	1	0	0	0	1	1	1	0	NaN
	Rank	6	1	10	4	7	3	8	8	5	2

Table 4 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F20	Best	-2.56150E+00	-3.13035E+00	-2.02347E+00	-1.13838E+00	-2.84063E+00	-3.32145E+00	-3.20310E+00	-1.71698E+00	-2.81572E+00	-3.19807E+00
	Average	-2.87326E+00	-3.19751E+00	-2.89761E+00	-2.15896E+00	-3.09884E+00	-3.32168E+00	-3.26255E+00	-2.17743E+00	-2.92781E+00	-3.29101E+00
	Worst	-3.02503E+00	-3.32200E+00	-3.26703E+00	-2.99472E+00	-3.28920E+00	-3.32184E+00	-3.32200E+00	-2.80086E+00	-3.06235E+00	-3.32200E+00
	STD	2.10598E-01	8.67997E-02	5.85550E-01	9.07767E-01	1.87733E-01	1.70894E-04	6.86433E-02	5.40090E-01	1.01520E-01	6.19607E-02
	p-value	5.32480E-02	9.74173E-03	1.20588E-01	9.73231E-01	1.80714E-02	5.45639E-03	7.23186E-03	6.38021E-03	3.41469E-02	NaN
	h	0	1	0	0	1	1	1	1	1	NaN
	Rank	8	4	7	10	5	1	3	9	6	2
F21	Best	-9.05018E+00	-2.63047E+00	-8.81459E-01	-2.73115E-01	-4.14562E+00	-2.63008E+00	-2.63047E+00	-2.68286E+00	-1.47774E+00	-4.88073E+00
	Average	-9.69889E+00	-6.40493E+00	-5.05355E+00	-7.58014E-01	-4.57451E+00	-3.25861E+00	-5.14183E+00	-7.02251E+00	-2.61209E+00	-7.28458E+00
	Worst	-9.98480E+00	-1.01532E+01	-9.42069E+00	-1.53454E+00	-5.00745E+00	-5.09282E+00	-1.01532E+01	-1.01532E+01	-3.78482E+00	-1.01375E+01
	STD	4.40094E-01	4.32818E+00	3.48801E+00	5.82155E-01	3.72284E-01	1.22305E+00	3.53407E+00	3.74736E+00	9.81017E-01	2.72146E+00
	p-value	1.30415E-01	7.42507E-01	3.52104E-01	3.35992E-03	9.59170E-02	3.56384E-02	3.73770E-01	9.13584E-01	1.79027E-02	NaN
	h	0	0	0	1	0	1	0	0	1	NaN
	Rank	1	4	6	10	7	8	5	3	9	2
F22	Best	-9.60690E+00	-2.75193E+00	-1.71560E+00	-3.74816E-01	-3.65973E+00	-4.83329E+00	-2.76590E+00	-3.72430E+00	-2.26041E+00	-1.03964E+01
	Average	-1.00074E+01	-8.49019E+00	-3.91841E+00	-7.67057E-01	-6.15286E+00	-7.53852E+00	-6.82392E+00	-7.06362E+00	-3.40594E+00	-1.03998E+01
	Worst	-1.02793E+01	-1.04029E+01	-5.03039E+00	-1.26563E+00	-9.69960E+00	-1.03323E+01	-1.04029E+01	-1.04029E+01	-4.68264E+00	-1.04021E+01
	STD	3.29629E-01	3.82550E+00	1.54260E+00	4.01099E-01	2.57194E+00	2.98656E+00	4.15095E+00	3.85591E+00	1.11493E+00	2.50745E-03
	p-value	1.51420E-01	7.08480E-01	7.47012E-02	4.12900E-03	5.08319E-01	1.03823E-01	7.89256E-01	8.52021E-01	4.10634E-02	NaN
	h	0	0	0	1	0	0	0	0	1	NaN
	Rank	2	3	8	10	7	4	6	5	9	1

Table 4 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F23	Best	-4.16182E+00	-1.67655E+00	-9.44938E-01	-1.52013E+00	-2.72353E+00	-2.42161E+00	-3.83543E+00	-2.42173E+00	-1.39198E+00	-5.04220E+00
	Average	-8.65283E+00	-2.33446E+00	-4.07272E+00	-2.26275E+00	-5.36120E+00	-7.15008E+00	-8.86116E+00	-5.48051E+00	-2.68359E+00	-8.90865E+00
	Worst	-1.04318E+01	-2.80663E+00	-9.97444E+00	-4.25338E+00	-1.03488E+01	-1.05271E+01	-1.05364E+01	-1.05364E+01	-3.70382E+00	-1.04493E+01
	STD	3.01549E+00	4.73652E-01	4.26279E+00	1.33164E+00	3.48919E+00	4.05089E+00	3.35049E+00	3.54723E+00	1.18866E+00	2.58406E+00
	p-value	9.01697E-01	6.07961E-03	1.29913E-01	8.19936E-03	2.03340E-01	5.73493E-01	9.29363E-01	2.21888E-01	1.02911E-02	NaN
	h	0	1	0	1	0	0	0	0	1	NaN
	Rank	3	9	7	10	6	4	2	5	8	1
	Mean ranking	4.4000	4.3000	6.4000	6.9000	7.3000	4.6000	4.7000	6.1000	7.9000	2.3000
	Final ranking	3	2	7	8	9	4	5	6	10	1

4.1.1 Behavior and performance analysis

Figure 1 presents an analysis of the distribution of the candidate solutions along the iterative process. They have selected six representative benchmark functions (F1-F4, F9, and F10); from the plots, it is possible to see that the IPDOA has a better distribution of the population elements along the iterations. From Fig. 1, it is also possible to see that the PDOA is the second algorithm with a good distribution of the solutions, and in the third place is the DMOA.

Following the study of the behavior of the proposed IPDOA in Fig. 2, they used the benchmark function F1-F13 to show the topology of the function, the evolution of the best solution, and its fitness obtained by the IPDOA. Besides, in the last column of Fig. 2, the plots of the convergence curves of the PDO, DMOA, and IPDOA obtained from the selected functions are shown. From such plots, it is possible to see that the IPDOA has fast convergence and accuracy. A good example could be seen in F8, a multi-modal function; the IPDOA converges faster than the PDOA and DMOA.

4.1.2 Statistical analysis for 10 dimensions

On the other hand, the proposed IPDOA is also compared with other swarm-inspired algorithms from state of the art. In this case, the comparison is conducted considering the Best and Worst fitness values, the average fitness values of the 20 independent runs, and the Standard Deviation (STD). A Rank is also performed based on the optimal value obtained for each algorithm. Tables 3 and 4 show the results of the nine algorithms and the IPDOA over the 23 benchmark functions in 10 dimensions. In Table 3 are listed the values for functions F1-13 and in Table 4 for F14-F23. Based on the ranking, it is possible to see that the IPDOA is in the first rank for the 23 functions for ten dimensions. For the F1-F13 the AO is in the second rank, followed by the DMOA in the third place. Finally, the last rank is the AOA. For F14-F23 the SSA is in the second place and the AO in the third rank; meanwhile, the worst is the DMOA. Finally, the p-values obtained after performing the Wilcoxon signed-rank test confirm that the algorithms are statically different in most cases. This fact could be seen since most experiments' p-value is less than 0.005. However, in some functions, such as F7, the IPDOA has overcome the other comparative algorithms.

The behavior and convergence of the IPDOA are compared with the AO, WOA, AOA, PSO, DMOA, SSA, SCA, GWO, and PDOA. Figure 3 presents the converged curves of the ten algorithms for the 23 benchmark functions in 10 dimensions. From the plots, it is possible to observe the convergence speed and accuracy of the IPDOA. Here is demonstrated that by merging the two approaches, the performance is substantially improved. Different functions show the capabilities of the IPDOA to escape from suboptimal solutions. For example, in F12, F14, and F19, in these plots, it is clear that the algorithm finds a lower value after certain iterations. Some examples of convergence speed are F5 and F16 where the proposed algorithms achieve the global optimal in fewer iterations than the rest of the methods.

