Parrot optimizer: Algorithm and applications to medical problems

Junbo Lian [1,2,3], Guohua Hui [1,2,3], Ling Ma [1,2,3], Ting Zhu [1,2,3], Xincan Wu [1,2,3], Ali Asghar Heidari [4],

Yi Chen [5], Huiling Chen [5*]

[1] College of Mathematics and Computer Sciences, Zhejiang A & F University, Hangzhou 311300
junbolian@qq.com, deliver1982@163.com, maling@stu.zafu.edu.cn, ai@stu.zafu.edu.cn, y.yr.r@qq.com
[2] Key Laboratory of Forestry Sensing Technology and Intelligent Equipment of Department of Forestry, Zhejiang A & F University, Hangzhou 311300
junbolian@qq.com, deliver1982@163.com, maling@stu.zafu.edu.cn, ai@stu.zafu.edu.cn, y.yr.r@qq.com
[3] Key Laboratory of Forestry Intelligent Monitoring and Information Technology of Zhejiang Province, Zhejiang A & F University, Hangzhou 311300
junbolian@qq.com, deliver1982@163.com, maling@stu.zafu.edu.cn, ai@stu.zafu.edu.cn, y.yr.r@qq.com
[4] School of Surveying and Geospatial Engineering, College of Engineering, University of Tehran, Tehran, Iran
as_heidari@ut.ac.ir
[5] College of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou 325035, PR China
kenyoncy2016@gmail.com, chenhuiling.jlu@gmail.com
*Corresponding Author: Huiling Chen (chenhuiling.jlu@gmail.com)

Abstract:

Stochastic optimization methods have gained significant prominence as effective techniques in contemporary research, addressing complex optimization challenges efficiently. This paper introduces the Parrot Optimizer (PO), an efficient optimization method inspired by key behaviors observed in trained Pyrrhura Molinae parrots. The study features qualitative analysis and comprehensive experiments to showcase the distinct characteristics of the Parrot Optimizer in handling various optimization problems. Performance evaluation involves benchmarking the proposed PO on 35 functions, encompassing classical cases and problems from the IEEE CEC 2022 test sets, and comparing it with eight popular algorithms. The results vividly highlight the competitive advantages of the PO in terms of its exploratory and exploitative traits. Furthermore, parameter sensitivity experiments explore the adaptability of the proposed PO under varying configurations. The developed PO demonstrates effectiveness and superiority when applied to engineering design problems. To further extend the assessment to real-world applications, we included the application of PO to disease diagnosis and medical image segmentation problems, which are highly relevant and significant in the medical field. In conclusion, the findings substantiate that the PO is a promising and competitive algorithm, surpassing some existing algorithms in the literature. The supplementary files and open source codes of the proposed parrot optimizer (PO) is available at https://aliasgharheidari.com/PO.html and https://github.com/junbolian/PO.

Keywords: Optimization; Metaheuristic; Swarm optimization; Medical problem; Genetic algorithm; Parrot Optimizer

1 Introduction

With the ongoing advancements in artificial intelligence (AI), the realm of optimization faces an array of challenges that are encountered in both academic research and practical engineering applications. These challenges often encompass characteristics such as nonlinearity, discontinuity, uncertainty, high dimensionality, multi-objectivity, and non-convexity [1-3]. As a result, optimization algorithms find extensive application across various industries and domains, such as medicine [4, 5], manufacturing [6], agriculture [7, 8], and tourism [9]. Currently, conventional optimization techniques encompass classical methods like linear programming, nonlinear programming, mixed-integer programming [10], and integer programming, as well as more recent developments such as Newton's method [11], the conjugate gradient method [12], and the gradient descent method [13]. While these methods can globally optimize specific problems, they often require the feasible domain to be convex, the objective function to be continuously differentiable, or impose additional constraints [14]. However, complex optimization problems frequently are non-differentiable and non-convex or with multimodal characteristics and they prove challenging to be solved by traditional frameworks [15-17]. Consequently, traditional optimization methods face considerable difficulty when addressing large-scale and complex problems [18, 19].

In light of these challenges, there is a demand for precise, efficient, and robust optimization algorithms capable of tackling these hard problems [20-22]. Metaheuristic optimization algorithms have emerged as a prominent and widely adopted solution to meet this demand. Most of these methods exhibit stochastic properties and possess the ability to approximate optimal solutions across diverse optimization scenarios [23, 24]. Notably, the strength of metaheuristic optimization algorithms over traditional methods is a feature of their gradient-free approach and their capacity to avoid local optima [25].

Most metaheuristic algorithms draw inspiration from natural or social phenomena and are widely applied to optimization problems in various fields [26, 27]. These algorithms can be categorized into four primary types based on their internal operation processes. The first category includes evolution-based algorithms, which primarily emulate the natural law of *"survival of the fittest"* to advance the overall population and ultimately converge to an optimal or sub-optimal solution. Notable examples of evolutionary algorithms within this category are genetic algorithm (GA) [28] and differential evolutionary algorithm (DE) [29]. With the increasing popularity of evolutionary algorithms, numerous related algorithms have been developed, such as evolutionary strategy (ES) [30], evolutionary programming (EP) [31], and gene expression programming (GEP) [32].

While biological evolutionary algorithms have been extensively studied, some researchers have identified that biological groups also exhibit behaviors aimed at seeking some advantages. Consequently, population-based class has emerged. These algorithms aim to iteratively update solutions that are close to optimal by modeling collaboration or information exchange between groups to find a global optimal solution or an approximation, eventually. In recent years, numerous algorithms have been developed, including particle swarm optimization (PSO) [33, 34], Harris hawk optimization (HHO) [35], slime mould algorithm (SMA) [36], hunger games search (HGS) [37], Runge-Kutta optimizer (RUN) [20], and INFO Optimizer [21]. Additionally, in the realm of swarm intelligence algorithms, researchers have introduced several other algorithms inspired by biological swarms, such as ant colony optimization (ACO) [38, 39], artificial bee colony (ABC) [40], the moth-flame optimizer [41], and bat algorithm (BA) [42].

Another class of optimization algorithms, which draws inspiration from real-life laws rooted in the natural world, comprises methods based on natural phenomena. These algorithms primarily harness physical and chemical principles found in nature. Notable algorithms falling into this category include the gravitational search algorithm [43], the multi-verse optimizer [44], and the RIME optimizer [45].

The final category of optimization methods encompasses human-based approaches inspired by human cooperation and collective behavior [46]. An example of an algorithm frequently employed within this group is the imperialist competition algorithm [47], which derives its inspiration from socio-political growth practices observed in human societies. Another algorithm in this category is the teaching-learning-based optimization algorithm [48, 49].

While all algorithms play a significant role in metaheuristic optimization, they still present certain limitations:

(i) General metaheuristic algorithms typically comprise two primary phases: exploration and exploitation. Striking the right balance between these phases is a challenging yet crucial task, directly influencing the overall algorithm's performance.

(ii) Parameters have a weighty influence on the optimization performance of most algorithms. Determining the optimal parameters for a specific optimization problem is exceedingly challenging. When a newly proposed algorithm lacks a comprehensive qualitative analysis and does not address parameter sensitivity, it can struggle to tackle complex problems thoroughly and effectively [45].

(iii) Some algorithms emphasize introducing novel metaphors without highlighting the computational performance advantages in solving complex problems. This approach can lead to inefficiency, causing these algorithms to underperform when applied to different problem types and yield suboptimal results.

Researchers typically avoid to always utilize a single algorithm for different problems, which is a practice aligned with the No Free Lunch (NFL) theorem [50, 51]. This theorem theorizes that no individual algorithm can comprehensively address all optimization problems with best ever performance and best quality solutions. Consequently, it becomes imperative to adapt and modify existing algorithms or introduce new methodologies to tackle usual scenarios and meet the challenges that continue to emerge more effectively. Motivated by this rationale, we drew inspiration from the domesticated Pyrrhura Molinae population, a kind of parrot, to introduce an efficient optimization method, called parrot optimizer.

The PO is an efficient metaheuristic algorithm that draws inspiration from the foraging, staying, communicating, and fear of strangers' behaviors observed in domesticated Pyrrhura Molinae parrots. These behaviors are encapsulated in four distinct formulas to facilitate the search for optimal solutions. In contrast to traditional metaheuristic algorithms that follow separate exploration and exploitation phases, each individual within the PO population randomly exhibits one of these four behaviors during each iteration. This approach provides a more fitting representation of the behavioral randomness observed in domesticated Pyrrhura Molinae parrots and significantly enhances population diversity. By deviating from the conventional two-phase structure of exploration-exploitation, the PO effectively mitigates the risk of trapping in local optima while maintaining solution quality. The stochastic structure of PO distinguishes it from traditional algorithms, making it particularly well-suited for avoiding local optima and applicable to real-world problem-solving, especially in the medical domain.

In our experiments, we conducted qualitative analyses and parameter sensitivity tests to reveal the characteristics and adaptability of the PO algorithm. To comprehensively assess the algorithm, we employed 23 classical benchmark functions [31, 52] and 12 IEEE CEC 2022 test functions [53] for comparison against 8 widely used metaheuristic methods. Furthermore, we investigated the performance of the PO in solving diverse optimization problems with varying parameters. Additionally, we validated the algorithm's capacity to address 5 computational challenges by applying it to three classical engineering optimization problems and two medical problems.

This paper makes substantial contributions in several key areas:

1. Introduce the parrot optimizer, an efficient optimizer for various optimization cases.
2. The construction of an efficient optimization mechanism in PO is characterized by its absence of a clear distinction between the Exploration and Exploitation (E&P) phase yet enhanced optimization capabilities.
3. Conducting a comprehensive qualitative analysis and parameter sensitivity experiments to deeply explore the characteristics of the PO algorithm, enhancing its applicability across various optimization problems.
4. Validation of the algorithm's performance through comparative experiments with 8 widely used algorithms demonstrates strong competitiveness of the PO in solving diverse optimization challenges.
5. Applying the PO algorithm to 5 real-world optimization problems substantiates its potential to address a wide range of practical optimization tasks.

The rest of the paper is organized as follows: Section 2 provides a detailed description of our proposed PO method. Section 3 presents the results of experiments conducted to solve various benchmark functions and real-world problems. Finally, Section 4 offers conclusions and outlines future research directions.

2 The parrot optimizer (PO)

This section explains the overall background of the PO and the formulated optimization models.

2.1 Inspiration

The Pyrrhura Molinae, a well-liked parrot species, is a popular choice for pet owners owing to its attractive features, close bonding with its owners, and ease of training [54, 55]. Previous studies and breeding efforts have revealed that Pyrrhura Molinae exhibits four distinct behavioral traits: foraging, staying, communicating, and a fear of strangers [56, 57]. These behaviors, illustrated in Fig. 1 within real-world contexts, form the basis of our motivation for designing the PO.

(a) foraging (b) staying (c) communicating (d) fear of strangers

Fig. 1. Four behaviors of Pyrrhura Molinae

- The ***foraging*** behavior of domesticated Pyrrhura Molinae is fascinating, as individuals choose to forage in small groups where food is abundant [54]. They can find the food by heading toward it, utilizing their owner's location and the group's presence. They enhance their search using smell and visual hints.
- The ***staying*** behavior involves Pyrrhura Molinae perching randomly on various areas of their owner's body.
- These sociable birds produce distinctive calls to ***communicate*** within their group, serving both for social interaction and information spread.
- The natural ***fear of strangers***, a common trait among birds, prompts Pyrrhura Molinae to move away from unfamiliar individuals and seek safety with their owners for protection [58].
- Importantly, the unpredictability of Pyrrhura Molinae behavior stresses the motivation for our design, as these four behaviors occur randomly in each individual during each iteration within domesticated flocks.

2.2 Mathematical model of PO

2.2.1 Population initialization

The initialization formulation for the proposed PO, considering a swarm size of N, maximum iterations of Max_{iter}, and search space limits of lb (lower bound) and ub (upper bound), can be shown as:

$$X_i^0 = lb + rand(0,1) \cdot (ub - lb) \tag{1}$$

where $rand(0,1)$ denotes a random number in the range [0, 1] and X_i^0denotes the position of the i^{th} Pyrrhura Molinae in the initial phase.

2.2.2 Foraging behavior

During the foraging behavior in PO, they estimate the approximate location of food primarily by observing the food's location or by considering the owner's position, then they fly towards the respective location. Therefore, the positional movement follows the equation:

$$X_i^{t+1} = (X_i^t - X_{best}) \cdot Levy(dim) + rand(0,1) \cdot (1 - \frac{t}{Max_{iter}})^{\frac{2t}{Max_{iter}}} \cdot X_{mean}^t \tag{2}$$

In Eq. (2), X_i^t denotes the present location, while X_i^{t+1} means the location of the succeeding update. X_{mean}^t represents the average location inside the present population, and $Levy(D)$ denotes the Levy distribution, It is used to describe the flight of parrots. X_{best} denotes the best position that has been searched from initialization to the current, and it also represents the host's current position. t denotes the current number of iterations. $(X_i^t - X_{best}) \cdot Levy(dim)$ indicates movement based on one's position in relation to the owner, and $rand(0,1) \cdot (1 - \frac{t}{Max_{iter}})^{\frac{2t}{Max_{iter}}} \cdot X_{mean}^t$ indicates observation of the position of the population as a whole to further target the orientation of the food. The process is depicted in Fig. 2.

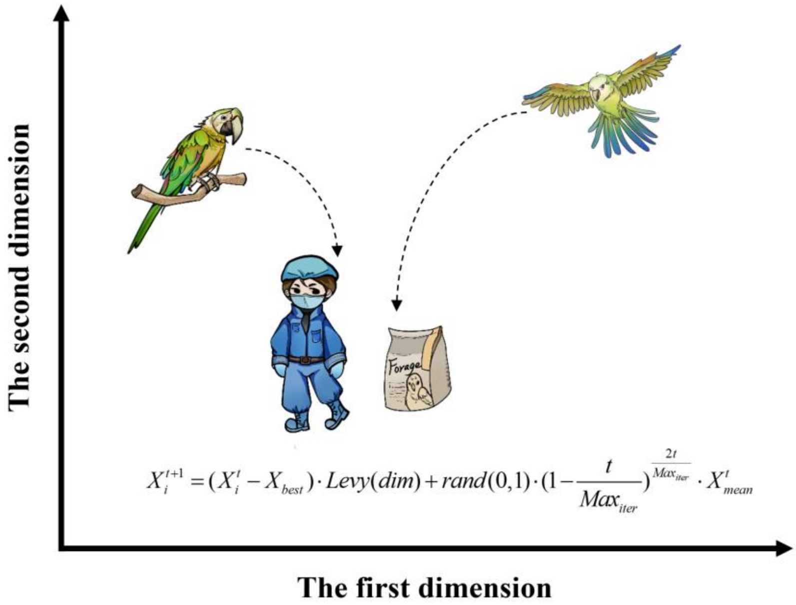


Fig. 2. The foraging behavior

The average location of the current swarm, is shown by X_{mean}^t, is attained using the formula shown in Eq. (3).

$$X_{mean}^t = \frac{1}{N} \sum_{k=1}^{N} X_k^t \tag{3}$$

The Levy distribution can be obtained based on rule in Eq. (4), where γ is assigned the value of 1.5.

$$\begin{cases} Levy(dim) = \frac{\mu \cdot \sigma}{|v|^{\frac{1}{\gamma}}} \\ \mu \sim N(0, dim) \\ v \sim N(0, dim) \\ \sigma = \left(\frac{\Gamma(1+\gamma) \cdot sin(\frac{\pi\gamma}{2})}{\Gamma(\frac{1+\gamma}{2}) \cdot \gamma \cdot 2^{\frac{1+\gamma}{2}}}\right)^{\gamma+1} \end{cases} \tag{4}$$

2.2.3 Staying behavior

Pyrrhura Molinae is a highly sociable creature, and its staying behavior primarily involves the sudden flight to any part of its owner's body, where it remains stationary for a certain period. This process is shown in Fig. 3. This process can be represented as:

$$X_i^{t+1} = X_i^t + X_{best} \cdot Levy(dim) + rand(0,1) \cdot ones(1, dim) \tag{5}$$

where $ones(1, dim)$ denotes the all-1 vector of dimension dim. $X_{best} \cdot Levy(dim)$ denotes the flight to the host, and $rand(0,1) \cdot ones(1, dim)$ denotes the process of randomly stopping at a part of the host's body.

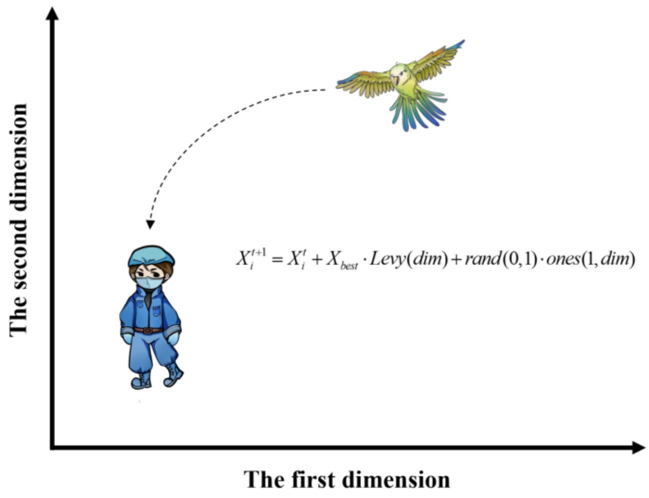


Fig. 3. The staying behavior

2.2.4 Communicating behavior

Pyrrhura Molinae parrots are inherently social animals characterized by close communication within their groups. This communication behavior encompasses flying to the flock and communicating without flying to the flock. In the PO, both behaviors are assumed to occur with equal probability, and the mean position of the current population is employed to symbolize the center of the flock. This process is shown in Fig. 4. This process can be represented as:

$$X_i^{t+1} = \begin{cases} 0.2 \cdot rand(0,1) \cdot (1 - \frac{t}{Max_{iter}}) \cdot (X_i^t - X_{mean}^t), P \leq 0.5 \\ 0.2 \cdot rand(0,1) \cdot exp(-\frac{t}{rand(0,1) \cdot Max_{iter}}), P > 0.5 \end{cases} \tag{6}$$

where, $0.2 \cdot rand(0,1) \cdot (1 - \frac{t}{Max_{iter}}) \cdot (X_i^t - X_{mean}^t)$ denotes the process of an individual joining a parrot's group to communicate and $0.2 \cdot rand(0,1) \cdot exp(-\frac{t}{rand(0,1) \cdot Max_{iter}})$ denotes the process of an individual flying away immediately after communicating. Both behaviors are feasible and, as such, are implemented using a randomly generated P within the range of [0, 1].

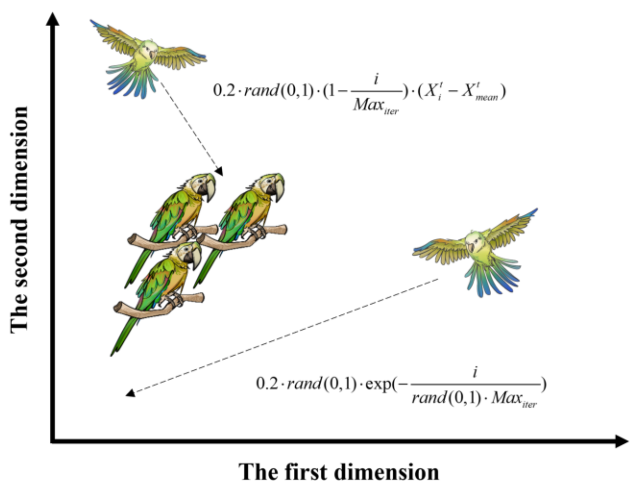


Fig. 4. The communicating behavior

2.2.5 Fear of strangers' behavior

As a general rule, birds exhibit a natural fear of strangers, and Pyrrhura Molinae parrots are not an exception. Their behavior of distancing themselves from unfamiliar individuals and seeking safety with their owners in search of a secure environment is illustrated in Fig. 5, as described below:

$$X_i^{t+1} = X_i^t + rand(0,1) \cdot cos(\,0.5\pi \cdot \frac{t}{Max_{iter}}) \cdot (X_{best} - X_i^t)$$

$$- cos(rand(0,1) \cdot \pi) \cdot (\frac{t}{Max_{iter}})^{\frac{2}{Max_{iter}}} \cdot (X_i^t - X_{best}) \quad (7)$$

where $rand(0,1) \cdot cos(\,0.5\pi \cdot \frac{t}{Max_{iter}}) \cdot (X_{best} - X_i^t)$ shows the process of reorientating to fly towards the owner and $cos(rand(0,1) \cdot \pi) \cdot (\frac{t}{Max_{iter}})^{\frac{2}{Max_{iter}}} \cdot (X_i^t - X_{best})$ shows the process of moving away from the strangers.

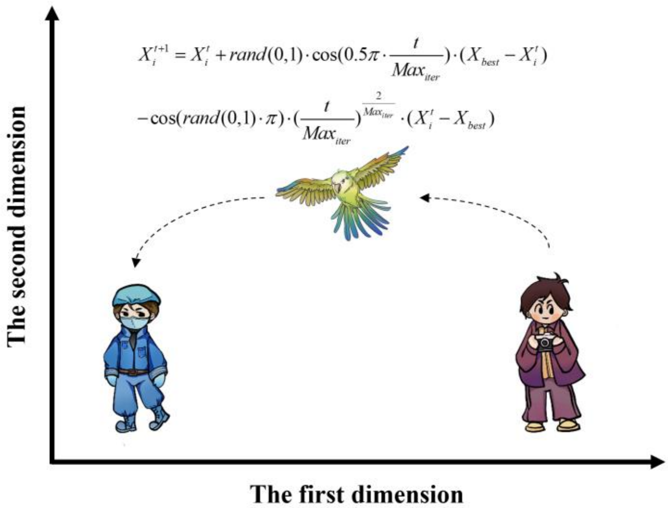


Fig. 5. The fear of strangers' behavior

2.3 Pseudo-code of the PO algorithm

As per **Algorithm 1**, the PO optimization procedure begins by randomly generating a predefined set of candidate solutions, referred as the population. Utilizing a sequence of behaviors, PO's search strategy navigates locations near the optimal solution or where the best solution has been discovered. During the optimization process, each solution adapts its position dynamically, influenced by the best solution identified thus far in the PO algorithm. The search process in PO persists until the predetermined termination criterion is satisfied. The algorithm's complete structure, illustrated through pseudo-code in Algorithm 1 and visually in Fig. 6, is provided, offering a comprehensive roadmap for the entire optimization procedure, including its iterative steps and search strategies. PO leverages the advantages of both exploration and exploitation, enabling it to navigate the search space effectively while converging towards optimal solutions.

Algorithm 1: Pseudo-code of the PO algorithm

```
Initialize the PO parameters
Initialize the solutions' positions randomly
For i = 1:Max_iter do
    Calculate the fitness function
    Find the best position
    For j = 1:N do
    St = randi([1, 4])
        Behavior 1: The foraging behavior
        If St == 1 Then
            Update position by Eq. (2)
        Behavior 2: The staying behavior
        Elseif St == 2 Then
            Update position by Eq. (5)
        Behavior 3: The communicating behavior
        Elseif St == 3 Then
            Update position by Eq. (6)
      Behavior 4: The fear of strangers' behavior
      Elseif St == 4 Then
            Update position by Eq. (7)
        End
    End
    Return the best solution
End
```

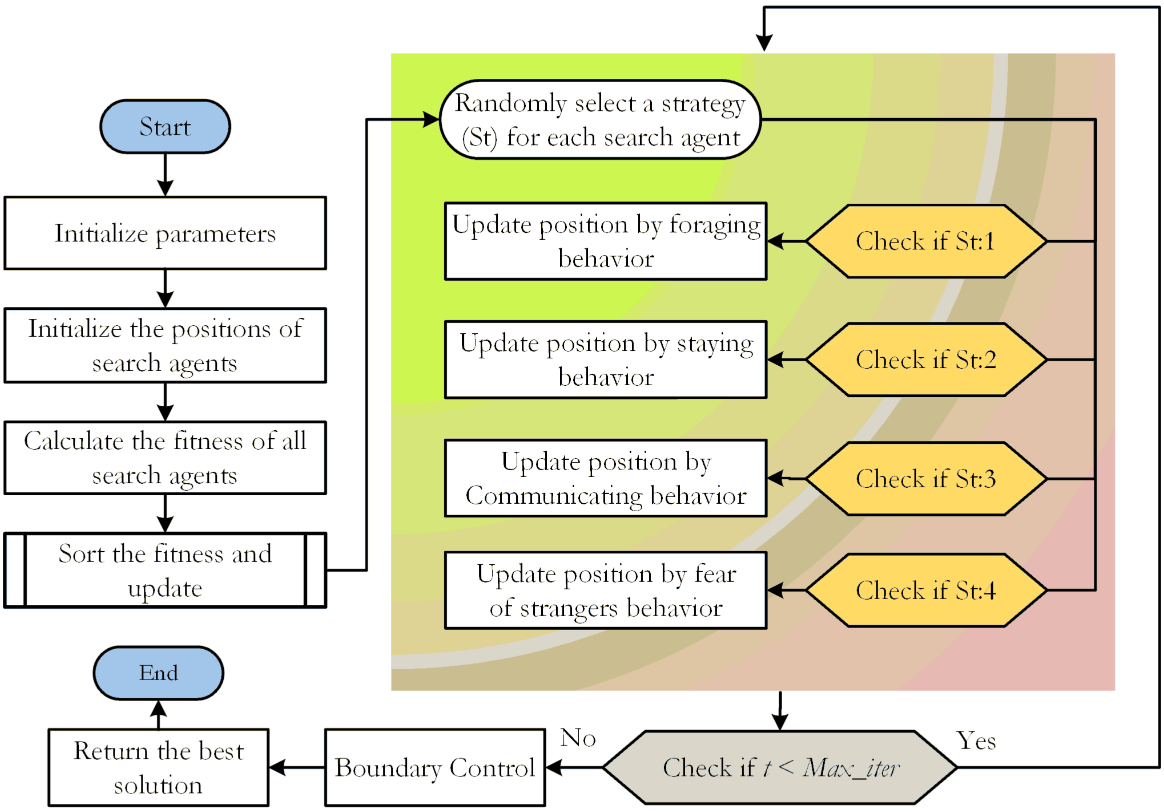


Fig. 6. Flowchart of PO algorithm

2.4 The computational complexity of the PO

This section provides an overview of the overall computational complexity of the PO approach. The computational complexity of PO is primarily dependent on three fundamental aspects: the initialization of solutions, the calculation of fitness functions, and the updating of solutions. Consider N as the count of solutions and $O(N)$ as the computational complexity associated with the initialization of these solutions. The updating processes exhibit computational complexity as follows $O(T \times N) + O(T \times N \times dim) + O(T \times N \times logN)$. This involves searching for optimal locations and updating the locations of all solutions, where T represents the total number of iterations and N signifies the dimension size of the problem at hand.

3 Results and discussion

We assess the efficacy of the PO algorithm by exposing it to testing across 23 classical benchmark functions [31], 12 IEEE CEC 2022 test functions [53], and 5 real-world problems spanning diverse domains. The IEEE CEC2005, recognized as a classic test function, has been widely utilized for assessing the performance of optimizers in recent years [46, 59-61]. The introduction of CEC2022 by IEEE as the latest test function serves as a contemporary benchmark, effectively highlighting the competitiveness of PO. In addition to these classic benchmarks, we chose three engineering design optimization problems from IEEE in 2020 to evaluate PO's proficiency in addressing engineering challenges [20, 21, 35, 36, 46]. To further extend the assessment to real-world applications, we included the application of PO to disease diagnosis and medical image segmentation problems, which are highly relevant and significant in the medical field [5, 62-64]. Through this diverse set of benchmarks and challenges, we aim to showcase the versatility and effectiveness of PO in addressing various optimization and real-world problems.

Subsequently, we conduct a comparative analysis, comparing the performance results of PO with those of eight well-established algorithms documented in the existing literature. These algorithms include the harris hawks optimization (HHO) [35], whale optimization algorithm (WOA) [65], remora optimization algorithm (ROA) [66], fire hawk optimizer (FHO) [67], arithmetic optimization algorithm (AOA) [68], sine cosine algorithm (SCA) [59], multi-verse optimizer (MVO) [44], and bat algorithm (BA) [42]. In this research, we focus on optimization capacities and performance of methods, and we do not confirm or reject validity and novelty of the modeling in FHO, AOA, and ROA approaches.

Table 1 provides a comprehensive overview of the algorithms used in this study, including their respective control parameters. Our experimentation encompassed the application of these algorithms to classical benchmark functions, CEC 2022 test functions, and engineering design problems, all implemented using MATLAB R2019b. Each algorithm experienced 30 independent runs. To measure the quality of the solutions obtained, we employed a set of five performance metrics, including the best, worst, average, standard deviation (STD), and median values. These metrics were utilized to convey the outcomes achieved through the PO approach.

Table 1. Parameter settings

Algorithm	Name of parameters	Value of parameters
HHO	E_0	[-1,1]
WOA	α b	Decreased from 2 to 0 2
ROA	-	-
FHO	-	-
AOA	α μ	5 0.05
SCA	-	-
MVO	WEP_{max} WEP_{min}	1 0.2
BA	A r	0.5 0.5

When evaluating the effectiveness and potential of optimizers, researchers frequently employ a collection of 23 established evaluation measures. These measures have been widely used across different studies exploring optimization algorithms. They fall into three main categories: those with a single peak point, those with multiple peak points, and those with a set dimension for multiple peak points. Details regarding these measures, including their types, search parameters, and theoretical optimal values, are elaborated in Tables A1 to A3.

In addition, to further underscore the effectiveness of the PO algorithm, the CEC 2022 test functions was selected. These test functions encompass a diverse array of characteristics, including unimodal, multimodal, hybrid, and composition functions. To provide a more comprehensive understanding of these selected functions, detailed information is presented in Table A4.

Table A1. Unimodal benchmark functions

Function	Dim	Range	Shift position	f_{min}
$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	$[-100,100]$	$[-30,-30,\cdots,-30]$	0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	$[-10,10]$	$[-3,-3,\cdots,-3]$	0
$F_3(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2$	30	$[-100,100]$	$[-30,-30,\cdots,-30]$	0
$F_4(x) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2$	30	$[-100,100]$	$[-30,-30,\cdots,-30]$	0
$F_5(x) = \sum_{i=1}^{n-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2 \right]$	30	$[-30,30]$	$[-15,-15,\cdots,-15]$	0
$F_6(x) = \sum_{i=1}^{n} (\lvert x_i + 0.5 \rvert)^2$	30	$[-100,100]$	$[-750,-750,\cdots,-750]$	0
$F_7(x) = \sum_{i=1}^{n} ix^4 + random0,1)$	30	$[-1.28,1.28]$	$[-0.25,-0.25,\cdots,-0.25]$	0

Table A2. Multimodal benchmark functions.

Function	Dim	Range	Shift position	f_{min}
$F_8(x) = \sum_{i=1}^{n} -x_i sin\left(\sqrt{\lvert x_i \rvert}\right)$	30	$[-500,500]$	$[-300,\cdots,-300]$	-12569.5
$F_9(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\,cos(2\pi x_i) + 10\right]$	30	$[-5.12,5.12]$	$[-2,-2,\cdots,-2]$	0
$F_{10}(x) = -20exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - exp\left(\frac{1}{n}\sum_{i=1}^{n} cos(\,2\pi x_i)\right) + 20 + e$	30	$[-32,32]$		0
$F_{11}(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} cos\left(\frac{x_i}{\sqrt{(i)}}\right) + 1$	30	$[-600,600]$	$[-400,\cdots,-400]$	0

$F_{12}(x) = \frac{\pi}{n}\left\{10sin(\pi y_i) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10sin^2(\pi y_{i+1})] + (y_n - 1)^2\right\} + \sum_{i=1}^{n} u(x_i, 10,100,4)$ $y_i = 1 + \frac{x_i + 1}{4}$ $u(x_i, a, k, m) = \begin{cases} k(x_i - \\ 0 - a < \\ k(-x_i - \end{cases}$	30	$[-50,50]$	$[-30,-30,\cdots,-3$	0
$F_{13}(x) = 0.1\left\{sin^2(3\pi x_1) + \sum_{i=1}^{n1}(x_i - 1)^2[1 + sin^2(3\pi x_i + 1)] + (x_n - 1)^2[1 + sin^2(2\pi x_n)]\right\} + \sum_{i=1}^{n} u(x_i, 5,100,4)$	30	$[-50,50]$	$[-100,\cdots,-100]$	0

Table A3. Fixed-dimension multimodal benchmark functions.

Function	Dim	Range	Shift position	f_{min}
$F_{14}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2}(x_i - a_{ij})^6}\right)^{-1}$	2	$[-65,65]$	$[-2,-2,\cdots,-2]$	1
$F_{15}(x) = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2$	4	$[-5,5]$	$[-2,-2,\cdots,-2]$	0.0003075
$F_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	2	$[-5,5]$	$[-2,-2,\cdots,-2]$	-1.0316285
$F_{17}(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right) + 10\left(1 - \frac{1}{8\pi}\right)cos\, x_1 + 10$	2	$[-5,5]$	$[-2,-2,\cdots,-2]$	0.398
$F_{18}(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 16x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	$[-2,2]$	$[-2,-2,\cdots,-2]$	3

$F_{19}(x) = -\sum_{i=1}^{4} c_i exp\left(-\sum_{j=1}^{3} a_{ij}\left(x_j - p_{ij}\right)^2\right)$	3	[1,3]	$[-2,-2,\cdots,-2]$	-3.86
$F_{20}(x) = -\sum_{i=1}^{4} c_i exp\left(-\sum_{j=1}^{6} a_{ij}\left(x_j - p_{ij}\right)^2\right)$	6	[0,1]	$[-2,-2,\cdots,-2]$	-3.32
$F_{21}(x) = -\sum_{i=1}^{5}[(X-a_i)(X-a_i)^T + c_i]^{-1}$	4	[0,10]	$[-2,-2,\cdots,-2]$	-10.1532
$F_{22}(x) = -\sum_{i=1}^{7}[(X-a_i)(X-a_i)^T + c_i]^{-1}$	4	[0,10]	$[-2,-2,\cdots,-2]$	-10.4028
$F_{23}(x) = -\sum_{i=1}^{1}[(X-a_i)(X-a_i)^T + c_i]^{-1}$	4	[0,10]	$[-2,-2,\cdots,-2]$	-10.536

Table A4. CEC 2022 benchmark functions

		Functions	f_i
Unimodal Functions	F1	Shifted and full Rotated Zakharov Function	300
Multimodal Functions	F2	Shifted and full Rotated Rosenbrock's Function	400
	F3	Shifted and full Rotated Expanded Schaffer's f6 Function	600
	F4	Shifted and full Rotated Non-Continuous Rastrigin's Function	800
	F5	Shifted and full Rotated Levy Function	900
Hybrid	F6	Hybrid Function 1 (N = 3)	1800
	F7	Hybrid Function 2 (N = 6)	2000
	F8	Hybrid Function 3 (N = 5)	2200
Composition Functions	F9	Composition Function 1 (N = 5)	2300
	F10	Composition Function 2 (N = 4)	2400
	F11	Composition Function 3 (N = 5)	2600
	F12	Composition Function 4 (N = 6)	2700

3.1 Comparison of different algorithms on test functions

To assess and compare the search capabilities of the PO algorithm, 8 algorithms were chosen for a comparative analysis of classical test functions and IEEE CEC 2022 test functions. To ensure a fair comparison, all examined algorithms were run with uniform parameters on the classical test sets, comprising 1000 iterations and a population size of 30, mirroring PO's settings. In the IEEE CEC 2022 test sets, the same parameters were maintained for all examined algorithms, involving 30000 iterations and a population size of 30, reflecting PO's configurations.

This approach allowed for the evaluation of the relative performance and efficiency of PO in comparison to the selected algorithms, all under consistent experimental conditions. Applying this standardized evaluation framework, the search capabilities of the algorithms were effectively compared and analyzed on a fair basis. The classical test functions were first used to compare different algorithms, and then the results of the CEC 2022 test functions were compared and analyzed.

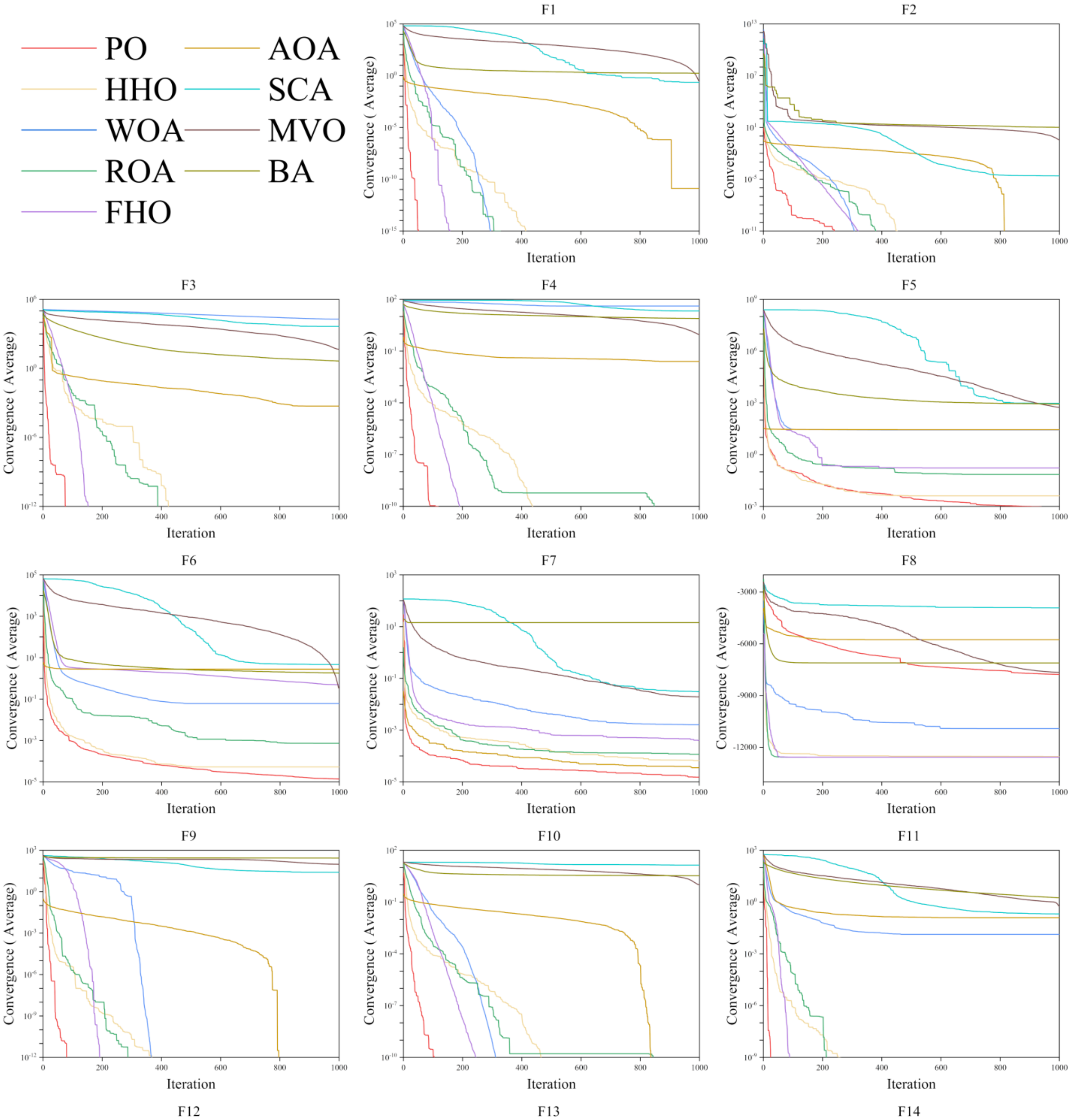


Fig. 7. Comparison of convergence for different algorithms (classical test sets)