4.1.3 Statistical analysis for 100 dimensions

In order to verify the capabilities of the proposed IPDOA over high-dimensional problems, the dimensionality of the 13 benchmark functions is set to 100. In this case, they used the functions F1-F13. Table 5 presents the results of the ten compared algorithms over the 13 selected functions. From this table, it is possible to see that the IPDOA is the first in the rank, followed by AO, and in third place is the GWO. Finally, the worst method for this series of

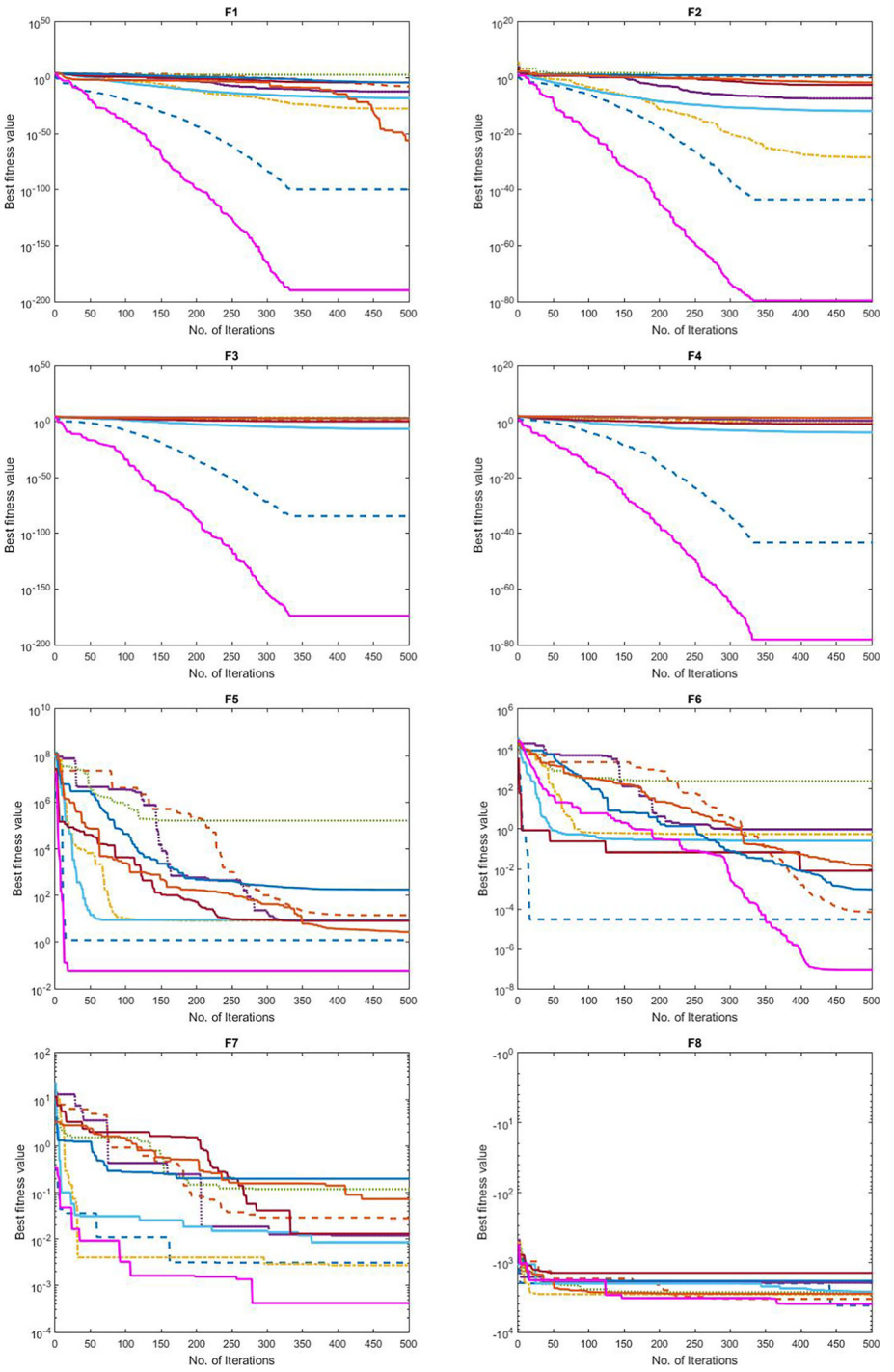


Fig. 3 Convergence behavior of the tested algorithms on the functions (F1-F23) in 10 dimensions