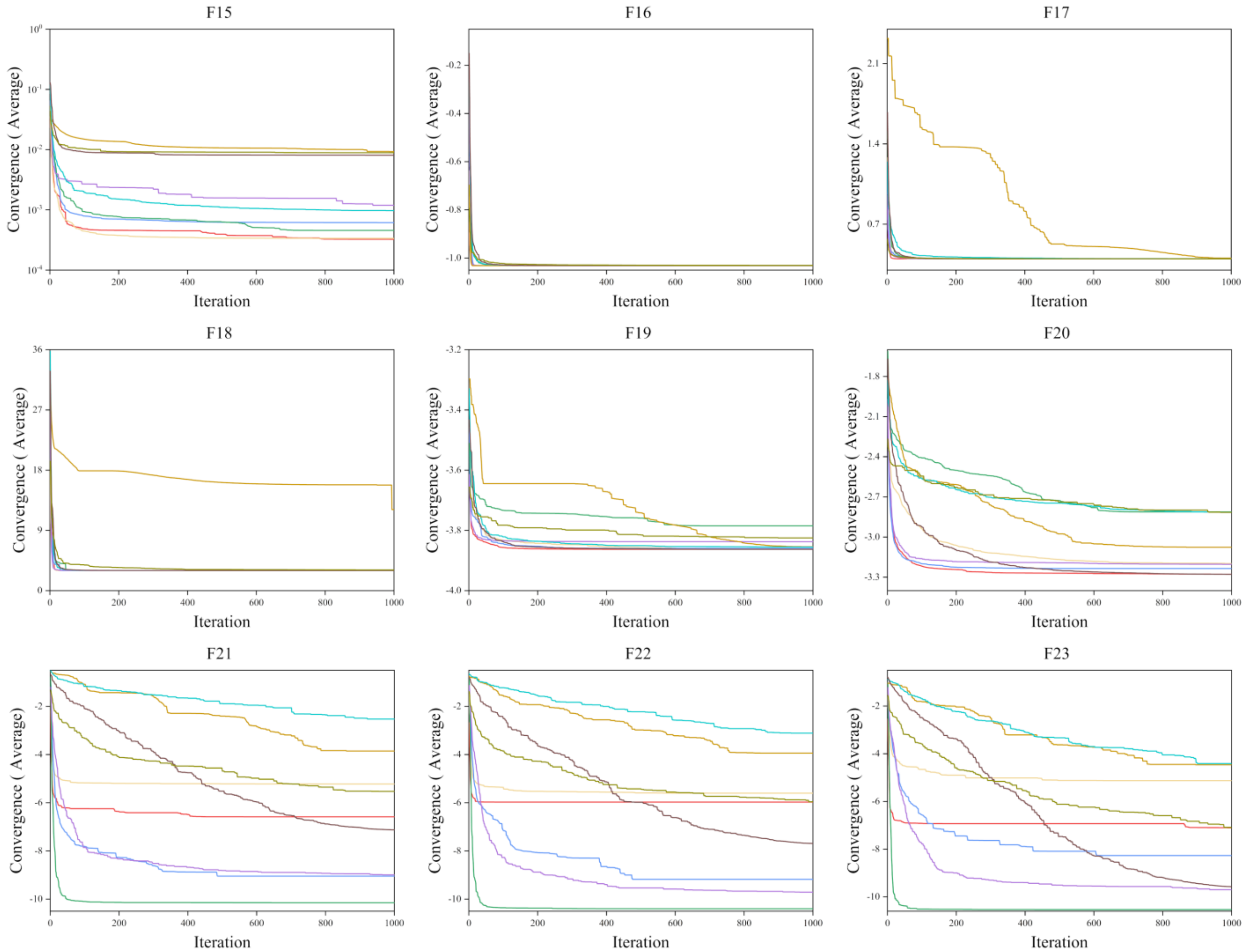


Fig. 7. (Continued)

Table 2. Results of unimodal classical test functions (different algorithms)

Function	Item	PO	HHO	WOA	ROA	FHO	AOA	SCA	MVO	BA
F1	Mean	1.06E-41	**3.14E-187**	3.55E-151	7.25E-19	7.04E-153	1.29E-11	2.24E-01	3.07E-01	1.71E+00
	Best	1.61E-143	1.17E-213	1.13E-168	3.97E-48	6.30E-181	**1.96E-285**	2.94E-08	2.03E-01	3.68E-01
	Worst	3.05E-40	**9.43E-186**	1.03E-149	2.07E-17	2.11E-151	3.88E-10	6.41E+00	5.02E-01	3.36E+00
	Median	1.19E-61	**4.40E-195**	5.75E-158	2.34E-29	4.95E-167	2.49E-153	9.44E-04	2.96E-01	1.72E+00
	STD	3.10E-81	**0.00E+00**	3.51E-300	1.43E-35	1.49E-303	5.01E-21	1.36E+00	6.30E-03	6.66E-01
F2	Mean	3.81E-18	3.05E-99	2.61E-103	6.39E-12	9.21E-40	**0.00E+00**	2.46E-05	3.64E-01	1.03E+01
	Best	7.25E-48	3.26E-109	2.01E-114	1.17E-23	7.59E-44	**0.00E+00**	7.65E-08	2.24E-01	4.61E+00
	Worst	1.14E-16	4.13E-98	7.60E-102	1.35E-10	2.55E-38	**0.00E+00**	2.04E-04	6.14E-01	2.16E+01
	Median	1.82E-25	9.31E-102	1.66E-108	1.19E-16	2.27E-41	**0.00E+00**	3.31E-06	3.50E-01	9.92E+00
	STD	4.30E-34	7.21E-197	1.92E-204	6.13E-22	2.15E-77	**0.00E+00**	2.39E-09	1.20E-02	1.85E+01
F3	Mean	1.88E-33	7.13E-138	1.93E+04	2.64E-13	**3.28E-157**	5.27E-04	4.45E+03	4.62E+01	4.34E+00
	Best	1.34E-69	2.23E-185	1.69E+03	4.58E-37	1.86E-173	1.55E-249	6.70E+01	2.28E+01	1.52E+00
	Worst	3.48E-32	2.14E-136	4.65E+04	7.91E-12	**7.38E-156**	1.21E-02	1.63E+04	9.18E+01	9.59E+00
	Median	9.19E-41	**3.91E-166**	1.82E+04	3.56E-26	2.67E-164	1.04E-124	3.15E+03	3.96E+01	3.58E+00
	STD	5.23E-65	**1.53E-273**	1.39E+08	2.09E-24	1.97E+00	5.23E-06	1.82E+07	3.97E+02	5.99E+00
F4	Mean	6.19E-25	**5.74E-94**	4.07E+01	3.82E-11	1.84E-62	2.51E-02	2.12E+01	9.19E-01	7.78E+00
	Best	**0.00E+00**	2.55E-103	1.68E-01	1.46E-21	4.06E-71	3.03E-107	2.41E+00	5.24E-01	7.96E-01
	Worst	1.10E-23	**9.96E-93**	8.73E+01	8.27E-10	5.50E-61	4.37E-02	4.70E+01	2.03E+00	2.12E+01
	Median	1.59E-35	**2.14E-97**	4.12E+01	3.10E-15	6.87E-67	3.96E-02	1.74E+01	8.47E-01	6.28E+00
	STD	5.43E-48	**3.71E-186**	9.87E+02	2.53E-20	1.01E-122	3.93E-04	1.36E+02	1.11E-01	2.80E+01
F5	Mean	**8.33E-04**	4.10E-03	2.71E+01	7.12E-02	1.68E-01	2.82E+01	9.49E+02	5.57E+02	8.44E+02
	Best	**4.86E-08**	3.89E-05	2.62E+01	1.67E-04	4.24E-03	2.62E+01	2.82E+01	2.91E+01	5.88E+01
	Worst	**1.18E-02**	3.62E-02	2.79E+01	5.24E-01	4.89E-01	2.87E+01	1.59E+04	2.49E+03	2.57E+03
	Median	**2.40E-04**	1.47E-03	2.70E+01	3.22E-02	1.45E-01	2.83E+01	8.91E+01	1.90E+02	7.30E+02
	STD	**4.66E-06**	5.61E-05	1.96E-01	1.51E-02	1.26E-02	2.82E-01	8.76E+06	5.60E+05	4.91E+05
F6	Mean	**1.38E-05**	5.27E-05	6.06E-02	7.30E-04	4.91E-01	2.78E+00	4.68E+00	3.26E-01	1.80E+00
	Best	**2.31E-08**	4.50E-08	9.81E-03	3.03E-07	5.63E-03	2.15E+00	3.65E+00	1.32E-01	4.00E-01
	Worst	**5.81E-05**	3.27E-04	4.02E-01	5.33E-03	4.39E+00	3.25E+00	6.73E+00	6.05E-01	4.33E+00
	Median	**7.52E-06**	1.58E-05	2.68E-02	2.02E-04	2.16E-02	2.82E+00	4.63E+00	3.18E-01	1.63E+00
	STD	**2.74E-10**	6.38E-09	6.48E-03	1.49E-06	1.30E+00	7.26E-02	5.03E-01	1.15E-02	9.66E-01
F7	Mean	**1.52E-05**	6.63E-05	1.62E-03	1.17E-04	3.98E-04	3.52E-05	3.06E-02	1.93E-02	1.43E+01
	Best	1.03E-06	**6.47E-07**	4.66E-06	4.34E-06	1.28E-05	7.43E-07	2.58E-03	6.80E-03	3.95E+00
	Worst	**5.98E-05**	2.91E-04	8.91E-03	7.07E-04	9.97E-04	1.38E-04	1.59E-01	3.38E-02	3.84E+01
	Median	**1.10E-05**	4.74E-05	8.29E-04	6.62E-05	3.16E-04	2.21E-05	2.03E-02	2.00E-02	1.20E+01
	STD	**2.20E-10**	3.65E-09	4.67E-06	2.09E-08	7.80E-08	1.20E-09	8.88E-04	4.69E-05	7.64E+01

Table 3. Result of multimodal classical test functions (different algorithms)

Function	Item	PO	HHO	WOA	ROA	FHO	AOA	SCA	MVO	BA
F8	Mean	-7.77E+03	-1.25E+04	-1.09E+04	**-1.26E+04**	-1.26E+04	-5.76E+03	-3.91E+03	-7.65E+03	-7.11E+03
	Best	**-1.26E+04**	**-1.26E+04**	**-1.26E+04**	**-1.26E+04**	**-1.26E+04**	-6.37E+03	-4.67E+03	-9.35E+03	-8.34E+03
	Worst	-5.24E+03	-1.13E+04	-6.79E+03	**-1.26E+04**	-1.26E+04	-5.22E+03	-3.49E+03	-6.03E+03	-6.15E+03
	Median	-7.24E+03	-1.26E+04	-1.18E+04	**-1.26E+04**	-1.26E+04	-5.74E+03	-3.86E+03	-7.56E+03	-7.06E+03
	STD	5.64E+05	5.39E+04	3.33E+06	**5.51E-05**	1.85E-02	1.10E+05	6.78E+04	6.25E+05	4.72E+05
F9	Mean	**0.00E+00**	**0.00E+00**	1.89E-15	**0.00E+00**	**0.00E+00**	**0.00E+00**	2.56E+01	9.81E+01	2.74E+02
	Best	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	7.41E-06	5.49E+01	2.27E+02
	Worst	**0.00E+00**	**0.00E+00**	5.68E-14	**0.00E+00**	**0.00E+00**	**0.00E+00**	1.08E+02	1.49E+02	3.50E+02
	Median	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	1.71E+01	9.67E+01	2.69E+02
	STD	**0.00E+00**	**0.00E+00**	1.08E-28	**0.00E+00**	**0.00E+00**	**0.00E+00**	7.78E+02	5.37E+02	7.56E+02
F10	Mean	**8.88E-16**	**8.88E-16**	3.97E-15	6.28E-11	**8.88E-16**	**8.88E-16**	1.39E+01	1.01E+00	3.36E+00
	Best	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**	**8.88E-16**	2.20E-03	1.30E-01	2.02E+00
	Worst	**8.88E-16**	**8.88E-16**	7.99E-15	1.63E-09	**8.88E-16**	**8.88E-16**	2.03E+01	2.56E+00	4.33E+00
	Median	**8.88E-16**	**8.88E-16**	4.44E-15	**8.88E-16**	**8.88E-16**	**8.88E-16**	1.97E+01	9.94E-01	3.35E+00
	STD	**0.00E+00**	**0.00E+00**	4.12E-30	8.86E-20	**0.00E+00**	**0.00E+00**	7.64E+01	6.06E-01	2.47E-01
F11	Mean	**0.00E+00**	**0.00E+00**	1.34E-02	7.40E-18	**0.00E+00**	1.22E-01	2.05E-01	5.81E-01	1.78E+00
	Best	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	7.75E-03	2.89E-06	4.08E-01	6.69E-04
	Worst	**0.00E+00**	**0.00E+00**	7.54E-02	2.22E-16	**0.00E+00**	3.98E-01	6.58E-01	7.44E-01	2.49E+01
	Median	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	8.52E-02	1.84E-01	5.87E-01	1.63E-02
	STD	**0.00E+00**	**0.00E+00**	6.59E-04	1.64E-33	**0.00E+00**	1.12E-02	4.14E-02	6.35E-03	2.38E+01
F12	Mean	**7.38E-07**	3.00E-06	7.59E-03	9.17E-06	7.84E-04	4.08E-01	2.41E+02	1.53E+00	1.25E+01
	Best	**1.17E-12**	7.20E-11	1.22E-03	2.84E-09	3.92E-04	2.84E-01	4.69E-01	2.90E-03	6.14E+00
	Worst	**6.13E-06**	3.02E-05	3.19E-02	1.86E-04	1.73E-03	5.02E-01	7.18E+03	3.67E+00	2.16E+01
	Median	**2.39E-07**	7.81E-07	3.26E-03	7.10E-07	7.69E-04	4.16E-01	9.63E-01	1.32E+00	1.21E+01
	STD	**1.39E-12**	4.57E-11	7.61E-05	1.15E-09	9.49E-08	2.11E-03	1.72E+06	1.28E+00	1.47E+01
F13	Mean	**4.38E-06**	1.99E-05	2.49E-01	5.38E-04	6.24E-03	2.75E+00	5.22E+03	5.94E-02	4.15E-01
	Best	**9.15E-13**	8.47E-08	3.59E-02	1.15E-06	1.63E-03	2.47E+00	2.18E+00	1.93E-02	1.76E-01
	Worst	**2.89E-05**	8.85E-05	7.07E-01	5.59E-03	1.33E-02	2.99E+00	1.33E+05	1.31E-01	8.96E-01
	Median	**1.60E-06**	1.17E-05	2.36E-01	1.71E-04	5.62E-03	2.74E+00	3.11E+00	5.80E-02	3.82E-01
	STD	**5.76E-11**	5.77E-10	3.09E-02	1.10E-06	8.71E-06	1.99E-02	5.94E+08	6.80E-04	3.20E-02

Table 4. Result of fixed-dimension multimodal classical test functions (different algorithms)

Function	Item	PO	HHO	WOA	ROA	FHO	AOA	SCA	MVO	BA
F14	Mean	3.56E+00	1.06E+00	2.67E+00	9.98E-01	1.57E+00	8.53E+00	1.92E+00	9.98E-01	3.78E+00
	Best	9.98E-01	9.98E-01	9.98E-01	9.98E-01	9.98E-01	9.98E-01	9.98E-01	9.98E-01	9.98E-01
	Worst	1.27E+01	1.99E+00	1.08E+01	9.98E-01	4.69E+00	1.27E+01	2.98E+00	9.98E-01	1.17E+01
	Median	9.98E-01	9.98E-01	9.98E-01	9.98E-01	1.09E+00	1.08E+01	9.98E-01	9.98E-01	2.98E+00
	STD	2.10E+01	6.36E-02	8.90E+00	1.11E-19	6.92E-01	1.97E+01	1.01E+00	1.04E-22	1.08E+01
F15	Mean	3.25E-04	3.36E-04	6.13E-04	4.55E-04	1.19E-03	9.28E-03	9.70E-04	8.07E-03	8.86E-03
	Best	3.08E-04	3.08E-04	3.08E-04	3.09E-04	3.77E-04	3.27E-04	4.07E-04	3.20E-04	7.57E-04
	Worst	7.84E-04	4.04E-04	1.51E-03	1.75E-03	6.09E-03	8.67E-02	1.53E-03	6.32E-02	5.89E-02
	Median	3.08E-04	3.26E-04	4.93E-04	3.44E-04	6.51E-04	2.92E-03	8.08E-04	7.64E-04	1.40E-03
	STD	7.52E-09	7.60E-10	1.10E-07	1.05E-07	2.07E-06	2.75E-04	1.27E-07	1.85E-04	1.68E-04
F16	Mean	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00
	Best	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00
	Worst	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00

	Median	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00
	STD	5.74E-21	4.78E-22	3.02E-20	3.03E-10	7.35E-11	5.98E-15	3.38E-10	1.00E-14	1.05E-06
F17	Mean	3.98E-01	3.98E-01	3.98E-01	3.98E-01	3.98E-01	4.04E-01	3.99E-01	3.98E-01	3.98E-01
	Best	3.98E-01	3.98E-01	3.98E-01	3.98E-01	3.98E-01	3.98E-01	3.98E-01	3.98E-01	3.98E-01
	Worst	3.98E-01	3.98E-01	3.98E-01	4.00E-01	3.98E-01	4.30E-01	4.02E-01	3.98E-01	4.00E-01
	Median	3.98E-01	3.98E-01	3.98E-01	3.98E-01	3.98E-01	4.02E-01	3.99E-01	3.98E-01	3.98E-01
	STD	9.61E-20	9.18E-13	2.80E-12	1.19E-07	2.78E-08	4.45E-05	9.18E-07	2.95E-14	1.97E-07
F18	Mean	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	1.21E+01	3.00E+00	3.00E+00	3.09E+00
	Best	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.01E+00
	Worst	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.01E+00	3.53E+01	3.00E+00	3.00E+00	3.40E+00
	Median	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.00E+00	3.07E+00
	STD	3.70E-19	1.25E-16	1.04E-09	3.31E-07	1.74E-06	1.73E+02	7.95E-09	1.07E-12	8.64E-03
F19	Mean	-3.86E+00	-3.86E+00	-3.86E+00	-3.78E+00	-3.84E+00	-3.85E+00	-3.86E+00	-3.86E+00	-3.83E+00
	Best	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00
	Worst	-3.86E+00	-3.86E+00	-3.84E+00	-3.60E+00	-3.67E+00	-3.84E+00	-3.85E+00	-3.86E+00	-3.76E+00
	Median	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00	-3.85E+00	-3.85E+00	-3.86E+00	-3.84E+00
	STD	2.11E-10	2.92E-06	2.42E-05	8.49E-03	2.88E-03	1.22E-05	6.94E-06	6.89E-13	8.41E-04
F20	Mean	-3.28E+00	-3.20E+00	-3.23E+00	-2.81E+00	-3.20E+00	-3.08E+00	-2.81E+00	-3.28E+00	-2.82E+00
	Best	-3.32E+00	-3.31E+00	-3.32E+00	-3.19E+00	-3.29E+00	-3.18E+00	-3.22E+00	-3.32E+00	-3.10E+00
	Worst	-3.11E+00	-3.01E+00	-3.04E+00	-1.50E+00	-3.00E+00	-2.83E+00	-1.57E+00	-3.20E+00	-2.58E+00
	Median	-3.32E+00	-3.23E+00	-3.20E+00	-2.96E+00	-3.25E+00	-3.11E+00	-3.01E+00	-3.32E+00	-2.79E+00
	STD	5.04E-03	1.05E-02	8.37E-03	2.07E-01	7.63E-03	7.78E-03	2.40E-01	3.42E-03	2.04E-02
F21	Mean	-6.58E+00	-5.22E+00	-9.05E+00	-1.02E+01	-8.99E+00	-3.85E+00	-2.53E+00	-7.12E+00	-5.53E+00
	Best	-1.02E+01	-1.01E+01	-1.02E+01	-1.02E+01	-1.00E+01	-9.44E+00	-8.96E+00	-1.02E+01	-9.65E+00
	Worst	-5.06E+00	-5.05E+00	-2.63E+00	-1.01E+01	-4.91E+00	-1.75E+00	-3.51E-01	-2.63E+00	-2.36E+00
	Median	-5.06E+00	-5.05E+00	-1.02E+01	-1.02E+01	-9.46E+00	-3.63E+00	-8.82E-01	-5.10E+00	-4.65E+00
	STD	5.77E+00	8.39E-01	5.17E+00	5.54E-06	1.96E+00	2.32E+00	5.07E+00	8.92E+00	5.80E+00
F22	Mean	-5.97E+00	-5.60E+00	-9.18E+00	-1.04E+01	-9.71E+00	-3.94E+00	-3.11E+00	-7.69E+00	-5.96E+00
	Best	-1.04E+01	-1.04E+01	-1.04E+01	-1.04E+01	-1.03E+01	-7.41E+00	-6.06E+00	-1.04E+01	-1.00E+01
	Worst	-5.09E+00	-5.08E+00	-2.77E+00	-1.04E+01	-8.83E+00	-1.43E+00	-5.21E-01	-1.84E+00	-2.29E+00
	Median	-5.09E+00	-5.09E+00	-1.04E+01	-1.04E+01	-9.75E+00	-3.89E+00	-3.53E+00	-1.04E+01	-6.53E+00
	STD	4.06E+00	2.47E+00	6.37E+00	1.51E-05	1.16E-01	1.77E+00	3.51E+00	1.07E+01	6.16E+00
F23	Mean	-7.11E+00	-5.13E+00	-8.27E+00	-1.05E+01	-9.70E+00	-4.46E+00	-4.41E+00	-9.57E+00	-7.10E+00
	Best	-1.05E+01	-5.13E+00	-1.05E+01	-1.05E+01	-1.04E+01	-9.66E+00	-8.04E+00	-1.05E+01	-1.04E+01
	Worst	-5.13E+00	-5.10E+00	-1.68E+00	-1.05E+01	-9.01E+00	-1.88E+00	-9.43E-01	-2.43E+00	-2.59E+00
	Median	-5.13E+00	-5.13E+00	-1.05E+01	-1.05E+01	-9.69E+00	-4.32E+00	-4.89E+00	-1.05E+01	-7.75E+00
	STD	6.76E+00	2.91E-05	9.80E+00	2.38E-06	1.53E-01	3.60E+00	2.85E+00	6.43E+00	4.03E+00

Table 5. Rank of different algorithms (classical test function)

	PO	HHO	WOA	ROA	FHO	AOA	SCA	MVO	BA
F1	4	1	3	5	2	6	7	8	9
F2	5	3	2	6	4	1	7	8	9
F3	3	2	9	4	1	5	8	7	6
F4	3	1	9	4	2	5	8	6	7
F5	1	2	5	3	4	6	9	7	8
F6	1	2	4	3	6	8	9	5	7
F7	1	3	6	4	5	2	8	7	9
F8	5	3	4	2	1	8	9	6	7
F9	1	4	6	3	2	5	7	8	9
F10	1	3	5	6	2	4	9	7	8
F11	1	3	5	4	2	6	7	8	9

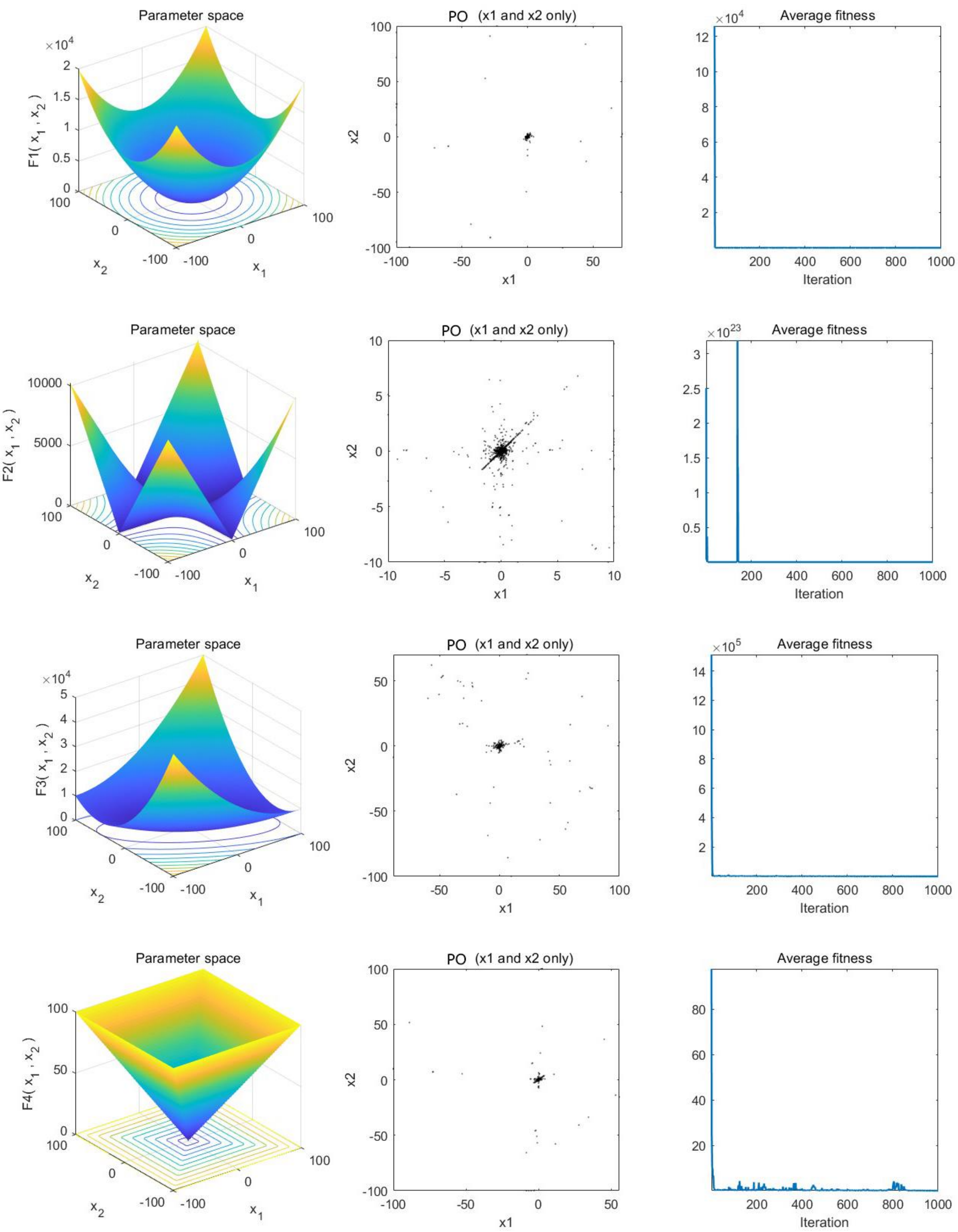
F12	1	2	5	3	4	6	9	7	8
F13	1	2	6	3	4	8	9	5	7
F14	7	3	6	2	4	9	5	1	8
F15	1	2	4	3	6	9	5	7	8
F16	3	4	2	6	5	1	8	7	9
F17	1	3	2	5	6	9	8	4	7
F18	1	2	4	6	7	9	5	3	8
F19	1	3	4	9	7	6	5	2	8
F20	2	5	3	9	4	6	8	1	7
F21	5	7	2	1	3	8	9	4	6
F22	5	7	3	1	2	8	9	4	6
F23	5	7	4	1	2	8	9	3	6
Average Rank	2.57	3.22	4.48	4.04	3.70	6.22	7.70	5.43	7.65
Final Ranking	1	2	5	4	3	7	9	6	8

Table 2 - Table 4 comprehensively compares the search results achieved by the PO and 8 popular optimization algorithms across functions F1 - F23, utilizing five evaluation metrics. The average convergence curves of these nine algorithms are depicted in Fig. 7. The results reveal that, in the majority of the tested functions, PO outperforms the other algorithms. PO attains a high ranking in most test functions, underscoring its strong search capabilities. Particularly in Fig. 7, functions F5, F6, F9, F10, F11, and F16, it becomes evident that PO excels in search capability and rapid convergence, enabling it to identify optimal or near-optimal solutions swiftly. Furthermore, PO exhibits superior convergence ability in most test functions, further establishing PO as a comprehensive optimization algorithm.

The performance ranking of the PO algorithm and eight other leading algorithms across functions F1 to F23 is presented in Table 5. It is worth emphasizing that in ranking these nine algorithms, the criteria are prioritized in order of mean. The results unequivocally establish the PO algorithm's superiority over the other eight optimization algorithms, securing the top position with an average ranking of 2.57. This performance repeats the PO algorithm's ability in addressing the considered optimization problems.

This comprehensive evaluation underscores the effectiveness and competitiveness of the PO in comparison to other well-established optimization algorithms, validating its healthy performance across various function types and its suitability for tackling complicated optimization challenges.

To investigate the performance of the developed PO algorithm further, we conducted experiments and recorded convergence and trajectory data. Fig. 8 visually depicts these results, showcasing the history of search points within the population and the corresponding fluctuations in the average fitness of the population during the PO algorithm's quest for an optimal solution. It is evident that PO exhibits strong convergence and stability across most of the tested functions. The average convergence value shift reveals that convergence is nearly instantaneous on unimodal functions F1 to F6. Conversely, convergence transpires more gradually on the multimodal functions, such as F8 to F10. It's worth noting that on the fixed-dimension multimodal functions, the convergence behavior varies; it can be instantaneous (F15, F16), gradual (F18, F19), or less stable (F21 to F23), though it continues to pursue optimal or near-optimal solutions.



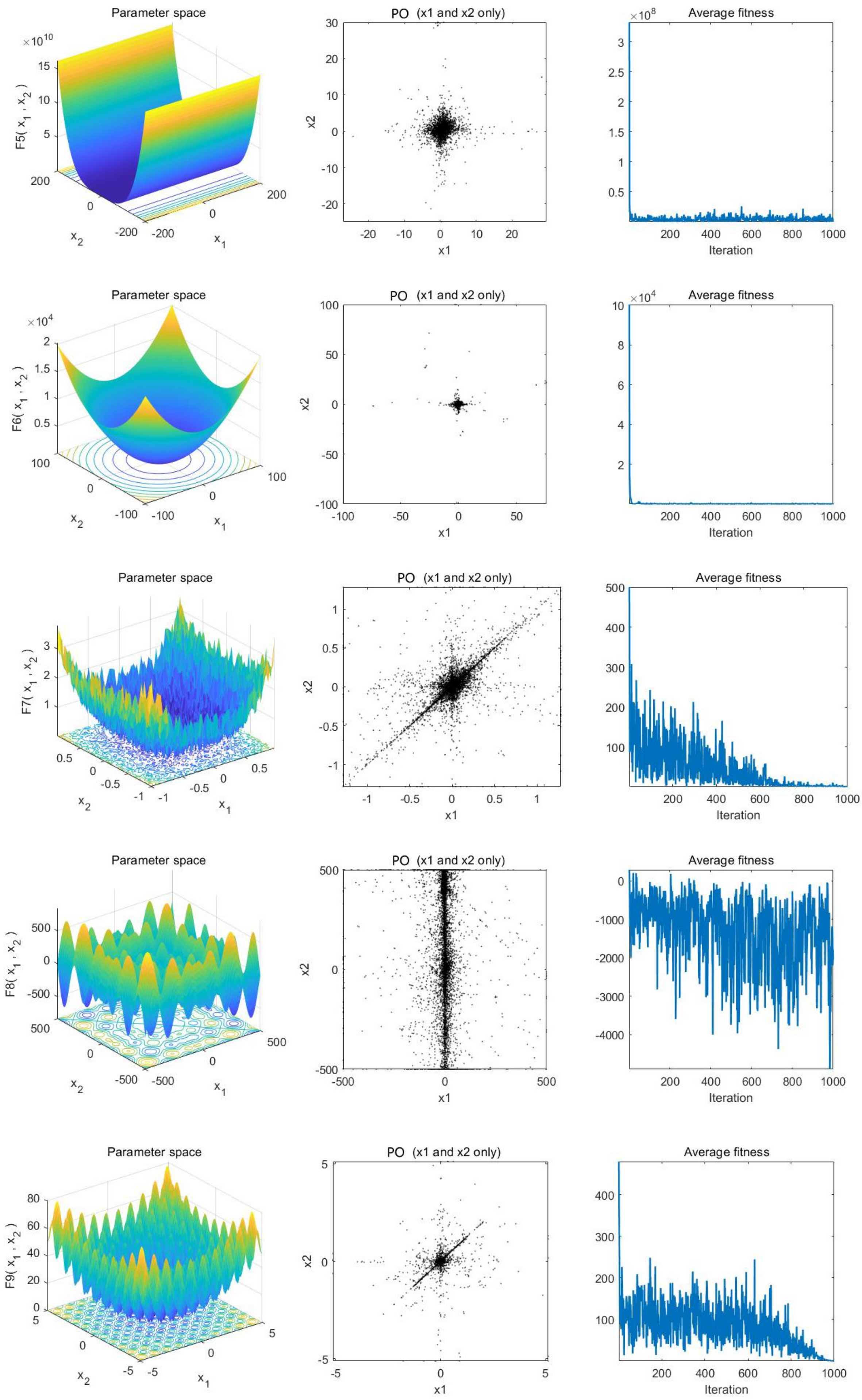


Fig. 8. Qualitative results for the classical functions

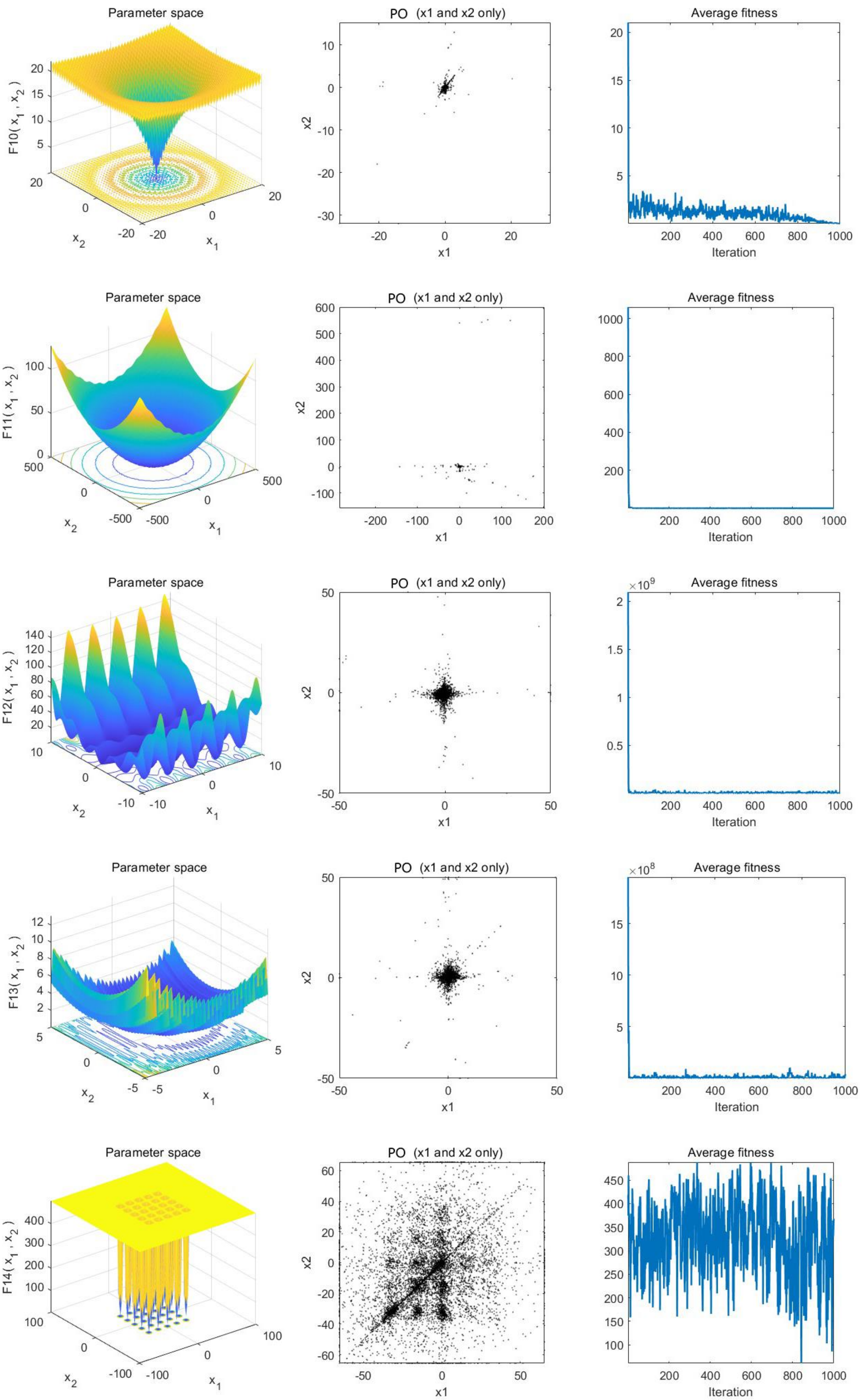


Fig. 8. (Continued)