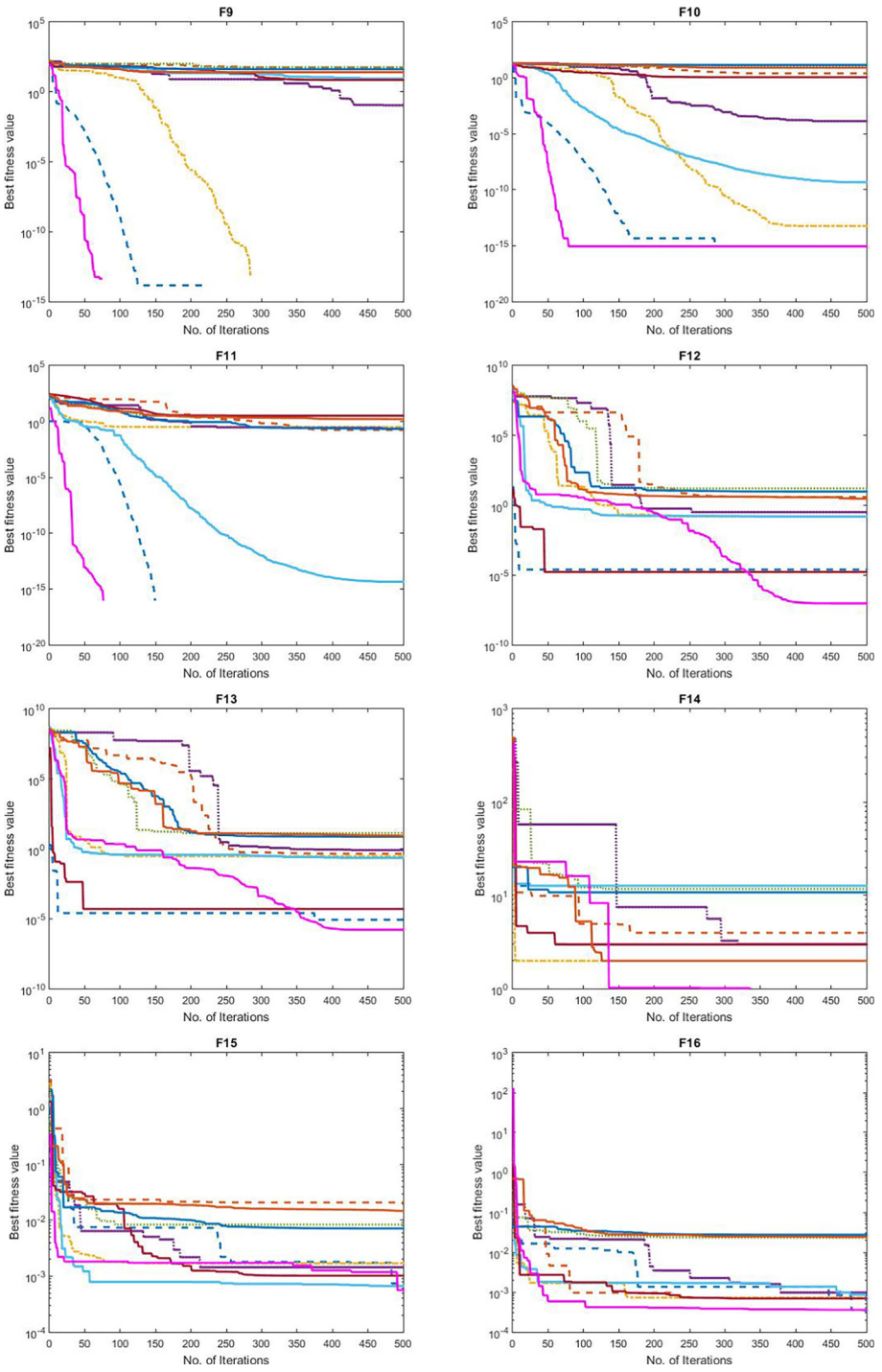


Fig. 3 continued

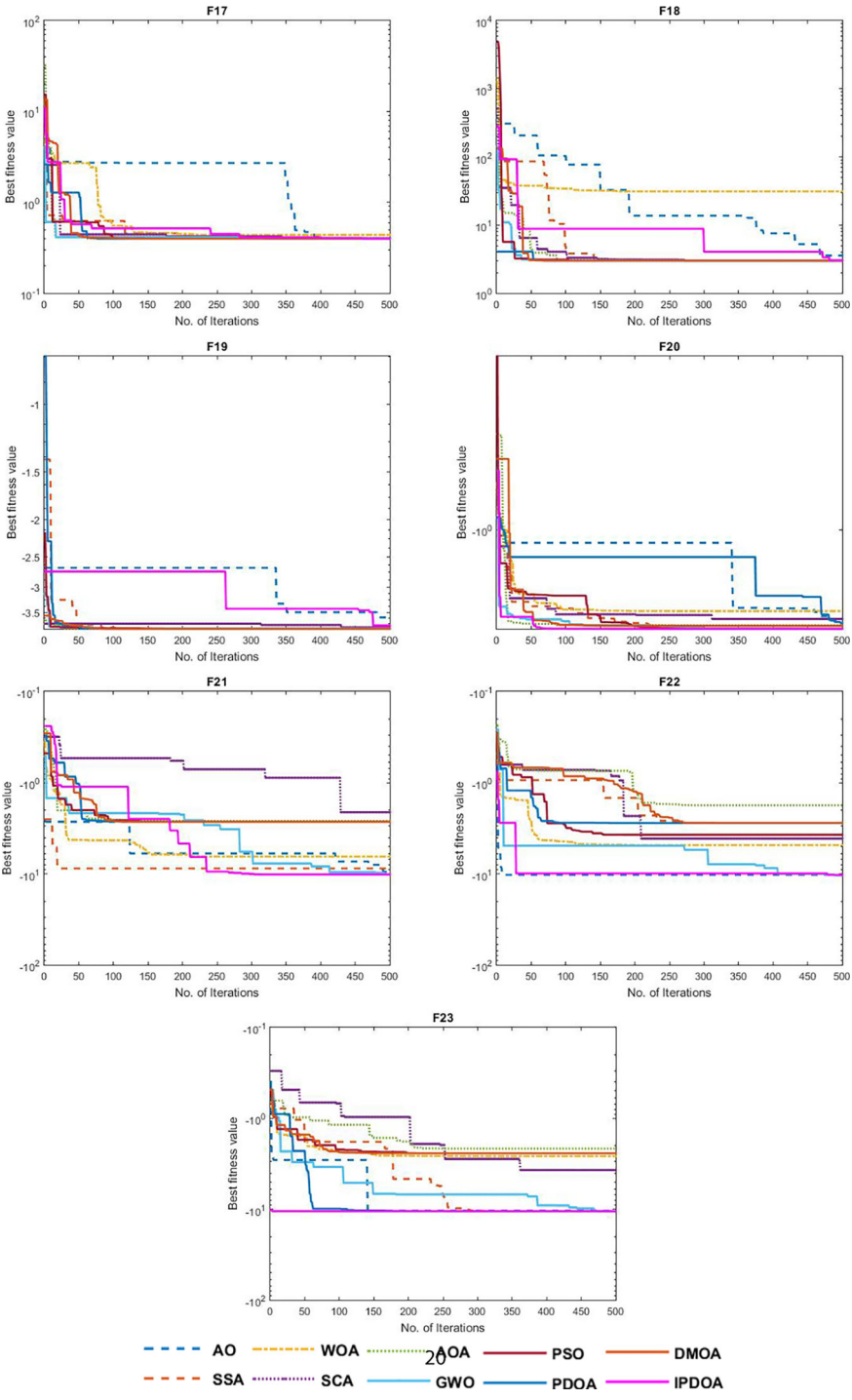


Fig. 3 continued

Table 5 The results of the tested algorithms using 13 problems, where the dimension is 100

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F1	Best	1.39089E-84	2.11869E+04	2.05468E-23	4.26361E+03	2.63898E+04	1.41976E-05	1.24110E+02	3.09071E+04	1.38378E-02	1.77800E-98
	Average	3.47893E-85	1.44767E+04	5.77442E-24	2.23798E+03	1.75202E+04	4.49993E-06	7.95642E+01	2.58346E+04	7.22449E-03	4.44501E-99
	Worst	2.01751E-99	9.58441E+03	5.75705E-44	3.86193E+02	6.50218E+03	4.70234E-07	5.38046E+01	2.30851E+04	1.10132E-03	2.24231E-116
	STD	6.95335E-85	5.59697E+03	9.88227E-24	1.67715E+03	9.64283E+03	6.52346E-06	3.27250E+01	3.67955E+03	5.37146E-03	8.89001E-99
	p-value	3.55629E-01	2.06826E-03	2.86874E-01	3.70864E-02	1.09150E-02	2.16917E-01	2.81541E-03	8.13768E-06	3.60549E-02	NaN
	h	0	1	0	1	1	0	1	1	1	NaN
	Rank	2	8	3	7	9	4	6	10	5	1
F2	Best	8.95812E-43	2.26601E+02	5.87380E-25	9.49815E+00	1.67219E+02	4.23681E-04	6.03622E+01	1.37959E+07	2.88900E-40	5.23832E-51
	Average	2.94611E-43	1.85255E+02	1.48454E-25	3.45719E+00	1.07855E+02	1.68006E-04	4.50001E+01	3.44913E+06	7.68372E-41	1.91051E-51
	Worst	4.27842E-46	1.47847E+02	5.76342E-37	1.41943E-01	6.77002E+01	6.72528E-05	3.11147E+01	1.83964E+02	1.86424E-55	3.61257E-56
	STD	4.06462E-43	3.27946E+01	2.92628E-25	4.21693E+00	4.38369E+01	1.71029E-04	1.22424E+01	6.89784E+06	1.41643E-40	2.49092E-51
	p-value	1.97339E-01	2.87686E-05	3.49439E-01	1.52185E-01	2.65483E-03	9.70673E-02	3.24241E-04	3.55891E-01	3.19603E-01	NaN
	h	0	1	0	0	1	0	1	0	0	NaN
	Rank	2	9	4	6	8	5	7	10	3	1
F3	Best	3.79245E-83	6.22353E+04	5.25539E+05	1.09646E+05	2.74196E+05	1.46715E+03	1.13751E+04	1.82901E+05	3.95721E-01	7.07299E-90
	Average	9.48401E-84	5.64017E+04	4.17481E+05	9.27282E+04	1.19941E+05	9.67376E+02	9.96234E+03	1.37682E+05	2.25341E-01	1.76825E-90
	Worst	3.35638E-88	5.08245E+04	2.30393E+05	6.69064E+04	6.32238E+04	4.10050E+02	8.78695E+03	7.56505E+04	1.36098E-01	3.07642E-111
	STD	1.89603E-83	6.20976E+03	1.30650E+05	1.83351E+04	1.02916E+05	4.41448E+02	1.20229E+03	4.66130E+04	1.18200E-01	3.53650E-90
	p-value	3.55737E-01	1.79173E-06	6.90940E-04	5.42675E-05	5.85715E-02	4.65441E-03	3.07865E-06	1.04636E-03	8.83385E-03	NaN
	h	0	1	1	1	0	1	1	1	1	NaN
	Rank	2	6	10	7	8	4	5	9	3	1