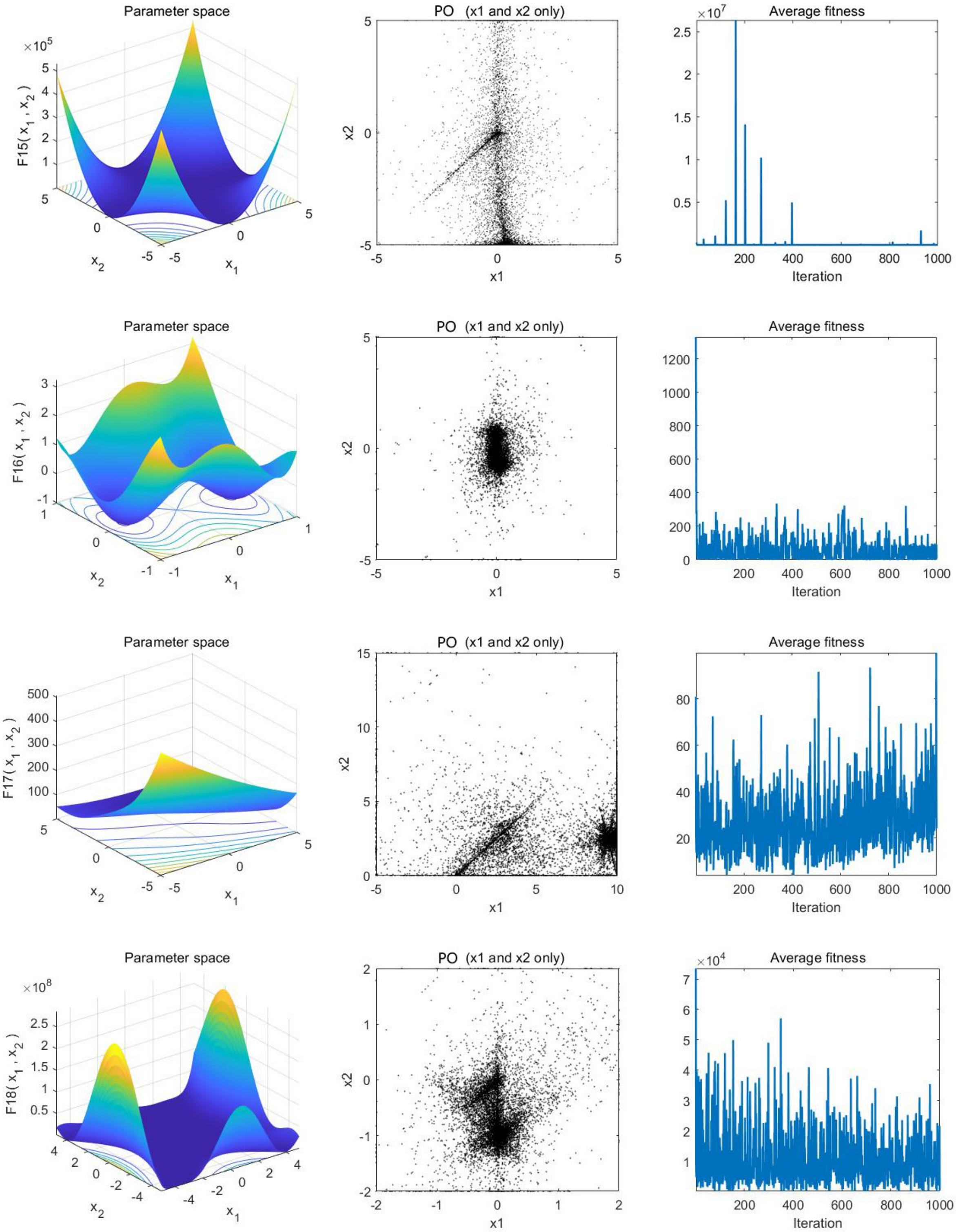


Fig. 8. (Continued)

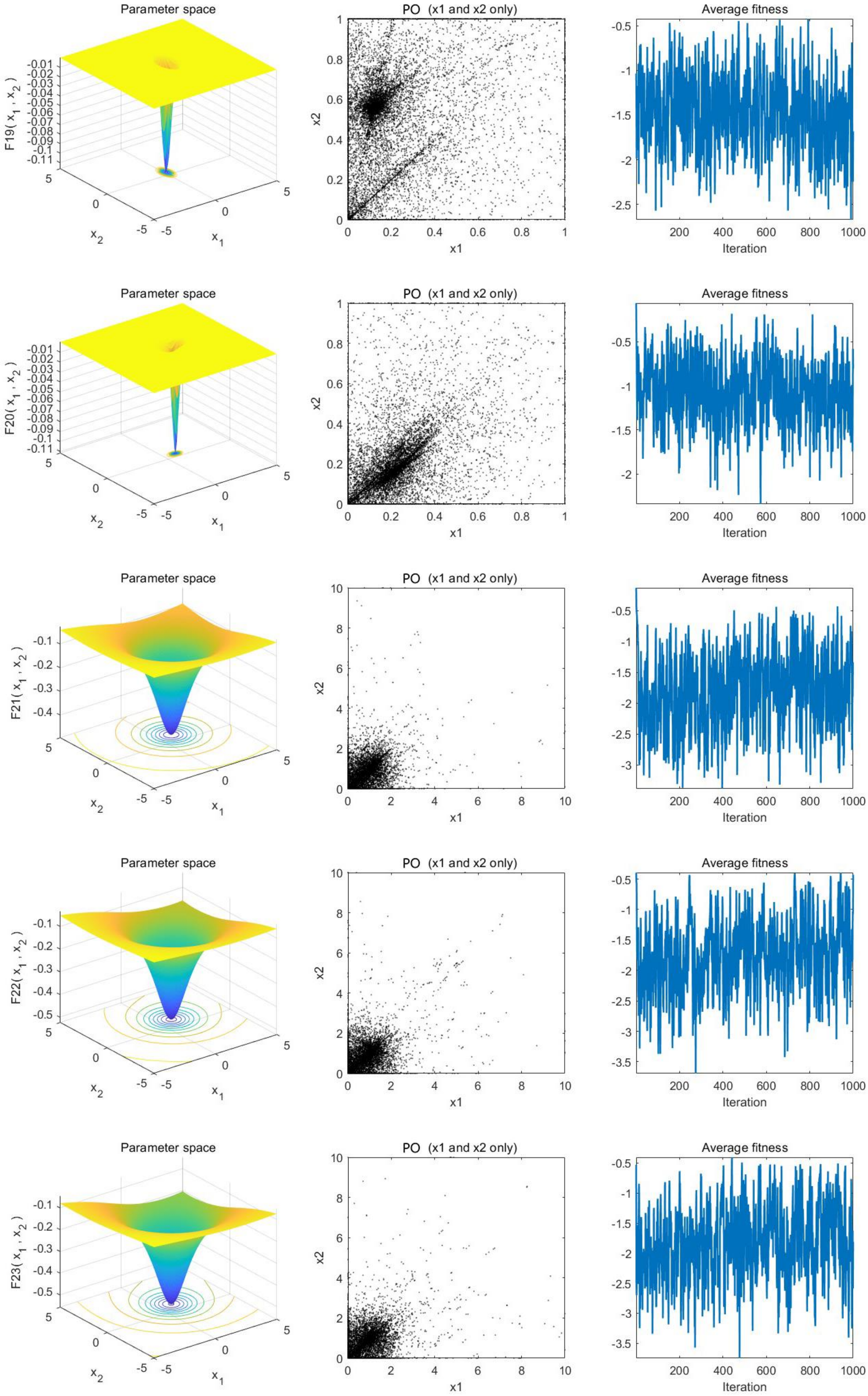


Fig. 8. (Continued)

To delve deeper into evaluating the effectiveness of the proposed PO algorithm and to scrutinize its ability to balance exploration, exploitation, and avoiding local optima, we subjected it to one of the most challenging benchmarks available, namely, the CEC 2022 test functions. We rigorously tested PO using these benchmarks and conducted a comparative analysis against the eight selected optimization algorithms mentioned earlier. All algorithms underwent 30 independent runs, each consisting of 30,000 iterations and employing a population size of 30.

The average convergence curves of the ten algorithms are depicted in Fig. 9 for the twelve tested function species in 20 dimensions. Tables 6 and 7 present the search results and ranking of the ten search algorithms for dimensions (dim) 20. It is essential to emphasize that the criteria prioritize the mean value when ranking these nine algorithms.

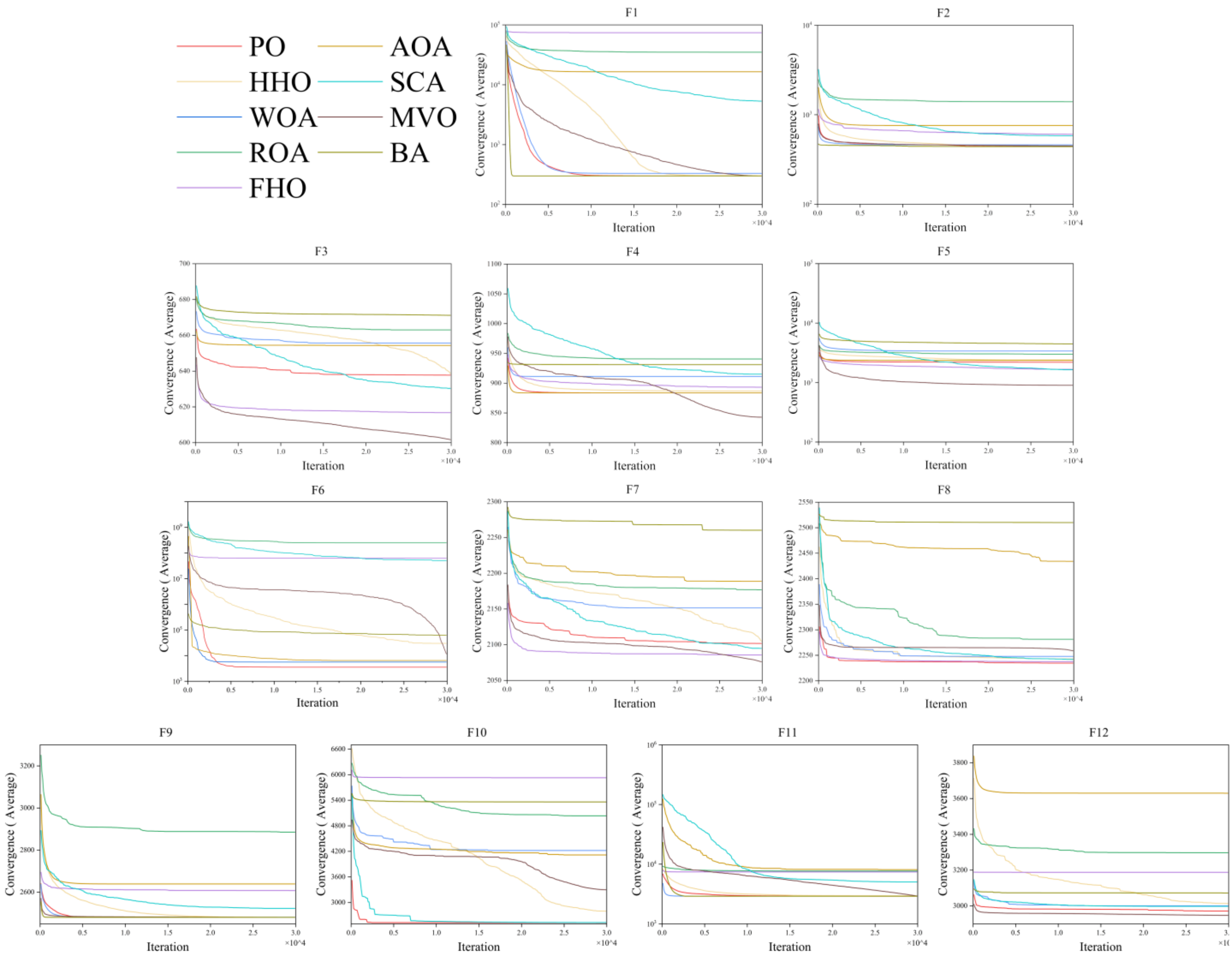


Fig. 9. Comparison of convergence for different algorithms (CEC 2022 test sets)

Table 6. Result of CEC 2022 test functions (different algorithms)

Function	Item	PO	HHO	WOA	ROA	FHO	AOA	SCA	MVO	BA
F1	Mean	3.00E+02	3.03E+02	3.31E+02	3.50E+04	7.41E+04	1.65E+04	5.34E+03	**3.00E+02**	3.00E+02
	Best	**3.00E+02**	3.02E+02	3.03E+02	2.00E+04	2.35E+04	9.02E+03	2.95E+03	3.00E+02	3.00E+02
	Worst	3.01E+02	3.06E+02	5.29E+02	6.42E+04	1.15E+05	2.51E+04	8.81E+03	**3.00E+02**	3.00E+02
	Median	3.00E+02	3.03E+02	3.16E+02	3.35E+04	7.67E+04	1.62E+04	5.01E+03	**3.00E+02**	3.00E+02
	STD	1.62E-01	5.83E-01	1.69E+03	1.26E+08	5.75E+08	2.14E+07	2.14E+06	**8.06E-05**	2.51E-03
F2	Mean	4.47E+02	4.44E+02	4.61E+02	1.40E+03	6.04E+02	7.58E+02	5.83E+02	4.48E+02	**4.40E+02**
	Best	4.05E+02	4.05E+02	4.05E+02	7.34E+02	5.17E+02	6.00E+02	5.40E+02	4.45E+02	**4.00E+02**

	Worst	4.74E+02	4.77E+02	4.75E+02	4.75E+03	7.26E+02	1.07E+03	6.73E+02	**4.68E+02**	4.76E+02
	Median	4.49E+02	4.49E+02	4.70E+02	1.22E+03	6.05E+02	7.23E+02	5.72E+02	**4.45E+02**	4.49E+02
	STD	4.70E+02	5.51E+02	2.47E+02	5.43E+05	3.24E+03	1.63E+04	1.19E+03	**4.60E+01**	5.66E+02
F3	Mean	6.38E+02	6.39E+02	6.56E+02	6.63E+02	6.17E+02	6.54E+02	6.30E+02	**6.02E+02**	6.71E+02
	Best	6.18E+02	6.22E+02	6.18E+02	6.43E+02	6.14E+02	6.40E+02	6.23E+02	**6.00E+02**	6.54E+02
	Worst	6.68E+02	6.56E+02	6.86E+02	6.85E+02	6.20E+02	6.70E+02	6.44E+02	**6.06E+02**	6.95E+02
	Median	6.35E+02	6.38E+02	6.57E+02	6.62E+02	6.17E+02	6.55E+02	6.31E+02	**6.01E+02**	6.69E+02
	STD	1.29E+02	8.63E+01	1.68E+02	1.48E+02	**1.93E+00**	4.44E+01	2.16E+01	3.13E+00	1.49E+02
F4	Mean	8.84E+02	8.87E+02	9.11E+02	9.41E+02	8.93E+02	8.84E+02	9.15E+02	**8.43E+02**	9.31E+02
	Best	8.65E+02	8.60E+02	8.59E+02	9.10E+02	8.76E+02	8.45E+02	8.97E+02	**8.17E+02**	8.61E+02
	Worst	9.19E+02	9.18E+02	9.98E+02	9.88E+02	9.26E+02	9.12E+02	9.48E+02	**8.94E+02**	1.01E+03
	Median	8.80E+02	8.90E+02	9.07E+02	9.38E+02	8.90E+02	8.87E+02	9.15E+02	**8.41E+02**	9.27E+02
	STD	1.66E+02	2.10E+02	1.03E+03	3.50E+02	1.35E+02	2.81E+02	**1.26E+02**	2.94E+02	1.31E+03
F5	Mean	2.19E+03	2.40E+03	3.39E+03	2.99E+03	1.67E+03	2.36E+03	1.64E+03	**9.00E+02**	4.47E+03
	Best	1.58E+03	1.77E+03	1.90E+03	1.80E+03	1.18E+03	1.65E+03	1.20E+03	**9.00E+02**	2.17E+03
	Worst	2.52E+03	2.71E+03	5.61E+03	3.95E+03	2.30E+03	2.93E+03	2.13E+03	**9.01E+02**	9.66E+03
	Median	2.24E+03	2.45E+03	3.10E+03	3.01E+03	1.66E+03	2.35E+03	1.58E+03	**9.00E+02**	4.04E+03
	STD	6.33E+04	4.89E+04	1.07E+06	2.26E+05	7.84E+04	9.39E+04	5.30E+04	**3.78E-02**	2.80E+06
F6	Mean	**3.53E+03**	2.99E+04	5.58E+03	2.49E+08	6.24E+07	6.45E+03	5.09E+07	1.17E+04	6.22E+04
	Best	2.10E+03	4.49E+03	**1.90E+03**	2.65E+06	1.75E+07	3.56E+03	1.03E+07	2.89E+03	1.59E+04
	Worst	**5.49E+03**	7.31E+04	1.90E+04	2.46E+09	1.61E+08	1.90E+04	1.13E+08	2.58E+04	1.08E+05
	Median	3.34E+03	2.93E+04	**3.19E+03**	9.37E+07	4.34E+07	5.24E+03	5.14E+07	8.27E+03	6.42E+04
	STD	**8.58E+05**	2.48E+08	2.12E+07	2.17E+17	1.74E+15	**1.02E+07**	6.53E+14	6.63E+07	4.98E+08
F7	Mean	2.10E+03	2.10E+03	2.15E+03	2.18E+03	2.09E+03	2.19E+03	2.09E+03	**2.08E+03**	2.26E+03
	Best	2.06E+03	2.05E+03	2.07E+03	2.10E+03	2.06E+03	2.12E+03	2.08E+03	**2.02E+03**	2.12E+03
	Worst	2.16E+03	2.19E+03	2.24E+03	2.31E+03	2.25E+03	2.49E+03	**2.12E+03**	2.27E+03	2.38E+03
	Median	**2.09E+03**	2.09E+03	2.15E+03	2.18E+03	2.07E+03	2.17E+03	2.09E+03	2.05E+03	2.26E+03
	STD	7.03E+02	1.59E+03	1.74E+03	2.01E+03	1.54E+03	5.97E+03	**1.16E+02**	3.61E+03	4.11E+03
F8	Mean	**2.23E+03**	2.24E+03	2.25E+03	2.28E+03	2.24E+03	2.43E+03	2.24E+03	2.26E+03	2.51E+03
	Best	**2.23E+03**	2.23E+03	2.23E+03	2.23E+03	2.23E+03	2.24E+03	2.24E+03	2.22E+03	2.25E+03
	Worst	2.25E+03	2.35E+03	2.37E+03	2.75E+03	**2.24E+03**	2.61E+03	2.25E+03	2.36E+03	2.78E+03
	Median	**2.23E+03**	2.23E+03	2.24E+03	2.25E+03	2.24E+03	2.45E+03	2.24E+03	2.24E+03	2.49E+03
	STD	3.98E+01	8.68E+02	5.98E+02	9.94E+03	**4.25E+00**	1.60E+04	9.16E+00	2.39E+03	1.88E+04
F9	Mean	**2.48E+03**	2.48E+03	2.48E+03	2.89E+03	2.61E+03	2.64E+03	2.52E+03	2.48E+03	**2.48E+03**
	Best	**2.48E+03**	2.48E+03	2.48E+03	2.63E+03	2.53E+03	2.53E+03	2.50E+03	2.48E+03	**2.48E+03**
	Worst	**2.48E+03**	2.48E+03	2.48E+03	3.75E+03	3.09E+03	2.71E+03	2.55E+03	2.48E+03	**2.48E+03**
	Median	**2.48E+03**	2.48E+03	2.48E+03	2.78E+03	2.58E+03	2.65E+03	2.52E+03	2.48E+03	**2.48E+03**
	STD	4.01E-05	8.25E-03	2.14E-01	6.60E+04	1.12E+04	1.98E+03	1.19E+02	2.73E-05	**6.91E-07**
F10	Mean	**2.53E+03**	2.80E+03	4.22E+03	5.04E+03	5.93E+03	4.12E+03	2.53E+03	3.30E+03	5.36E+03
	Best	**2.50E+03**	2.44E+03	2.50E+03	2.52E+03	2.50E+03	2.54E+03	2.51E+03	2.50E+03	4.45E+03
	Worst	**2.73E+03**	3.98E+03	5.59E+03	7.15E+03	7.58E+03	5.36E+03	2.76E+03	4.19E+03	6.85E+03
	Median	**2.50E+03**	2.68E+03	4.51E+03	5.44E+03	7.00E+03	4.23E+03	2.51E+03	3.32E+03	5.29E+03
	STD	**4.80E+03**	1.29E+05	1.05E+06	2.05E+06	3.71E+06	6.30E+05	3.67E+03	2.74E+05	3.55E+05
F11	Mean	**2.90E+03**	2.95E+03	2.90E+03	7.67E+03	7.42E+03	8.10E+03	5.02E+03	2.93E+03	2.90E+03
	Best	**2.60E+03**	2.63E+03	2.90E+03	5.17E+03	5.31E+03	6.29E+03	4.30E+03	2.92E+03	2.90E+03
	Worst	3.21E+03	3.27E+03	**2.91E+03**	9.37E+03	8.61E+03	1.08E+04	6.32E+03	2.93E+03	2.91E+03
	Median	2.91E+03	2.94E+03	2.90E+03	7.86E+03	7.48E+03	7.73E+03	4.89E+03	2.93E+03	**2.90E+03**
	STD	1.75E+04	1.47E+04	3.78E+00	1.08E+06	6.97E+05	1.42E+06	2.22E+05	4.82E+00	**2.58E+00**
F12	Mean	2.97E+03	3.01E+03	3.00E+03	3.30E+03	3.19E+03	3.63E+03	3.00E+03	**2.95E+03**	3.07E+03
	Best	2.94E+03	2.95E+03	2.95E+03	2.99E+03	3.01E+03	3.19E+03	2.97E+03	**2.93E+03**	2.95E+03
	Worst	3.04E+03	3.12E+03	3.10E+03	3.85E+03	3.57E+03	4.00E+03	3.02E+03	**2.99E+03**	3.47E+03
	Median	2.97E+03	3.01E+03	2.99E+03	3.20E+03	3.15E+03	3.61E+03	2.99E+03	**2.94E+03**	3.04E+03
	STD	7.27E+02	1.23E+03	1.32E+03	5.25E+04	2.01E+04	2.99E+04	1.71E+02	**1.38E+02**	1.05E+04

Table 7. Rank of different algorithms (CEC 2022)

Function	PO	HHO	WOA	ROA	FHO	AOA	SCA	MVO	BA
F1	3	4	5	8	9	7	6	1	2
F2	3	2	5	9	7	8	6	4	1
F3	4	5	7	8	2	6	3	1	9
F4	2	4	6	9	5	3	7	1	8
F5	4	6	8	7	3	5	2	1	9
F6	1	5	2	9	8	3	7	4	6
F7	4	5	6	7	2	8	3	1	9
F8	1	4	5	7	2	8	3	6	9
F9	2	4	5	9	7	8	6	3	1
F10	1	3	6	7	9	5	2	4	8
F11	1	5	3	8	7	9	6	4	2
F12	2	5	4	8	7	9	3	1	6
Average Rank	2.3	4.3	5.2	8	5.7	6.6	4.5	2.6	5.8
Final Ranking	1	3	5	9	6	8	4	2	7

The results clearly indicate PO's superior overall ranking when compared to the eight highly regarded algorithms. What's even more noteworthy is the remarkable stability of PO's performance. It excels not only on functions like F8, F10, and F11 but also maintains a consistent fourth place ranking on less favorable test functions such as F3, F5, and F7. This impressive stability can be attributed to the synergy of PO's four strategies, which effectively complement each other's strengths and weaknesses. Notably, we have conducted additional experiments for CEC2022 in 2 and 10 dimensions. However, we have included these supplementary results in the appendix due to space constraints.

3.2 Parameter sensitivity analysis

In the aforementioned comparative experiments, the default PO algorithm maintained an equal ratio of foraging (F), staying (S), communicating (C), and fear of strangers (O) behaviors, represented as 1:1:1:1. However, we hypothesize that altering the distribution of these behaviors may yield distinct optimization outcomes. To explore this hypothesis, we designed parameter sensitivity analysis experiments. Five variations of the PO algorithm with different behavior ratios were established. These include the original PO algorithm with a ratio of F:S:C:O = 1:1:1:1, the PO-F algorithm with F:S:C:O = 2:1:1:1, the PO-S algorithm with F:S:C:O = 1:2:1:1, the PO-C algorithm with F:S:C:O = 1:1:2:1, and the PO-O algorithm with F:S:C:O = 1:1:1:2. Maintaining the same number of iterations and population size as detailed in Section 3.1, these five PO algorithms with varying parameters were tested on both the classical test function set and the IEEE CEC 2022 test set to investigate parameter sensitivity.

The outcomes of the five PO algorithms on the classical test function set are detailed in Tables 8 to 10. Fig. 10 visually illustrates the convergence speed and search capabilities of these five algorithms across the set of 23 classical test functions. Table 11 ranks the five algorithms based on the mean value and convergence speed. The results clearly indicate that different ratios of the four states result in varying convergence speeds and search abilities among the algorithms. Among the five aforementioned algorithms, PO-O demonstrates the best overall performance and excels in the majority of the test functions. It exhibits not only strong search capabilities but also swift convergence speeds. Additionally, PO demonstrates the highest stability, consistently delivering respectable results. The findings underscore that PO can potentially yield different optimal parameter values (ratios) depending on the specific objective functions, thereby enabling precise and efficient searches.

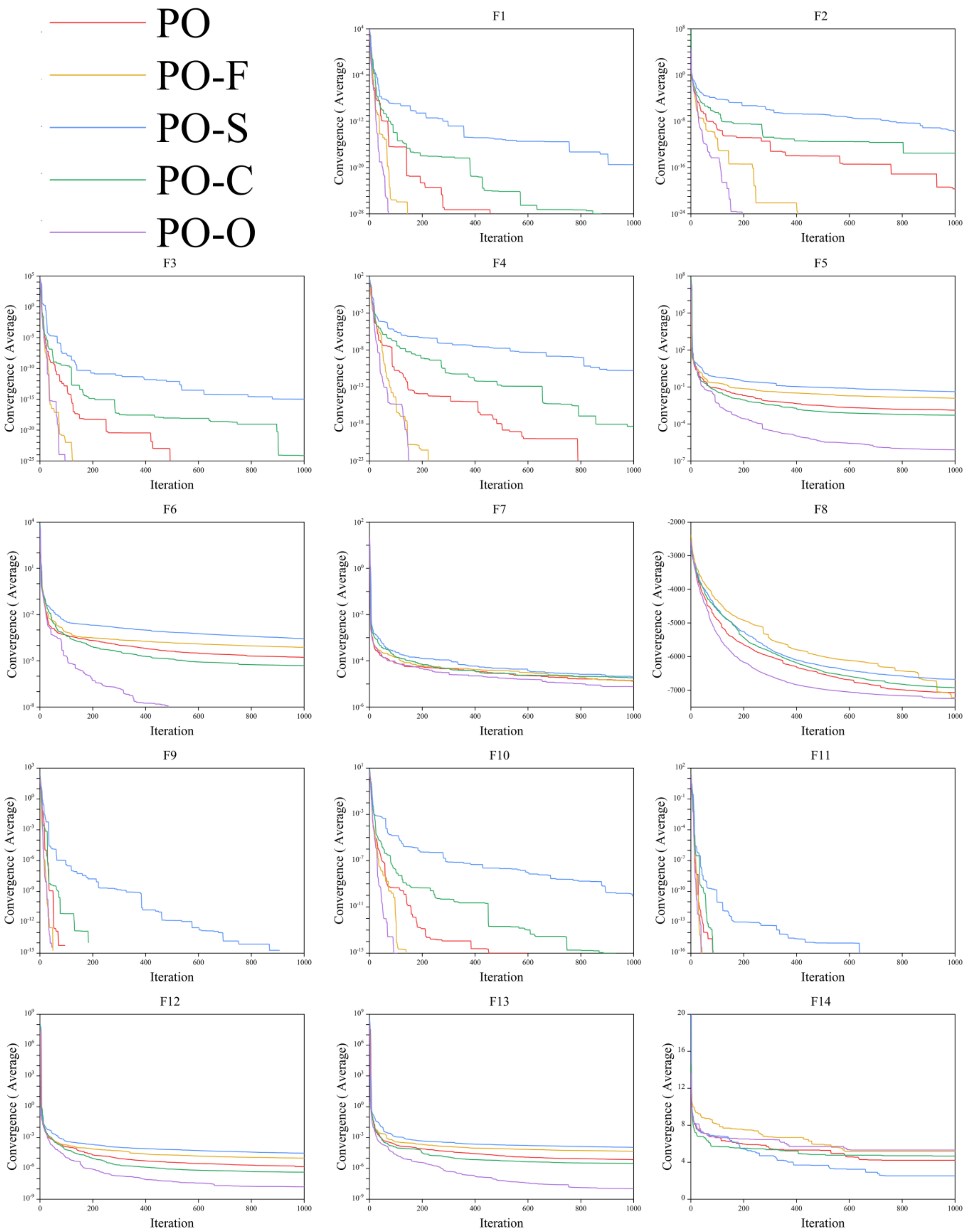


Fig. 10. Comparison of convergence for different parameters (classical test sets)

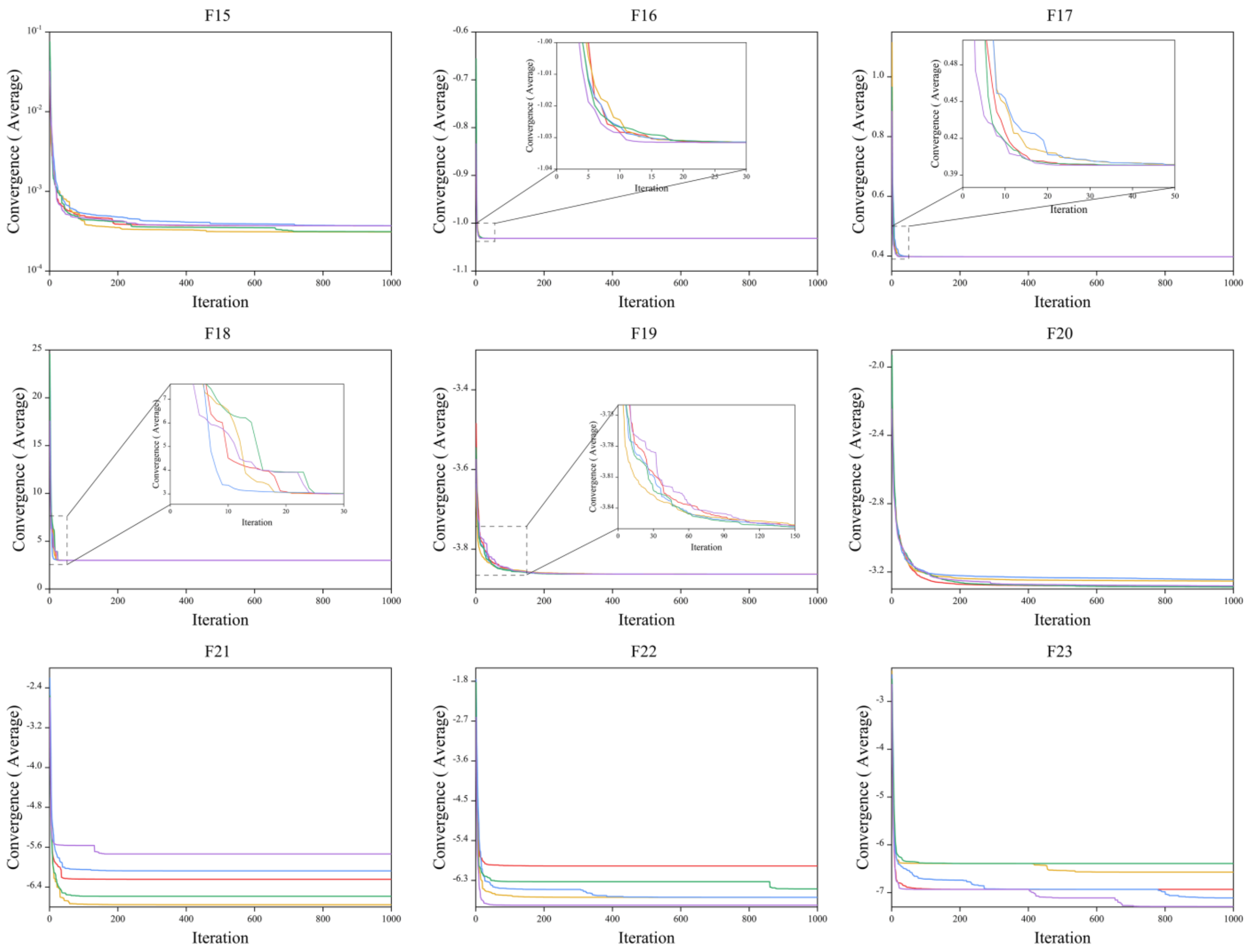


Fig. 10. (Continued)

Table 8. Parameter sensitivity analysis results of unimodal benchmark functions

Function	Item	PO	PO-F	PO-S	PO-C	PO-O
F1	Mean	1.88E-45	6.21E-86	2.51E-20	7.83E-33	**1.23E-104**
	Best	8.52E-121	0.00E+00	0.00E+00	4.79E-73	**0.00E+00**
	Worst	5.43E-44	1.86E-84	6.33E-19	2.34E-31	**3.70E-103**
	Median	5.06E-60	3.73E-159	7.43E-25	1.53E-42	**1.47E-162**
	STD	9.83E-89	1.16E-169	1.35E-38	1.82E-63	**4.55E-207**
F2	Mean	2.05E-20	6.53E-46	1.67E-10	3.11E-14	**2.47E-55**
	Best	8.85E-44	3.47E-148	**0.00E+00**	**0.00E+00**	**0.00E+00**
	Worst	4.04E-19	1.64E-44	2.77E-09	9.31E-13	**7.31E-54**
	Median	3.64E-27	5.63E-70	2.69E-15	7.64E-20	**5.97E-81**
	STD	5.98E-39	9.16E-90	3.37E-19	2.89E-26	**1.78E-108**
F3	Mean	2.73E-29	2.60E-54	1.08E-15	7.54E-25	**2.55E-74**
	Best	4.73E-120	7.79E-246	1.98E-35	1.53E-66	**0.00E+00**
	Worst	8.18E-28	7.81E-53	3.15E-14	2.26E-23	**7.64E-73**
	Median	9.77E-48	3.48E-115	1.03E-23	1.65E-35	**1.53E-138**
	STD	2.23E-56	2.03E-106	3.30E-29	1.70E-47	**1.95E-146**
F4	Mean	4.78E-24	**1.17E-65**	1.67E-11	4.66E-19	3.83E-51
	Best	**0.00E+00**	**0.00E+00**	8.70E-19	6.35E-37	2.23E-169
	Worst	1.43E-22	**3.52E-64**	3.62E-10	1.34E-17	1.15E-49
	Median	5.70E-39	**7.95E-110**	3.62E-13	9.98E-27	6.48E-97
	STD	6.85E-46	**4.13E-129**	4.39E-21	5.96E-36	4.40E-100
F5	Mean	1.32E-03	1.22E-02	4.17E-02	5.15E-04	**8.12E-07**
	Best	1.02E-06	8.68E-06	1.91E-04	4.92E-08	**1.29E-16**
	Worst	5.25E-03	5.76E-02	1.67E-01	3.68E-03	**1.20E-05**
	Median	7.83E-04	6.96E-03	2.95E-02	2.15E-04	**1.22E-08**
	STD	2.24E-06	1.97E-04	2.08E-03	5.79E-07	**5.23E-12**
F6	Mean	1.69E-05	7.51E-05	2.80E-04	4.93E-06	**3.28E-09**
	Best	1.66E-11	1.16E-09	6.31E-07	1.39E-08	**6.78E-13**
	Worst	1.24E-04	4.36E-04	1.07E-03	3.18E-05	**6.00E-08**
	Median	6.49E-06	4.54E-05	1.67E-04	2.69E-06	**1.11E-10**
	STD	6.62E-10	9.26E-09	8.95E-08	4.86E-11	**1.24E-16**

F7	Mean	1.33E-05	1.41E-05	2.02E-05	1.83E-05	**7.68E-06**
	Best	2.03E-06	1.07E-06	7.60E-07	**5.84E-07**	7.20E-07
	Worst	**2.77E-05**	3.61E-05	7.59E-05	7.26E-05	3.77E-05
	Median	1.05E-05	1.02E-05	1.30E-05	1.06E-05	**6.48E-06**
	STD	**5.41E-11**	1.33E-10	3.64E-10	3.85E-10	6.24E-11

Table 9. Parameter sensitivity analysis results of multimodal benchmark functions

Function	Item	PO	PO-F	PO-S	PO-C	PO-O
F8	Mean	-7.07E+03	-7.22E+03	-6.68E+03	-6.92E+03	**-7.25E+03**
	Best	**-9.75E+03**	-1.21E+04	-8.81E+03	-8.50E+03	-9.56E+03
	Worst	-4.56E+03	-4.97E+03	-4.53E+03	**-5.43E+03**	-5.32E+03
	Median	-7.00E+03	-6.71E+03	-6.79E+03	-7.05E+03	**-7.16E+03**
	STD	1.79E+06	2.82E+06	7.66E+05	6.82E+05	**7.08E+05**
F9	Mean	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	Best	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	Worst	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	Median	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
	STD	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
F10	Mean	**8.88E-16**	**8.88E-16**	8.76E-11	**8.88E-16**	**8.88E-16**
	Best	**8.88E-16**	**8.88E-16**	8.88E-16	**8.88E-16**	**8.88E-16**
	Worst	**8.88E-16**	**8.88E-16**	2.02E-09	**8.88E-16**	**8.88E-16**
	Median	**8.88E-16**	**8.88E-16**	4.17E-14	**8.88E-16**	**8.88E-16**
	STD	**0.00E+00**	**0.00E+00**	1.36E-19	**0.00E+00**	**0.00E+00**
F11	Mean	**0.00E+00**	**0.00E+00**	3.33E-17	**0.00E+00**	**0.00E+00**
	Best	**0.00E+00**	**0.00E+00**	0.00E+00	**0.00E+00**	**0.00E+00**
	Worst	**0.00E+00**	**0.00E+00**	9.99E-16	**0.00E+00**	**0.00E+00**
	Median	**0.00E+00**	**0.00E+00**	0.00E+00	**0.00E+00**	**0.00E+00**
	STD	**0.00E+00**	**0.00E+00**	3.33E-32	**0.00E+00**	**0.00E+00**
F12	Mean	1.48E-06	1.02E-05	3.09E-05	4.33E-07	**1.62E-08**
	Best	6.31E-10	2.18E-08	3.48E-07	8.75E-10	**9.90E-16**
	Worst	1.13E-05	4.55E-05	1.94E-04	3.86E-06	**4.53E-07**
	Median	5.92E-07	5.26E-06	2.28E-05	2.03E-07	**5.86E-11**
	STD	4.89E-12	1.32E-10	1.45E-09	5.68E-13	**6.82E-15**
F13	Mean	7.18E-06	4.70E-05	1.11E-04	3.10E-06	**1.09E-08**
	Best	5.79E-09	7.60E-07	1.08E-07	1.23E-11	**4.83E-15**
	Worst	3.48E-05	2.19E-04	4.26E-04	1.65E-05	**1.03E-07**
	Median	2.47E-06	1.95E-05	7.13E-05	4.02E-07	**3.46E-10**
	STD	9.03E-11	3.37E-09	1.39E-08	2.55E-11	**5.06E-16**

Table 10. Parameter sensitivity analysis results of fixed-dimension multimodal benchmark functions