Table 5 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F4	Best	1.02711E-41	7.30649E+01	9.29632E+01	8.59439E+01	4.03240E+01	4.05616E+00	2.75864E+01	5.51302E+01	8.38946E-02	2.35288E-51
	Average	2.81193E-42	6.26424E+01	8.38400E+01	8.47657E+01	3.41063E+01	1.82687E+00	2.45898E+01	5.11033E+01	6.49547E-02	5.88271E-52
	Worst	1.08586E-45	5.48066E+01	7.01775E+01	8.41938E+01	2.01184E+01	7.03654E-01	2.26590E+01	4.94768E+01	5.33089E-02	2.42170E-63
	STD	4.99378E-42	7.62668E+00	1.00452E+01	7.99477E-01	9.52473E+00	1.55198E+00	2.15583E+00	2.69210E+00	1.32578E-02	1.17641E-51
	p-value	3.03111E-01	3.24228E-06	2.95032E-06	7.42138E-13	3.74051E-04	5.67285E-02	4.64748E-07	2.22967E-08	6.50341E-05	NaN
	h	0	1	1	1	1	0	1	1	1	NaN
	Rank	2	8	9	10	6	4	5	7	3	1
F5	Best	9.00841E-02	8.64674E+06	4.88726E+01	6.95507E+07	1.91673E+07	4.87942E+01	6.12350E+04	3.03381E+07	4.89455E+01	4.72456E+01
	Average	3.07900E-02	5.41451E+06	4.88350E+01	3.90063E+07	1.32246E+07	4.85186E+01	4.30366E+04	1.82424E+07	4.88903E+01	3.17874E+01
	Worst	1.29144E-03	2.60050E+06	4.88046E+01	2.56705E+07	4.39147E+06	4.79602E+01	2.71570E+04	5.84156E+06	4.88503E+01	1.50138E+01
	STD	4.01304E-02	3.20634E+06	2.83149E-02	2.06894E+07	6.80694E+06	3.90842E-01	1.57441E+04	1.06693E+07	3.99403E-02	1.31806E+01
	p-value	2.94403E-03	1.49063E-02	4.13911E-02	9.28167E-03	8.11783E-03	4.42210E-02	1.56819E-03	1.41507E-02	4.09271E-02	NaN
	h	1	1	1	1	1	1	1	1	1	NaN
	Rank	1	7	4	10	8	3	6	9	5	2
F6	Best	1.33662E-01	1.47671E+04	9.84310E+00	2.52089E+03	3.77867E+04	7.95059E+00	2.05734E+02	3.40483E+04	9.40136E+00	2.24170E-01
	Average	3.90746E-02	1.27628E+04	8.33089E+00	8.45887E+02	2.06850E+04	6.97348E+00	1.10708E+02	2.76714E+04	8.87001E+00	7.51539E-02
	Worst	9.36442E-04	1.11573E+04	7.59823E+00	6.95905E+01	7.22768E+03	6.05757E+00	6.66209E+01	2.02391E+04	8.47315E+00	4.19196E-05
	STD	6.34605E-02	1.49493E+03	1.04303E+00	1.14233E+03	1.44591E+04	1.05567E+00	6.39579E+01	7.08348E+03	4.28990E-01	1.05548E-01
	p-value	5.79294E-01	2.58203E-06	4.15332E-06	1.89148E-01	2.87581E-02	1.27355E-05	1.34747E-02	2.31961E-04	1.67765E-08	NaN
	h	0	1	1	0	1	1	1	1	1	NaN
	Rank	1	8	4	7	9	3	6	10	5	2

Table 5 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F7	Best	5.03714E-03	2.14839E+01	7.16802E-02	2.18310E+01	2.26876E+01	1.33312E-02	6.26818E+02	4.92488E+01	7.15838E-04	1.05912E-03
	Average	3.24541E-03	1.39007E+01	2.91127E-02	1.33519E+01	1.39949E+01	1.02448E-02	2.29805E+02	2.39856E+01	3.41285E-04	6.87655E-04
	Worst	1.96725E-03	9.39918E+00	1.54743E-03	8.35100E+00	8.57402E+00	6.32427E-03	9.33534E+01	1.46402E+01	1.51529E-04	4.72012E-04
	STD	1.42695E-03	5.25737E+00	3.06365E-02	6.06824E+00	6.10409E+00	2.95515E-03	2.64698E+02	1.68850E+01	2.55292E-04	2.56718E-04
	p-value	1.23934E-02	1.85128E-03	1.12915E-01	4.56662E-03	3.74995E-03	6.61172E-04	1.33180E-01	2.95290E-02	1.04217E-01	NaN
	h	1	1	0	1	1	1	0	1	0	NaN
	Rank	3	7	5	6	8	4	10	9	1	2
F8	Best	-3.97279E+03	-9.10320E+03	-1.30643E+04	-4.13864E+03	-2.92673E+03	-4.18226E+03	-2.71482E+03	-9.02946E+03	-5.43197E+03	-3.17106E+03
	Average	-5.26217E+03	-9.60929E+03	-1.43383E+04	-4.53235E+03	-4.72002E+03	-5.99944E+03	-3.09677E+03	-9.02946E+03	-5.66551E+03	-8.43391E+03
	Worst	-6.22120E+03	-1.02229E+04	-1.49355E+04	-4.85865E+03	-6.15724E+03	-8.42029E+03	-3.61803E+03	-9.02946E+03	-5.91953E+03	-2.06183E+04
	STD	1.08689E+03	4.88414E+02	8.60430E+02	2.97585E+02	1.40172E+03	2.02065E+03	3.77774E+02	0.00000E+00	2.43171E+02	8.19448E+03
	p-value	4.71955E-01	7.84231E-01	2.01789E-01	3.78036E-01	4.06019E-01	5.84990E-01	2.40908E-01	8.89192E-01	5.24596E-01	NaN
	h	0	0	0	0	0	0	0	0	0	NaN
	Rank	7	2	1	9	8	5	10	3	6	4
F9	Best	6.37808E-05	4.49180E+02	1.13687E-13	2.01073E+02	5.67620E+02	2.92636E+01	3.56846E+02	3.71087E+02	0.00000E+00	0.00000E+00
	Average	1.59452E-05	4.03633E+02	2.84217E-14	1.55238E+02	5.30339E+02	1.97734E+01	3.13660E+02	3.17865E+02	0.00000E+00	0.00000E+00
	Worst	0.00000E+00	3.31756E+02	0.00000E+00	8.81792E+01	4.81491E+02	1.31565E+01	2.80786E+02	2.48864E+02	0.00000E+00	0.00000E+00
	STD	3.18904E-05	5.04841E+01	5.68434E-14	5.19709E+01	3.65537E+01	6.80077E+00	3.55020E+01	5.34984E+01	0.00000E+00	0.00000E+00
	p-value	3.55918E-01	3.79906E-06	3.55918E-01	9.86736E-04	1.10992E-07	1.13601E-03	2.10949E-06	2.14907E-05	NaN	NaN
	h	0	1	0	1	1	1	1	1	NaN	NaN
	Rank	4	9	3	6	10	5	7	8	1	1