Function	Item	PO	PO-F	PO-S	PO-C	PO-O
F14	Mean	4.22E+00	5.16E+00	**2.53E+00**	4.67E+00	5.32E+00
	Best	**9.98E-01**	**9.98E-01**	**9.98E-01**	**9.98E-01**	**9.98E-01**
	Worst	**1.27E+01**	**1.27E+01**	**1.27E+01**	**1.27E+01**	**1.27E+01**
	Median	1.50E+00	1.99E+00	**9.98E-01**	**9.98E-01**	2.98E+00
	STD	2.31E+01	2.84E+01	**1.11E+01**	2.74E+01	2.85E+01
F15	Mean	3.69E-04	**3.09E-04**	3.71E-04	3.12E-04	3.70E-04
	Best	**3.07E-04**	**3.07E-04**	3.08E-04	3.08E-04	**3.07E-04**
	Worst	1.22E-03	**3.17E-04**	1.22E-03	3.86E-04	1.22E-03
	Median	**3.08E-04**	**3.08E-04**	**3.08E-04**	**3.08E-04**	**3.08E-04**
	STD	5.39E-08	**4.09E-12**	5.37E-08	2.08E-10	5.38E-08
F16	Mean	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**
	Best	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**
	Worst	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**
	Median	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**	**-1.03E+00**
	STD	5.37E-21	1.77E-18	1.87E-19	3.74E-22	**6.26E-30**
F17	Mean	**3.98E-01**	**3.98E-01**	**3.98E-01**	**3.98E-01**	**3.98E-01**
	Best	**3.98E-01**	**3.98E-01**	**3.98E-01**	**3.98E-01**	**3.98E-01**
	Worst	**3.98E-01**	**3.98E-01**	**3.98E-01**	**3.98E-01**	**3.98E-01**
	Median	**3.98E-01**	**3.98E-01**	**3.98E-01**	**3.98E-01**	**3.98E-01**
	STD	3.31E-19	9.66E-17	2.69E-16	3.34E-18	**2.19E-26**
F18	Mean	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**
	Best	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**
	Worst	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**
	Median	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**	**3.00E+00**
	STD	4.72E-19	1.04E-15	3.84E-15	4.83E-19	**1.47E-27**
F19	Mean	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**
	Best	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**
	Worst	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**
	Median	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**	**-3.86E+00**
	STD	4.67E-10	6.87E-09	1.00E-09	1.35E-10	**5.05E-11**
F20	Mean	-3.28E+00	-3.25E+00	-3.25E+00	**-3.29E+00**	-3.28E+00
	Best	**-3.32E+00**	**-3.32E+00**	**-3.32E+00**	**-3.32E+00**	**-3.32E+00**
	Worst	-3.15E+00	-3.11E+00	-3.09E+00	**-3.17E+00**	**-3.17E+00**
	Median	**-3.32E+00**	**-3.32E+00**	-3.26E+00	**-3.32E+00**	**-3.32E+00**
	STD	4.13E-03	5.87E-03	6.82E-03	**3.90E-03**	3.91E-03
F21	Mean	-6.24E+00	**-6.75E+00**	-6.07E+00	-6.58E+00	-5.73E+00
	Best	**-1.02E+01**	**-1.02E+01**	**-1.02E+01**	**-1.02E+01**	**-1.02E+01**

	Worst	**-5.06E+00**	**-5.06E+00**	**-5.06E+00**	**-5.06E+00**	**-5.06E+00**
	Median	**-5.06E+00**	**-5.06E+00**	**-5.06E+00**	**-5.06E+00**	**-5.06E+00**
	STD	4.81E+00	5.97E+00	4.30E+00	5.65E+00	**3.11E+00**
F22	Mean	-5.97E+00	-6.68E+00	-6.68E+00	-6.49E+00	**-6.86E+00**
	Best	**-1.04E+01**	**-1.04E+01**	**-1.04E+01**	**-1.04E+01**	**-1.04E+01**
	Worst	**-5.09E+00**	**-5.09E+00**	**-5.09E+00**	**-5.09E+00**	**-5.09E+00**
	Median	**-5.09E+00**	**-5.09E+00**	**-5.09E+00**	**-5.09E+00**	**-5.09E+00**
	STD	**4.06E+00**	6.14E+00	6.14E+00	5.61E+00	6.49E+00
F23	Mean	-6.93E+00	-6.57E+00	-7.11E+00	-6.39E+00	**-7.29E+00**
	Best	**-1.05E+01**	**-1.05E+01**	**-1.05E+01**	**-1.05E+01**	**-1.05E+01**
	Worst	**-5.13E+00**	**-5.13E+00**	**-5.13E+00**	**-5.13E+00**	**-5.13E+00**
	Median	**-5.13E+00**	**-5.13E+00**	**-5.13E+00**	**-5.13E+00**	**-5.13E+00**
	STD	6.72E+00	**5.91E+00**	7.03E+00	5.41E+00	7.26E+00

Table 11. Rank of five PO algorithms (classical test function)

Function	PO	PO-F	PO-S	PO-C	PO-O
F1	3	2	5	4	1
F2	3	2	5	4	1
F3	3	2	5	4	1
F4	3	1	5	4	2
F5	3	4	5	2	1
F6	3	4	5	2	1
F7	2	3	5	4	1
F8	3	2	5	4	1
F9	3	2	5	4	1
F10	3	2	5	4	1
F11	3	2	5	4	1
F12	3	4	5	2	1
F13	3	4	5	2	1
F14	2	4	1	3	5
F15	3	1	5	2	4
F16	2	5	4	3	1
F17	3	2	5	4	1
F18	2	5	4	3	1
F19	3	5	4	2	1
F20	2	4	5	1	3
F21	3	1	4	2	5
F22	5	2	3	4	1
F23	3	4	2	5	1
Average Rank	2.87	2.91	4.43	3.17	1.61
Final ranking	2	3	5	4	1

To further validate this concept, the five PO algorithms was assembled to address the CEC 2022 test function challenge, maintaining consistent dimensions, population size, and iteration count as specified in Section 3.1. The outcomes are presented in Table 12 and Fig. 11, with the rankings of the five PO algorithms documented in Table 13.

PO exhibits varied optimization capabilities across different test functions and under distinct parameter settings. The optimization outcomes differ as the proportions of the four behavioral formulas are altered, underscoring the equilibrium and effectiveness of the four strategies embedded in the algorithm's design. This observation further validates the applicability of the NFL theory in optimizing the algorithm itself. Additionally, the findings affirm that PO can effectively address real-world problems through sensible adjustments of its parameters. This adaptability highlights PO's potential for achieving more efficient optimization when tailored to the specific requirements of different problems.

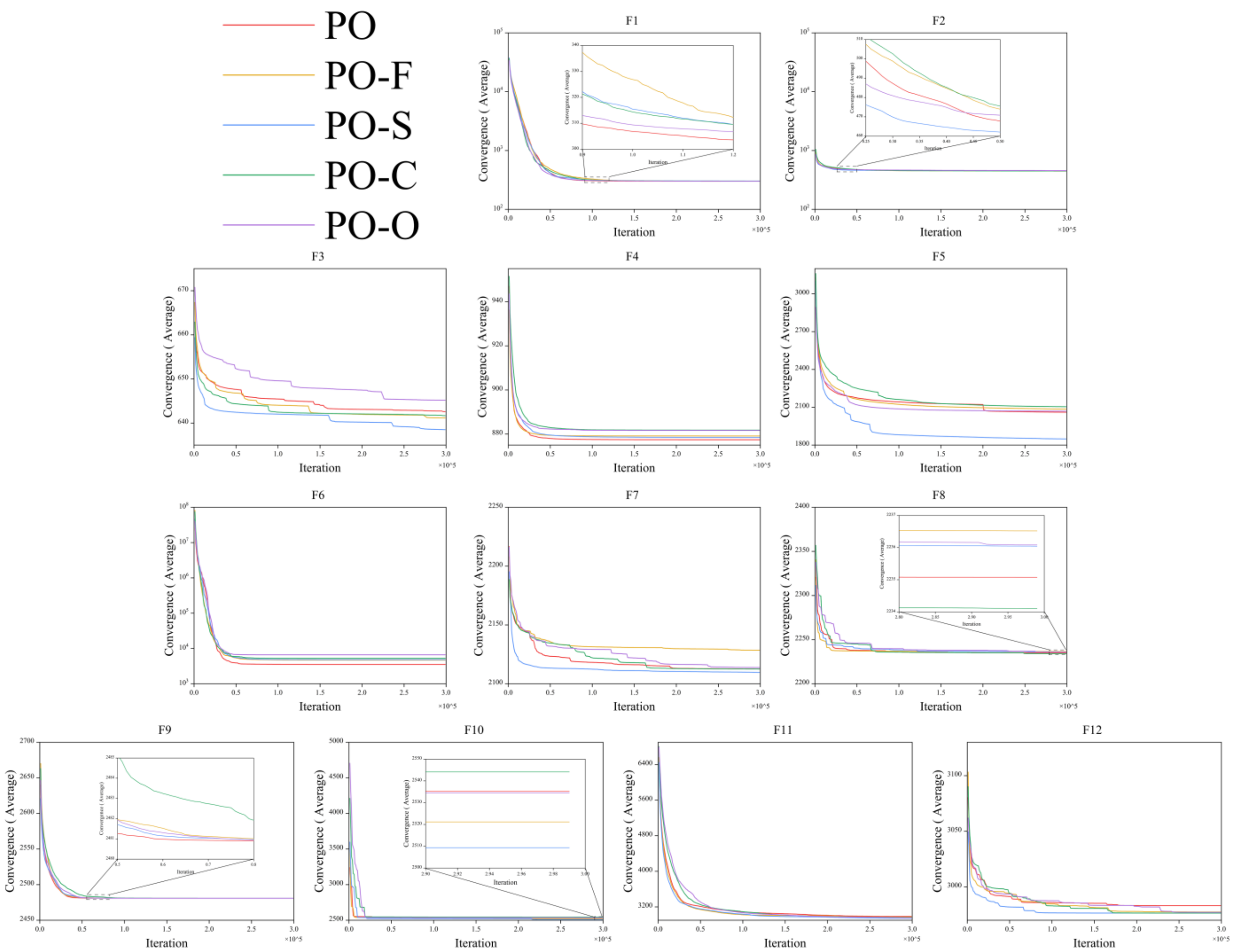


Fig. 11. Comparison of convergence for different parameters (CEC 2022)

Table 12. Parameter sensitivity analysis results of CEC 2022 test sets

Function	Item	PO	PO-F	PO-S	PO-C	PO-O
F1	Mean	**3.00E+02**	3.01E+02	3.03E+02	3.02E+02	3.00E+02
	Best	3.00E+02	3.00E+02	3.00E+02	3.00E+02	**3.00E+02**
	Worst	**3.02E+02**	3.02E+02	3.07E+02	3.06E+02	3.02E+02
	Median	**3.00E+02**	3.00E+02	3.02E+02	3.01E+02	3.00E+02
	STD	**1.63E-01**	5.41E-01	5.73E+00	3.41E+00	2.59E-01
F2	Mean	**4.48E+02**	4.48E+02	4.51E+02	4.49E+02	4.55E+02
	Best	4.05E+02	4.07E+02	**4.05E+02**	4.07E+02	4.06E+02
	Worst	4.75E+02	4.80E+02	4.97E+02	**4.74E+02**	4.76E+02
	Median	**4.49E+02**	4.49E+02	4.49E+02	4.49E+02	4.70E+02
	STD	3.75E+02	4.01E+02	2.64E+02	**2.37E+02**	5.26E+02
F3	Mean	6.43E+02	6.41E+02	**6.38E+02**	6.42E+02	6.45E+02
	Best	**6.19E+02**	6.23E+02	6.24E+02	6.27E+02	6.20E+02
	Worst	6.58E+02	6.64E+02	**6.54E+02**	6.62E+02	6.63E+02
	Median	6.44E+02	6.44E+02	**6.39E+02**	6.41E+02	6.47E+02
	STD	1.01E+02	1.03E+02	**6.08E+01**	7.92E+01	1.01E+02
F4	Mean	**8.77E+02**	8.79E+02	8.78E+02	8.82E+02	8.82E+02
	Best	8.43E+02	8.52E+02	8.45E+02	8.60E+02	**8.42E+02**
	Worst	9.09E+02	9.05E+02	9.13E+02	**8.99E+02**	9.15E+02
	Median	**8.77E+02**	8.80E+02	8.79E+02	8.84E+02	8.82E+02
	STD	2.28E+02	**1.56E+02**	2.30E+02	1.63E+02	3.54E+02
F5	Mean	2.06E+03	2.09E+03	**1.85E+03**	2.10E+03	2.07E+03
	Best	1.37E+03	1.66E+03	**1.22E+03**	1.43E+03	1.48E+03
	Worst	2.45E+03	2.50E+03	**2.37E+03**	2.55E+03	2.47E+03
	Median	2.11E+03	2.02E+03	**1.81E+03**	2.35E+03	2.14E+03
	STD	8.02E+04	**5.24E+04**	1.06E+05	1.57E+05	8.75E+04
F6	Mean	**3.53E+03**	4.69E+03	4.81E+03	5.28E+03	6.59E+03
	Best	2.10E+03	2.05E+03	2.00E+03	2.00E+03	**1.98E+03**
	Worst	**5.49E+03**	1.56E+04	2.48E+04	1.98E+04	1.74E+04
	Median	**3.34E+03**	3.84E+03	3.36E+03	3.90E+03	4.73E+03
	STD	**8.58E+05**	9.53E+06	2.41E+07	1.75E+07	2.28E+07
F7	Mean	2.11E+03	2.13E+03	**2.11E+03**	2.11E+03	2.11E+03
	Best	2.05E+03	2.09E+03	2.07E+03	2.05E+03	**2.05E+03**
	Worst	2.17E+03	2.17E+03	**2.14E+03**	2.16E+03	2.15E+03

	Median	2.12E+03	2.13E+03	**2.11E+03**	2.12E+03	2.13E+03
	STD	1.22E+03	**3.55E+02**	6.25E+02	9.76E+02	6.66E+02
F8	Mean	2.24E+03	2.24E+03	2.24E+03	**2.23E+03**	2.24E+03
	Best	**2.22E+03**	2.23E+03	2.23E+03	2.23E+03	2.23E+03
	Worst	2.25E+03	2.25E+03	**2.25E+03**	2.25E+03	2.26E+03
	Median	2.24E+03	2.23E+03	2.23E+03	**2.23E+03**	2.23E+03
	STD	**3.41E+01**	4.38E+01	4.11E+01	3.76E+01	6.33E+01
F9	Mean	2.48E+03	2.48E+03	2.48E+03	2.48E+03	**2.48E+03**
	Best	**2.48E+03**	2.48E+03	2.48E+03	2.48E+03	2.48E+03
	Worst	2.48E+03	2.48E+03	2.48E+03	2.48E+03	**2.48E+03**
	Median	2.48E+03	2.48E+03	2.48E+03	2.48E+03	**2.48E+03**
	STD	1.03E-04	5.71E-04	3.78E-04	1.13E-02	**2.19E-05**
F10	Mean	2.54E+03	2.52E+03	**2.51E+03**	2.54E+03	2.53E+03
	Best	**2.50E+03**	2.50E+03	2.50E+03	2.50E+03	2.50E+03
	Worst	2.76E+03	2.72E+03	**2.67E+03**	2.75E+03	2.74E+03
	Median	2.50E+03	**2.50E+03**	2.50E+03	2.50E+03	2.50E+03
	STD	6.76E+03	3.76E+03	**1.37E+03**	7.63E+03	6.43E+03
F11	Mean	2.99E+03	2.97E+03	**2.94E+03**	2.96E+03	2.95E+03
	Best	2.90E+03	2.90E+03	2.60E+03	2.90E+03	**2.60E+03**
	Worst	3.28E+03	3.30E+03	3.30E+03	**3.22E+03**	3.35E+03
	Median	2.97E+03	2.91E+03	**2.91E+03**	2.92E+03	2.92E+03
	STD	1.10E+04	1.15E+04	1.42E+04	**8.84E+03**	3.85E+04
F12	Mean	2.98E+03	2.98E+03	**2.98E+03**	2.98E+03	2.98E+03
	Best	2.94E+03	2.94E+03	2.94E+03	**2.94E+03**	2.94E+03
	Worst	3.07E+03	3.04E+03	3.06E+03	3.05E+03	**3.03E+03**
	Median	2.98E+03	2.97E+03	2.97E+03	2.97E+03	**2.97E+03**
	STD	1.28E+03	8.47E+02	6.63E+02	8.20E+02	**6.42E+02**

Table 13. Rank of five PO algorithms (CEC 2022)

Function	PO	PO-F	PO-S	PO-C	PO-O
F1	1	3	5	4	2
F2	1	2	4	3	5
F3	4	2	1	3	5
F4	1	3	2	5	4
F5	2	4	1	5	3
F6	1	2	3	4	5
F7	3	5	1	2	4
F8	2	5	3	1	4
F9	2	4	3	5	1
F10	4	2	1	5	3
F11	5	4	1	3	2
F12	5	4	1	2	3
Average Rank	2.58	3.33	2.17	3.50	3.42
Final Ranking	2	3	1	5	4

The outcomes indicate that the inherently more stable PO baseline algorithm excels in CEC2022, showcasing not only its persistent optimization capacity but also its precision in search within test functions F1, F2, and F4. PO-O frequently seeks the best results but exhibits inconsistency. This emphasizes the necessity of configuring PO's parameters in alignment with the particular optimization problem.

3.3 Engineering design applications

In this section, the recently introduced PO algorithm is employed to address three distinct engineering design challenges [69], and the outcomes are subsequently presented. Formulated to pursue optimal solutions, these engineering optimization problems are designed to comply with predefined conditions and constraints. Typically, constrained optimization problems are not inherently addressed by metaheuristic algorithms. However, through the integration of constraint handling techniques (CHTs), these algorithms adeptly handle both the objective function and the accompanying constraints. At each iteration, the candidate population's fitness is measured by simultaneously considering both the objective function and the constraints. Afterward, the following generation of candidate populations undergoes evaluation based on the calculated fitness values. By employing the PO algorithm in these engineering design challenges, it accomplishes the simultaneous evaluation of both the objective function and constraints. This enables the search for optimal solutions that satisfy the design fundamentals and limitations. The efficiency of the PO algorithm in handling complicated engineering optimization challenges is showcased via the incorporation of CHTs. The five parameter scenarios of PO are consistently compared with other models using a population size of 50 and 1000 iterations.

3.3.1 Tension/compression spring design (TSD) problem

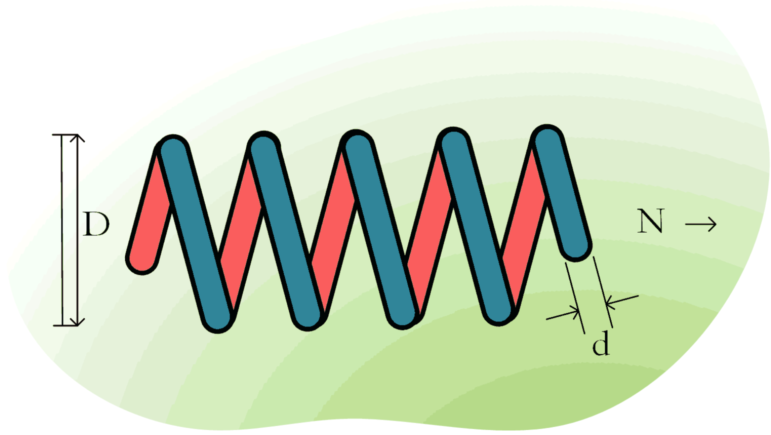


Fig. 12. TSD problem

In this problem, the main objective is to minimize the weight of the tension/compression spring, as illustrated in Fig. 12. The problem involves three decision variables: wire diameter ($d = x_1$), mean coil diameter ($D = x_2$), and the number of active coils ($N = x_3$). To address this optimization challenge, Eq. (9) is formulated. The proposed PO is compared with the following optimization algorithms, GSA [61], CPSO [70], CC [71], RO [72], GA [73], PSO [70]. The results are listed in Table 14 and showed that the PO obtained the best results than all other algorithms. It is worth noting that PO-O, PO-F, PO-C and PO-S are just as good as PO for obtaining the best approximate optimal solution 0.01267.