Table 5 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F10	Best	4.44089E-15	1.86978E+01	1.50990E-14	2.06156E+01	1.68896E+01	1.87669E-04	7.07002E+00	1.86674E+01	1.03962E-06	8.88178E-16
	Average	1.77636E-15	1.75367E+01	5.32907E-15	2.05387E+01	1.54767E+01	1.46616E-04	5.54199E+00	1.24123E+01	2.59905E-07	8.88178E-16
	Worst	8.88178E-16	1.62977E+01	8.88178E-16	2.04683E+01	1.21521E+01	1.14930E-04	4.76133E+00	8.88178E-16	8.88178E-16	8.88178E-16
	STD	1.77636E-15	9.83076E-01	6.72518E-15	6.61159E-02	2.23009E+00	3.59636E-05	1.05162E+00	8.79198E+00	5.19810E-07	0.00000E+00
	p-value	3.55918E-01	3.23296E-08	2.34746E-01	1.17354E-15	8.70972E-06	1.83064E-04	4.28830E-05	3.02125E-02	3.55918E-01	NaN
	h	0	1	0	1	1	1	1	1	0	NaN
	Rank	2	9	3	10	8	5	6	7	4	1
F11	Best	0.00000E+00	1.01822E+02	0.00000E+00	2.76697E+01	1.84767E+02	7.75663E-02	1.64927E+02	2.94634E+02	4.32446E+02	0.00000E+00
	Average	0.00000E+00	8.16854E+01	0.00000E+00	1.44489E+01	1.51133E+02	3.63403E-02	1.16914E+02	2.45880E+02	2.13931E+02	0.00000E+00
	Worst	0.00000E+00	5.02879E+01	0.00000E+00	2.60080E+00	9.40224E+01	7.40680E-07	5.97624E+01	2.17666E+02	1.01148E+02	0.00000E+00
	STD	0.00000E+00	2.21652E+01	0.00000E+00	1.05760E+01	4.05681E+01	4.21487E-02	4.32562E+01	3.53598E+01	1.52502E+02	0.00000E+00
	p-value	NaN	3.19669E-04	NaN	3.40776E-02	3.01246E-04	1.35402E-01	1.65503E-03	8.60998E-06	3.09339E-02	NaN
	h	NaN	1	NaN	1	1	0	1	1	1	NaN
	Rank	1	6	1	5	8	4	7	10	9	1
F12	Best	3.81735E-05	6.64339E+06	8.80905E-01	8.50195E+07	3.59639E+07	6.58235E-01	2.02685E+01	1.06942E+07	9.78629E-01	7.03552E-03
	Average	2.90036E-05	3.79737E+06	6.77254E-01	3.93621E+07	1.91639E+07	4.86145E-01	1.31210E+01	4.12603E+06	9.57246E-01	2.88730E-03
	Worst	2.31287E-05	9.63373E+05	3.60268E-01	1.38492E+07	6.83060E+02	3.31860E-01	7.75838E+00	6.59422E+05	9.33222E-01	2.51993E-05
	STD	7.21687E-06	2.31998E+06	2.32516E-01	3.13143E+07	1.77531E+07	1.50260E-01	5.34651E+00	4.64225E+06	2.30198E-02	3.00594E-03
	p-value	1.05911E-01	1.69571E-02	1.15132E-03	4.56556E-02	7.41856E-02	6.68228E-04	2.69134E-03	1.25794E-01	2.18013E-10	NaN
	h	0	1	1	1	0	1	1	0	1	NaN
	Rank	1	7	4	10	9	3	6	8	5	2

Table 5 continued

Fun	Measure	Comparative of Algorithms									
		AO	SSA	WOA	SCA	AOA	GWO	PSO	PDOA	DMOA	IPDOA
F13	Best	1.18487E-03	5.40725E+07	4.02336E+00	4.99675E+08	3.19538E+07	4.13226E+00	1.95761E+03	9.84588E+07	4.99861E+00	2.06318E-02
	Average	4.03343E-04	2.79745E+07	3.64981E+00	2.16899E+08	9.67936E+06	3.99707E+00	8.91002E+02	5.70674E+07	4.96424E+00	9.02390E-03
	Worst	8.94191E-05	9.66037E+06	2.92923E+00	2.04412E+06	4.72068E+05	3.90752E+00	3.76766E+01	2.16086E+07	4.94018E+00	6.34290E-04
	STD	5.22390E-04	1.88721E+07	5.13129E-01	2.39042E+08	1.49799E+07	1.05168E-01	8.48008E+02	3.18189E+07	2.61980E-02	9.96631E-03
	p-value	1.34807E-01	2.51313E-02	7.66137E-06	1.19487E-01	2.43785E-01	3.63347E-10	8.03246E-02	1.15452E-02	3.45466E-14	NaN
	h	0	1	1	0	0	1	0	1	1	NaN
	Rank	1	8	3	10	7	4	6	9	5	2
	Mean ranking	2.2308	7.2308	4.1538	7.9231	8.1538	4.0769	6.6923	8.3846	4.2308	1.6154
	Final ranking	2	7	4	8	9	3	6	10	5	1

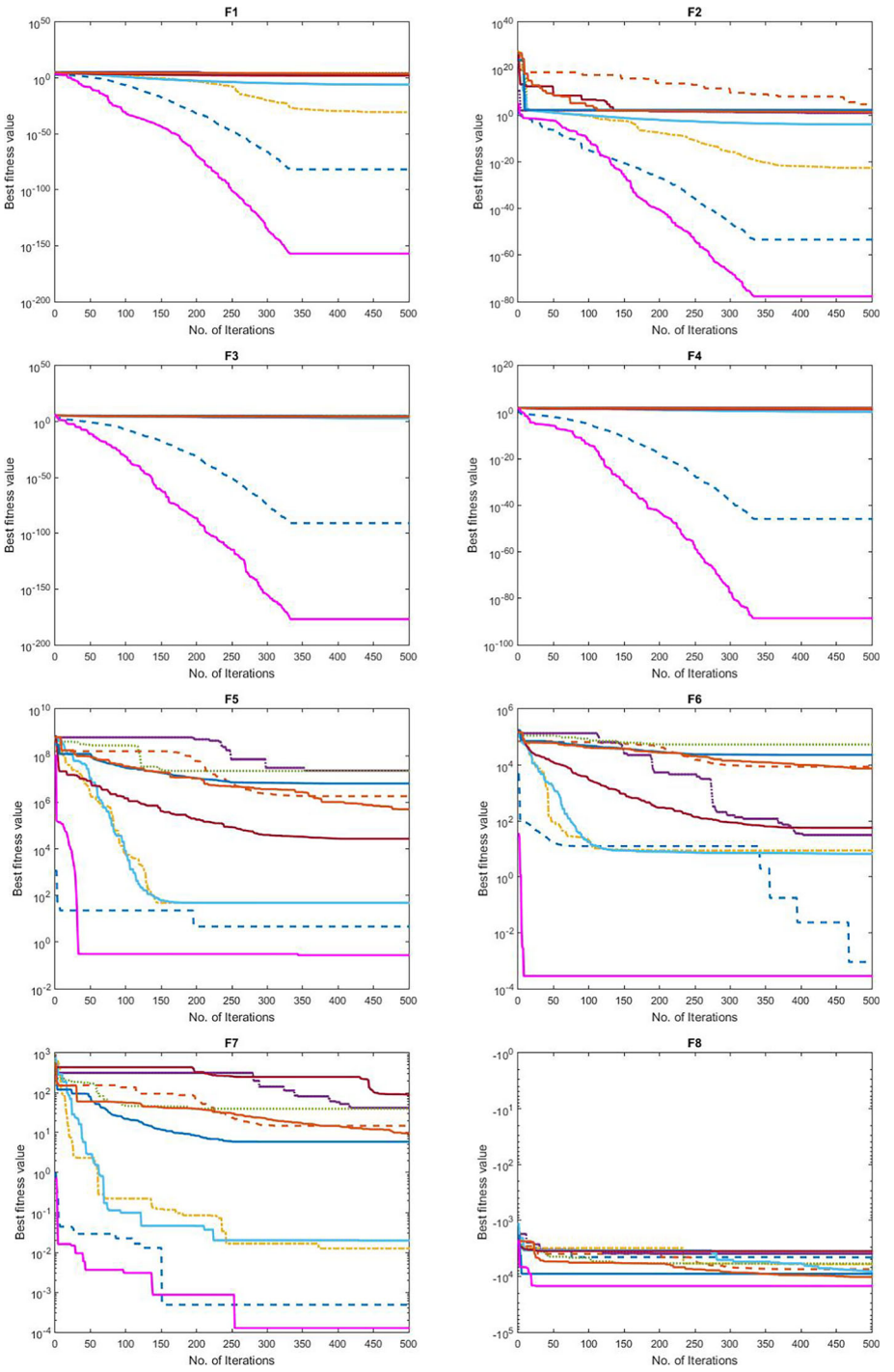


Fig. 4 Convergence behaviour of the tested algorithms on the functions (F1-F13), where the dimension is 100

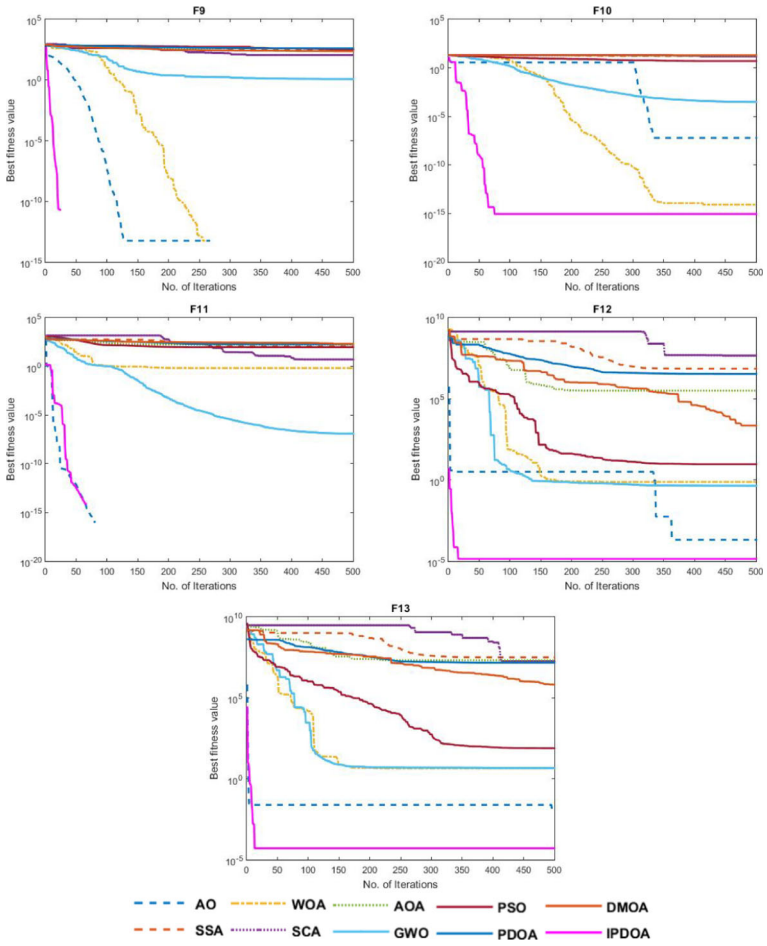


Fig. 4 continued

experiments is the PDOA. Regarding Wilcoxon's rank test, from the p-values is possible to see that in some cases, the IPDOA has some similarities with some algorithms. For example, F1 shares some statistical similarities with SSA, SCA, AOA, PSO, PDOA, and DMOA. However, it is statistically different from AO. WOA, and GWO. Also, F5 has similarities with all methods. However, in functions such as F8, the IPDOA is statistically different from the rest of the methods. These situations occur due to the nature of the problems. A high dimensionality implies more difficulties in finding the global optimal.

A graphical comparison of the performance of the ten algorithms in the functions F1-F13 in 100 dimensions is presented in Fig. 4. From the plots, it is observed how the proposed IPDOA converges faster in comparison with the rest of the methods. Besides, it can avoid suboptimal regions of the search space. For example, in F7, at 250 iterations, the algorithm escapes from a suboptimal finding to the global optimal. A similar situation occurs with F8. In other cases, such as F12 and F13, the IPDOA is clearly better than the rest of the algorithms. However, it is not the best in functions such as F9 and F11, but it is still competitive.

Figure 5 show the distribution values of the obtained fitness function. Clearly, the proposed method consistently got the best results and performed very well over several runs. This proved the searchability of the proposed method in getting the same results in different experiments.

This work undoubtedly has potential, but it's crucial to recognize its primary constraints. First, the somewhat constrained breadth of applicability is a noteworthy drawback. Although the hybrid method may successfully tackle optimization issues, its performance might not be very transferable to other problems. The algorithm's appropriateness for other optimization problems, such as continuous, discrete, or multi-objective optimization, will require more investigation and testing. Another difficulty is the algorithm's scalability. Numerous variables

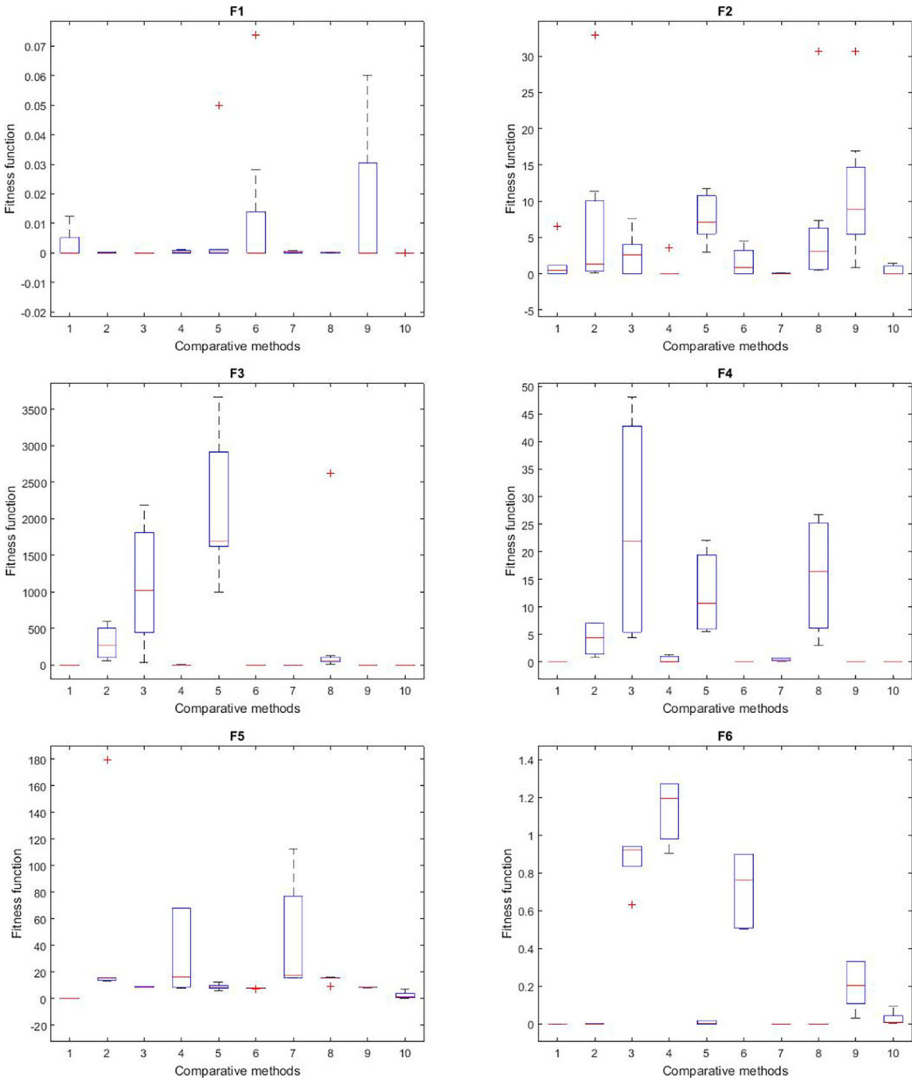


Fig. 5 Distribution plots of the obtained fitness values

or decision criteria are frequently used in optimization problems, which can exponentially expand the search area. A thorough evaluation is required to determine the upgraded algorithm's efficiency and efficacy in dealing with challenging, high-dimensional optimization issues. Due to their potential to have a major negative influence on the algorithm's usability in real-world circumstances, the computational costs related to such scalability problems must also be considered.

5 Conclusion

Recent emendations to many domains' optimization challenges necessitate using effective search techniques to solve them. This study proposes the IPDOA algorithm, a new hybrid optimization method for solving various benchmark functions. The suggested approach is centered on improving the Prairie Dog Optimization Algorithm's (PDOA) search process by utilizing the Dwarf Mongoose Optimization Algorithm's (DMOA) main update mechanism. The primary objective of the proposed IPDOA is to avoid the primary flaws in the original techniques, which include low convergence ability, imbalance in the search process, and premature convergence.

Experiments are conducted on 23 standard benchmark functions, and the results are compared with similar methods from the literature. The comparative methods are Aquila Optimizer (AO), Salp Swarm Algorithm (SSA), Whale Optimization Algorithm (WOA), Sine Cosine Algorithm (SCA), Arithmetic Optimization Algorithm (AOA), Grey Wolf Optimizer (GWO), Particle Swarm Optimizer (PSO), Prairie Dog Optimization Algorithm (PDOA), and Dwarf Mongoose Optimization Algorithm (DMOA). The results are recorded in terms of the best, worst, and average fitness function, showing that the proposed method is more vital to deal with various problems than other methods. Two statistical evaluation measures are used to validate the results of the proposed method and prove the significant improvements obtained by the proposed method. The results illustrated that the proposed method got better results in most tested cases than other state-of-the-art methods from the literature.

In future work direction, the proposed method will be investigated to solve other optimization problems like text clustering problems, data clustering problems, task schedule problems, power overflow problems, engineering design problems, cloud computing management problems, feature selection problems, intrusion detection problems, industrial engineering problems, graph coloring problems, image segmentation problems, and others. Moreover, other search operators can further enhance the proposed IPDOA to perform well for other accurate word and multiobjective problems.

Funding This research received no external funding.

Data Availibility Statement Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Compliance with ethical standards

Conflicts of interest The authors declare that there is no conflict of interest regarding the publication of this paper.

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

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Soerensen JS, Johannesen L, Grove USL, Lundhus K, Couderc J-P, Graff C (2010) A comparison of iir and wavelet filtering for noise reduction of the ecg. In: 2010 computing in cardiology, IEEE, pp 489–492
2. Liao Y, Zhao W, Wang L (2021) Improved manta ray foraging optimization for parameters identification of magnetorheological dampers. Mathematics 9(18):2230
3. Hussien AG, Hassanien AE, Houssein EH, Amin M, Azar AT (2020) New binary whale optimization algorithm for discrete optimization problems. Eng Optim 52(6):945–959
4. Hussien AG (2022) An enhanced opposition-based salp swarm algorithm for global optimization and engineering problems. J Ambient Intell Human Comput 13(1):129–150
5. Hussien AG, Hassanien AE, Houssein EH, Bhattacharyya S, Amin M (2019) S-shaped binary whale optimization algorithm for feature selection. In: Recent trends in signal and image processing, Springer, pp 79–87
6. Wu G, Pedrycz W, Suganthan PN, Mallipeddi R (2015) A variable reduction strategy for evolutionary algorithms handling equality constraints. Appl Soft Comput 37:774–786
7. Mostafa RR, Hussien AG, Khan MA, Kadry S, Hashim FA (2022) Enhanced coot optimization algorithm for dimensionality reduction. In: 2022 Fifth international conference of women in data science at prince sultan university (WiDS PSU), IEEE, pp 43–48
8. Abualigah L, Gandomi AH, Elaziz MA, Hussien AG, Khasawneh AM, Alshinwan M, Houssein EH (2020) Nature-inspired optimization algorithms for text document clustering—a comprehensive analysis. Algorithms 13(12):345
9. Nadimi-Shahraki MH, Zamani H (2022) Dmde: Diversity-maintained multi-trial vector differential evolution algorithm for non-decomposition large-scale global optimization. Expert Syst Appl 198:116895
10. Morales-Castaneda B, Zaldivar D, Cuevas E, Rodriguez A, Navarro MA (2021) Population management in metaheuristic algorithms: Could less be more? Appl Soft Comput 107:107389
11. Lynn N, Suganthan PN (2015) Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. Swarm Evol Comput 24:11–24
12. Hussien AG, Amin M (2022) A self-adaptive harris hawks optimization algorithm with opposition-based learning and chaotic local search strategy for global optimization and feature selection. Int J Mach Learn Cyber 13(2):309–336
13. Fathi H, AlSalman H, Gumaei A, Manhrawy II, Hussien AG, El-Kafrawy P (2021) An efficient cancer classification model using microarray and high-dimensional data. Comput Intell Neurosci
14. Navarro MA, Ramos-Michel A, Gaspar A, Oliva D, Hinojosa S, Mousavirad SJ, Pérez-Cisneros M (2022) Improving the convergence and diversity in differential evolution through a stock market criterion. In: International conference on the applications of evolutionary computation (Part of EvoStar), Springer, pp 157–172
15. Hussien AG, Abualigah L, Abu Zitar R, Hashim FA, Amin M, Saber A, Almotairi KH, Gandomi AH (2022) Recent advances in harris hawks optimization: A comparative study and applications. Electronics 11(12):1919
16. Hussien AG, Oliva D, Houssein EH, Juan AA, Yu X (2020) Binary whale optimization algorithm for dimensionality reduction. Mathematics 8(10):1821
17. Assiri AS, Hussien AG, Amin M (2020) Ant lion optimization: variants, hybrids, and applications. IEEE Access 8:77746–77764
18. Singh S, Singh H, Mittal N, Hussien AG , Sroubek F (2022) A feature level image fusion for night-vision context enhancement using arithmetic optimization algorithm based image segmentation. Expert Syst Appl 118272
19. Hussien AG, Hassanien AE, Houssein EH (2017) Swarming behaviour of salps algorithm for predicting chemical compound activities. In: 2017 eighth international conference on intelligent computing and information systems (ICICIS), IEEE, pp 315–320
20. Hussien AG, Houssein EH, Hassanien AE (2017) A binary whale optimization algorithm with hyperbolic tangent fitness function for feature selection. In: 2017 Eighth international conference on intelligent computing and information systems (ICICIS), IEEE, pp 166–172
21. Zamani H, Nadimi-Shahraki MH, Taghian S, Banaie-Dezfouli M (2020) Enhancement of bernstain-search differential evolution algorithm to solve constrained engineering problems. Int J Comput Sci Eng 9:386–396

22. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, Vol 4. IEEE, pp 1942–1948
23. Yang X-S, Deb S (2014) Cuckoo search: recent advances and applications. Neural Comput & Applic 24(1):169–174
24. Hussien AG, Heidari AA, Ye X, Liang G, Chen H, Pan Z (2022) Boosting whale optimization with evolution strategy and gaussian random walks: an image segmentation method. Engineering with Computers 1–45
25. Mirjalili SZ, Mirjalili S, Saremi S, Faris H, Aljarah I (2018) Grasshopper optimization algorithm for multi-objective optimization problems. Appl Intell 48(4):805–820
26. Cuevas E, Cienfuegos M, Zaldívar D, Pérez-Cisneros M (2013) A swarm optimization algorithm inspired in the behavior of the social-spider. Expert Syst Appl 40(16):6374–6384
27. Mirjalili S (2015) The ant lion optimizer. Adv Eng Softw 83:80–98
28. Hashim FA, Hussien AG (2022) Snake optimizer: A novel meta-heuristic optimization algorithm. Knowl-Based Syst 242:108320
29. Wang S, Hussien AG, Jia H, Abualigah L, Zheng R (2022) Enhanced remora optimization algorithm for solving constrained engineering optimization problems. Mathematics 10(10):1696
30. Zheng R, Hussien AG, Jia H-M, Abualigah L, Wang S, Wu D (2022) An improved wild horse optimizer for solving optimization problems. Mathematics 10(8):1311
31. Abualigah L, Elaziz MA, Hussien AG, Alsalibi B, Jalali SMJ, Gandomi AH (2021) Lightning search algorithm: a comprehensive survey. Appl Intell 51(4):2353–2376
32. Hussien AG, Amin M, Wang M, Liang G, Alsanad A, Gumaei A, Chen H (2020) Crow search algorithm: theory, recent advances, and applications. IEEE Access 8:173548–173565
33. Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomi AH (2021) The arithmetic optimization algorithm. Comput Methods Appl Mech Eng 376:113609
34. Abualigah L, Yousri D, Abd Elaziz M, Ewees AA, Al-Qaness MA, Gandomi AH (2021) Aquila optimizer: a novel meta-heuristic optimization algorithm. Comput Ind Eng 157:107250
35. Hussien AG, Amin M, Abd El Aziz M (2020) A comprehensive review of moth-flame optimisation: variants, hybrids, and applications. J Exp Theor Artif Intell 32(4):705–725
36. Abualigah L, Abd Elaziz M, Sumari P, Geem ZW, Gandomi AH (2022) Reptile search algorithm (rsa): A nature-inspired meta-heuristic optimizer. Expert Syst Appl 191:116158
37. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61
38. Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–73
39. Passino KM (2010) Bacterial foraging optimization. Int J Swarm Intell Res (IJSIR) 1(1):1–16
40. Price KV (2013) Differential evolution. In: Handbook of optimization, Springer, pp 187–214
41. Uymaz SA, Tezel G, Yel E (2015) Artificial algae algorithm (aaa) for nonlinear global optimization. Appl Soft Comput 31:153–171
42. Simon D (2008) Biogeography-based optimization. IEEE Trans Evol Comput 12(6):702–713
43. Kirkpatrick S, Gelatt CD Jr, Vecchi MP (1983) Optimization by simulated annealing. Science 220(4598):671–680
44. Erol OK, Eksin I (2006) A new optimization method: big bang-big crunch. Adv Eng Softw 37(2):106–111
45. Abualigah L, Elaziz MA, Sumari P, Khasawneh AM, Alshinwan M, Mirjalili S, Shehab M, Abuaddous HY, Gandomi AH (2022) Black hole algorithm: A comprehensive survey. Appl Intell 1–24
46. Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) Gsa: a gravitational search algorithm. Inf Sci 179(13):2232–2248
47. Mirjalili S, Mirjalili SM, Hatamlou A (2016) Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput & Applic 27(2):495–513
48. Hashim FA, Houssein EH, Mabrouk MS, Al-Atabany W, Mirjalili S (2019) Henry gas solubility optimization: A novel physics-based algorithm. Futur Gener Comput Syst 101:646–667
49. Agushaka JO, Ezugwu AE, Abualigah L (2022) Dwarf mongoose optimization algorithm. Comput Methods Appl Mech Eng 391:114570
50. Ezugwu AE, Agushaka JO, Abualigah L, Mirjalili S, Gandomi AH (2022) Prairie dog optimization algorithm. Neural Comput & Applic 1–49
51. Gürses D, Bureerat S, Sait SM, Yıldız AR (2021) Comparison of the arithmetic optimization algorithm, the slime mold optimization algorithm, the marine predators algorithm, the salp swarm algorithm for real-world engineering applications. Mater Test 63(5):448–452
52. Khatir S, Tiachacht S, Le Thanh C, Ghandourah E, Mirjalili S, Wahab MA (2021) An improved artificial neural network using arithmetic optimization algorithm for damage assessment in fgm composite plates. Compos Struct 273:114287

53. Premkumar M, Jangir P, Kumar BS, Sowmya R, Alhelou HH, Abualigah L, Yildiz AR, Mirjalili S (2021) A new arithmetic optimization algorithm for solving real-world multiobjective cec-2021 constrained optimization problems: diversity analysis and validations. IEEE Access 9:84263–84295
54. Al-Shourbaji I, Kachare PH, Alshathri S, Duraibi S, Elnaim B, Abd Elaziz M (2022) An efficient parallel reptile search algorithm and snake optimizer approach for feature selection. Mathematics 10(13):2351
55. Deeb H, Sarangi A, Mishra D, Sarangi SK (2023) Improved black hole optimization algorithm for data clustering. Journal of King Saud University-Computer and Information Sciences
56. Fan Q, Chen Z, Li Z, Xia Z, Lin Y (2020) An efficient refracted salp swarm algorithm and its application in structural parameter identification. Engineering with Computers 1–15
57. Zhang X, Zhao K, Niu Y (2020) Improved harris hawks optimization based on adaptive cooperative foraging and dispersed foraging strategies. IEEE Access 8:160297–160314
58. Kamboj VK, Nandi A, Bhadoria A, Sehgal S (2020) An intensify harris hawks optimizer for numerical and engineering optimization problems. Appl Soft Comput 89:106018
59. Ewees AA, Abd Elaziz M, Houssein EH (2018) Improved grasshopper optimization algorithm using opposition-based learning. Expert Syst Appl 112:156–172
60. Abualigah L, Shehab M, Alshinwan M, Alabool H (2020) Salp swarm algorithm: a comprehensive survey. Neural Comput & Applic 32(15):11195–11215
61. Nadimi-Shahraki MH, Taghian S, Mirjalili S, Abualigah L, Abd Elaziz M, Oliva D (2021) Ewoa-opf: effective whale optimization algorithm to solve optimal power flow problem. Electronics 10(23):2975
62. Abualigah L, Diabat A (2021) Advances in sine cosine algorithm: a comprehensive survey. Artif Intell Rev 54(4):2567–2608
63. Otair M, Ibrahim OT, Abualigah L, Altalhi M, Sumari P (2022) An enhanced grey wolf optimizer based particle swarm optimizer for intrusion detection system in wireless sensor networks. Wirel Netw 28(2):721–744
64. Abualigah LM, Khader AT, Hanandeh ES (2018) A new feature selection method to improve the document clustering using particle swarm optimization algorithm. J Comput Sci 25:456–466

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Authors and Affiliations

Laith Abualigah[1,2,3,4,5,6,7] · Diego Oliva[8] · Heming Jia[9] · Faiza Gul[10] · Nima Khodadadi[11] · Abdelazim G Hussien[12,13] · Mohammad Al Shinwan[14] · Absalom E. Ezugwu[15] · Belal Abuhaija[16] · Raed Abu Zitar[17]

Diego Oliva
diego.oliva@cucei.udg.mx

Heming Jia
jiaheming@fjsmu.edu.cn

Faiza Gul
faiza.gul@aack.au.edu.pk

Nima Khodadadi
nkhod002@fiu.edu

Abdelazim G Hussien
abdelazim.hussien@liu.se

Mohammad Al Shinwan
moshanwan@yahoo.com

Absalom E. Ezugwu
ezugwua@ukzn.ac.za

Raed Abu Zitar
raed.zitar@sorbonne.ae

1 Computer Science Department, Prince Hussein Bin Abdullah Faculty for Information Technology, Al al-Bayt University, Mafraq 25113, Jordan

2 Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman 19328, Jordan

3 Department of Electrical and Computer Engineering, Lebanese American University, Byblos 13-5053, Lebanon

4 MEU Research Unit, Middle East University, Amman 11831, Jordan

5 Applied Science Research Center, Applied Science Private University, Amman 11931, Jordan

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

7 School of Engineering and Technology, Sunway University Malaysia, Petaling Jaya 27500, Malaysia

8 Depto. de Innovación Basada en la Información y el Conocimiento, Universidad de Guadalajara, CUCEI, Guadalajara, Jal, Mexico

9 School of Information Engineering, Sanming University, Sanming 365004, China

10 Department of Electrical Engineering, Aerospace and Avionics, Campus Kamra, Air University, Kamra, Pakistan

11 Department of Civil and Environmental Engineering, Florida International University, Miami, FL, USA

12 Department of Computer and Information Science, Linköping University, Linköping, Sweden

13 Faculty of Science, Fayoum University, Faiyum, Egypt

14 Faculty of Information Technology, Applied Science Private University, Amman 11931, Jordan

15 Unit for Data Science and Computing, North-West University, 11 Hoffman Street, 2520 Potchefstroom, South Africa

16 Department of Computer Science, Wenzhou-Kean University, Wenzhou 325015, China

17 Sorbonne Center of Artificial Intelligence, Sorbonne University-Abu Dhabi, 38044 Abu Dhabi, United Arab Emirates