Minimize:

$$f(\vec{l}) = (l_3 + 2)l_2 l_1^2 \tag{8}$$

subject to:

$$g_1(\vec{l}) = 1 - \frac{4l_2^2 - l_2 l_3}{717851^4} \leq 0,$$

$$g_2(\vec{l}) = \frac{4l_2^2 - l_1 l_2}{12566(l_3 l_1^3 - l_1^4)} + \frac{1}{5108 l_1^2} \leq 0,$$

$$g_3(\vec{l}) = 1 - \frac{140.45 l_1}{l_2^2 l_3} \leq 0,$$

$$g_4(\vec{l}) = \frac{l_1 + l_2}{1.5} - 1 \leq 0,$$

with bounds:

$$0.05 \leq l_1 \leq 2.00, 0.25 \leq l_2 \leq 1.30, 2.00 \leq l_3 \leq 15.0$$

Table 14. Comparison results of TSD problem

Algorithm	Optimum variables			Optimum cost
	d	*D*	*N*	
PO	**0.051897**	**0. 361749**	**10.748207**	**0.0126660**
GSA	0.050276	0. 323680	13.525410	0.0127022
CPSO	0.051728	0.357644	11.244543	0.0126747
CC	70.050000	0.315900	14.250000	0.0128334
RO	0.051370	0.349096	11.76279	0.0126788
GA	0.051 480	0.351661	11 .632201	0.01270478
PSO	0.051728	0.357644	11.244543	0.0126747

3.3.2 Welded beam design (WBD) problem

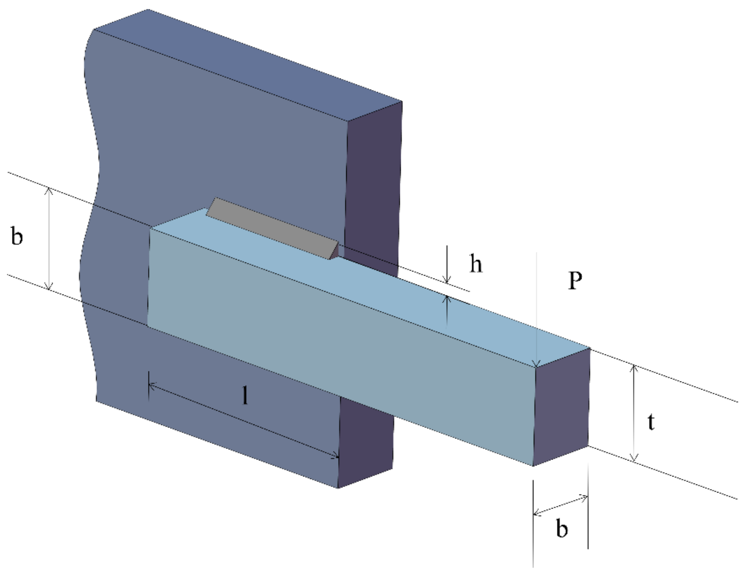


Fig. 13. WBD problem

The objective of this case is to determine the welded beam with the lowest cost given four limitations and the key characteristics of shear stress (τ), bending stress (θ), buckling load (P_c) and deflection (δ). As indicated in Fig. 13, this task includes the four variables: welding seam thickness (h); welding joint length (l); beam width (t); beam thickness (b). To address this optimization challenge, Eq. (11) is formulated. The proposed PO is compared with the following optimization algorithms, RIME [45], RO [72], SSA [74], CDE [70], GWO [60], GSA [43], NDE [75]. The results are listed in Table 15 and showed that the PO obtained the best results than all other algorithms. It is worth noting that PO-F, PO-C and PO-O are just as good as PO for obtaining the best approximate optimal solution 1.67.

Consider:

$$\vec{x} = [x_1\ x_2\ x_3\ x_4] = [h\ l\ t\ b]$$

Minimize:

$$f(\vec{x}) = 1.10471x_1^2 + 0.04811x_3x_4(14.0+x_4) \tag{9}$$

subject to:

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

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

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

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

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

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

$$g_7(\vec{x}) = 1.10471x_1^2 + 0.04811x_3x_4(14.0+x_2) - 5.0 \leq 0$$

with bounds:

$$0.1 \leq x_1 \leq 2,\ 0.1 \leq x_2 \leq 10,\ 0.1 \leq x_3 \leq 10,\ 0.1 \leq x_4 \leq 2$$

where:

$$\tau(\vec{x}) = \sqrt{\left(\tau'\right)^2 + 2\tau'\tau''\frac{x_2}{2R} + \left(\tau''\right)^2},\ \tau' = \frac{P}{\sqrt{2}x_1x_2}\quad \tau'' = \frac{MR}{J}\quad M = P\left(L + \frac{x_2}{2}\right)$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}$$

$$J = 2\left\{\sqrt{2}x_1x_2\left[\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\}$$

$$\sigma(\vec{x}) = \frac{6PL}{x_4{x_3}^2},\delta(\vec{x}) = \frac{6PL^3}{Ex_3^2x_4}$$

$$P_C(\vec{x}) = \frac{4.013E\sqrt{\frac{x_3^2 x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$$

$$P = 60001b,\ L = 14 \in ..\delta_{max} = 0.25 \in..$$

$$E = 30 \times 1^6 psi, G = 12 \times 10^6 psi$$

$$\tau_{max} = 13600psi, \sigma_{max} = 30000psi$$

Table 15. Comparison results of WBD problem

Algorithm	Optimum variables				Optimum cost
	h	***l***	***t***	***b***	
PO	**0.198601**	**3.341753**	**9.192220**	**0.198831**	**1.670481**
RIME	0.208000	3.250000	9.053702	0.208620	1.722821
RO	0.203687	3.528467	9.004233	0.207241	1.735344
SSA	0.205700	3.471400	9.036600	0.205700	1.724910
CDE	0.203137	3.542998	9.033498	0.206179	1.733462
GWO	0.205700	3.478400	9.036800	0.205800	1.726240
GSA	0.182129	3.856979	10.00000	0.202376	1.879950
NDE	0.205729	3.470488	9.903662	0.205729	1.724852

3.3.3 Multiple disk clutch brake design (MDCBD) problem

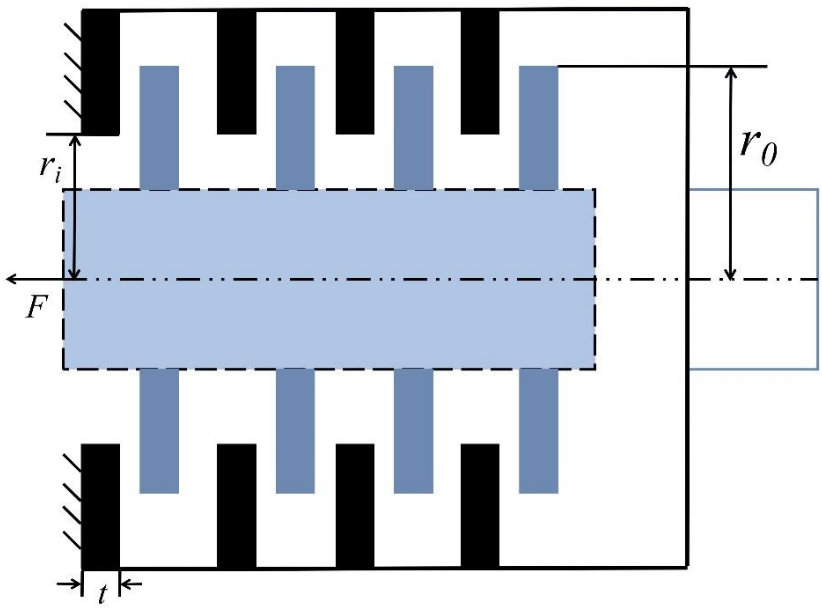


Fig. 14. MDCBD problem

The multiple disk clutch brake design problem is defined using Eq. (12) to minimize the mass of the multiple disk clutch brake. Nine nonlinear constraints characterize this particular problem and encompasses five discrete design variables, namely the inner radius (x_1), outer radius (x_2), disk thickness (x_3), actuator force (x_4), and number of frictional surfaces

(x_5).The schematic representation of this problem is depicted in Fig. 14. The proposed PO is compared with the following optimization algorithms, RIME [45],CBA [76], WCA [77], PVS [78], TLBO [48]. The results are listed in Table 16 and showed that the PO obtained the best results than all other algorithms. It is worth noting that PO-O, PO-F, PO-C and PO-S are just as good as PO for obtaining the best approximate optimal solution 0.2352.

Minimize:

$$f(\bar{x}) = \pi(x_2^2 - x_1^2)x_3(x_5 + 1)\rho \tag{10}$$

subject to:

$$g_1(\bar{x}) = -p_{max} + p_{rz} \leq 0,$$

$$g_2(\bar{x}) = p_{rz}V_{sr} - V_{sr,max}p_{max} \leq 0,$$

$$g_3(\bar{x}) = \Delta R + x_1 - x_2 \leq 0,$$

$$g_4(\bar{x}) = -L_{max} + (x_5 + 1)(x_3 + \delta) \leq 0,$$

$$g_5(\bar{x}) = sM_s - M_h \leq 0,$$

$$g_6(\bar{x}) = T \geq 0$$

$$g_6(\bar{x}) = T \geq 0,$$

$$g_7(\bar{x}) = -V_{sr,max} + V_{sr} \leq 0,$$

$$g_7(\bar{x}) = T - T_{max} \leq 0,$$

where,

$$M_h = \frac{2}{3}\mu x_4 x_5 \frac{x_2^3 - x_1^3}{x_2^2 - x_1^2} N \cdot \text{mm},$$

$$\omega = \frac{\pi n}{30}\frac{\text{rad}}{s},$$

$$A = \pi(x_2^2 - x_1^2)\text{mm}^2,$$

$$p_{rz} = \frac{x_4}{A}\frac{N}{\text{mm}^2},$$

$$V_{sr} = \frac{\pi R_{sr} n}{30}\frac{\text{mm}}{s},$$

$$R_{sr} = \frac{2}{3}\frac{x_2^3 - x_1^3}{x_2^2 x_1^2}\text{mm},$$

$$T = \frac{I_z\omega}{M_h + M_f},$$

$$\Delta R = 20\text{mm}, L_{max} = 30\text{mm}, \mu=0.6,$$

$$V_{sr,max} = 10\frac{m}{s}, \delta = 0.5\text{mm}, s=1.5,$$

$$T_{max} = 15s, n = 250rpm, I_z = 55\text{Kg} \cdot m^2,$$

$$M_s = 40\text{Nm}, M_f = 3\text{Nm}, \text{and} p_{max} = 1$$

with bounds:

$$60 \leq x_1 \leq 80, 90 \leq x_2 \leq 110, 1 \leq x_3 \leq 3$$

$$0 \leq x_4 \leq 1000, 2 \leq x_5 \leq 9.$$

Table 16. Comparison results of MDCBD problem

Algorithm	**Optimum variables**					**Optimum cost**
	r_i	r_0	t	F	Z	
PO	**70.0000**	**90.0000**	**1.0000**	**1000.0000**	**2.0000**	**0.235242**
RIME	75.0000	95.0000	1.0000	1000.0000	2.0000	0.249945
CBA	80.0000	90.0000	3.0000	1000.0000	2.0000	0.263684
WCA	70.0000	90.0000	1.0000	910.0000	3.0000	0.313656
PVS	70.0000	90.0000	1.0000	980.0000	3.0000	0.313660
TLBO	70.0000	90.0000	1.0000	810.0000	3.0000	0.313656

3.4 Real-world medical applications

The growing fascination with medical data systems utilizing images arises from their crucial impact on improving accuracy in diagnoses, formulating treatment strategies, and advancing holistic patient care within the healthcare domain [79, 80]. Therefore, one of the crucial requirements for an optimizer is its ability to serve as a core component in optimizing solutions for medical problems. To assess this, this section showcases the utilization of PO in the realm of medical applications, highlighting its optimization capabilities. We explore two primary use cases within the medical domain: tuning model parameters for medical problems and addressing medical image segmentation challenges. These practical examples underscore the efficacy of PO in medical optimization and its competitive edge against established algorithms.

3.4.1 Application of hyperparameter tuning

Table 17. Algorithms involved in the comparison and their parameter settings (hyperparameter tuning)

Algorithms	Name of parameters	Value of parameters
GWO [60]	α	Linear reduction from 2 to 0
MVO [44]	WEP_{max}	1
	WEP_{min}	0.2

SCA [59]	-	-
SSA [74]	v_0	0

Support vector machine (SVM) is a supervised machine learning method grounded in statistical learning theory. While SVM offers certain advantages in theory and application, its effectiveness hinges on selecting appropriate parameters. Traditional SVM techniques are gradually being replaced by more efficient methods, such as intelligent optimization algorithms, particularly when addressing medical problems [63]. The latter are preferred for their ability to expedite the process, as traditional SVM methods are often time-consuming and inefficient in their search for optimal solutions.

Table 18. Comparison results of hyperparameter tuning problem

Dataset	Item	PO	GWO	MVO	SCA	SSA
DARWIN	Mean	**8.407E+01**	8.330E+01	6.863E+01	8.314E+01	8.258E+01
	Best	**9.412E+01**	**9.412E+01**	9.118E+01	**9.412E+01**	**9.412E+01**
	Worst	**7.647E+01**	6.765E+01	3.529E+01	7.059E+01	7.059E+01
	Median	8.235E+01	**8.382E+01**	7.647E+01	8.235E+01	8.235E+01
	STD	**3.021E+01**	3.711E+01	3.601E+02	3.032E+01	3.325E+01
Cryotherapy	Mean	**8.757E+01**	8.651E+01	8.745E+01	8.508E+01	8.644E+01
	Best	**1.000E+02**	**1.000E+02**	**1.000E+02**	**1.000E+02**	9.444E+01
	Worst	**7.778E+01**	7.222E+01	7.222E+01	6.667E+01	**7.778E+01**
	Median	**8.889E+01**	8.611E+01	**8.889E+01**	8.611E+01	**8.889E+01**
	STD	5.489E+01	3.842E+01	5.394E+01	6.118E+01	**2.489E+01**

To illustrate PO's applicability in the medical domain, we've chosen publicly available datasets, including the DARWIN and Cryotherapy datasets from the UCI Machine Learning Library. These datasets are used for disease diagnosis in conjunction with the SVM model. We have selected the GWO, MVO, SCA, SSA as reference algorithms for comparative analysis, highlighting PO's challenging competitive edge. The parameters for each algorithm are standardized, with a population size of 30 and a consistent 100 iterations. The training-to-test set ratio is maintained at 8:2 and repeated 30 times to mitigate the influence of chance. The results are summarized in Table 18. The results show that PO is highly accurate and acceptable, and it can be competitive in optimizing hyperparameters to solve medical problems. In terms of consistency and stability, PO also reveals a competitive performance on DARWIN dataset. It demonstrates lower STD compared to other peers across DARWIN dataset, suggesting a more consistent performance.

3.4.2 Applications of medical image segmentation

Multi-Threshold Image Segmentation (MTIS) is a powerful method for partitioning an image into distinct regions using one or more thresholds. Unlike traditional binary segmentation, MTIS efficiently handles images with multiple objects or discontinuous color and brightness variations. It involves comparing pixel intensities with a set of thresholds to assign pixels to different regions. Although MTIS has low computational complexity, traditional methods may fail to utilize spatial information and can lead to segmentation errors, especially when objects occupy a small portion of the image. To address this, we developed an MTIS model combining non-local mean filtering, 2D histograms, Kapur entropy, and the PO algorithm [62]. This approach enhances image quality and minimizes errors while keeping computational costs in check.

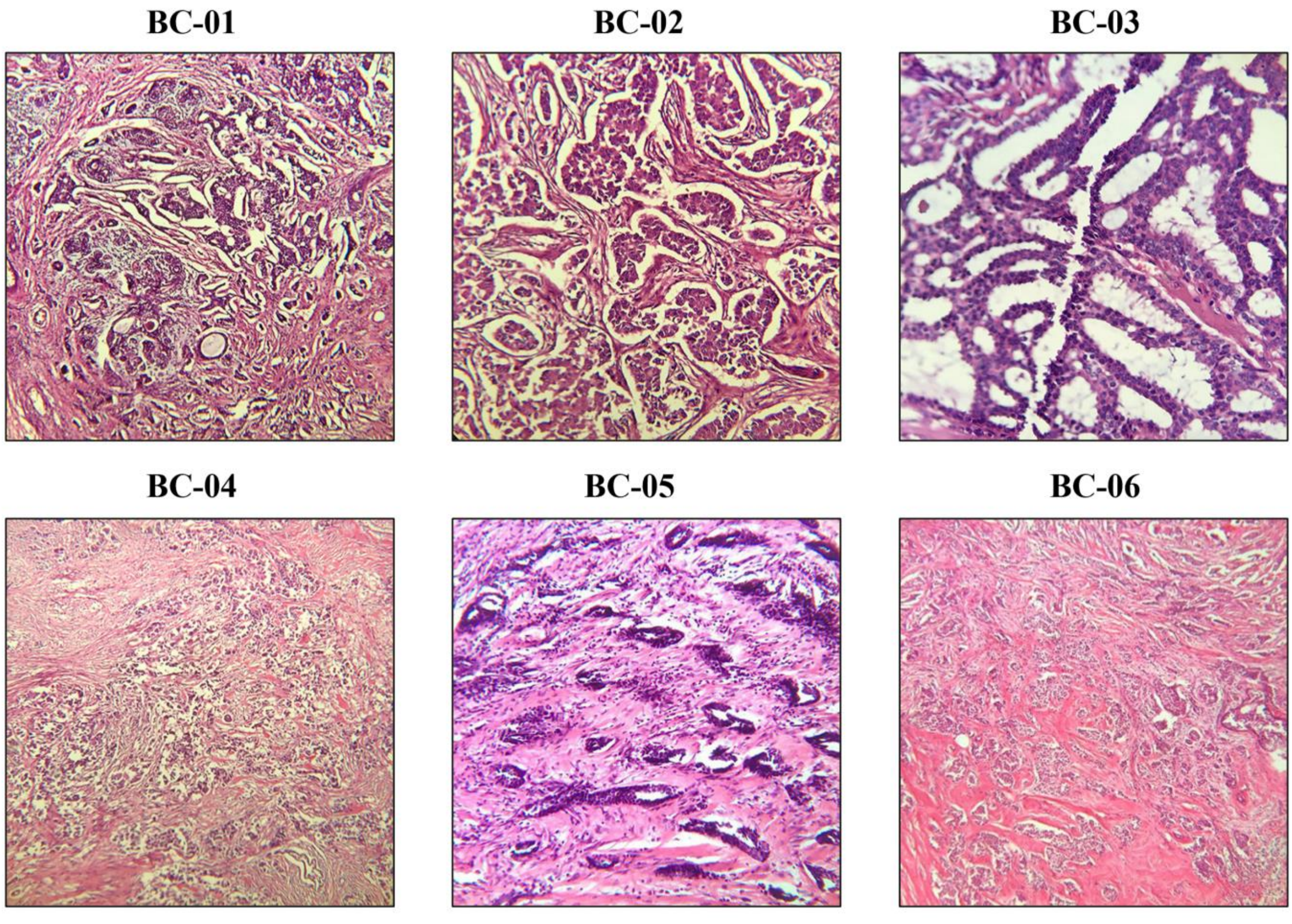


Fig. 15. The images used for testing the performance

The process begins with transforming the input image into grayscale and applying non-local mean filtering to reduce noise. A 2D histogram is constructed, and Kapur's entropy is used to calculate information content for different threshold combinations. The goal is to maximize entropy, signifying the selection of thresholds that optimize information preservation. Metaheuristic algorithms are employed for this purpose. The optimal threshold set is then used to segment the image, resulting in a refined segmented image. Choosing proper image data for validations of methods holds a significant weight, requiring meticulous evaluation of multiple essential elements [81, 82]. Hence, in this section, six cancer pathology images were utilized (see Fig. 15), sourced from the *invasive ductal carcinoma (IDC)* dataset [83]. These images are histopathology microscopy images of breast tissues stained with H&E, and they are annotated based on their grade and magnification level. To provide a clear overview of the MTIS operation process, a flowchart is presented as shown in Fig. 16 [64]. The goal was to assess the segmentation performance of PO across different images, recognizing that each image poses a distinct problem related to the choice of the segmentation threshold [84].

Table 19. Algorithms involved in the comparison and their parameter settings (image segmentation)

Algorithms	Name of parameters	Value of parameters
SSA [74]	v_0	0
MFO [41]	α	Decreased from -1 to -2
	b	1
PSO [33]	c_1, c_2	2
	w	[0.2, 0.9]
BA [42]	A	0.5
	r	0.5
WOA [65]	α	Decreased from 2 to 0

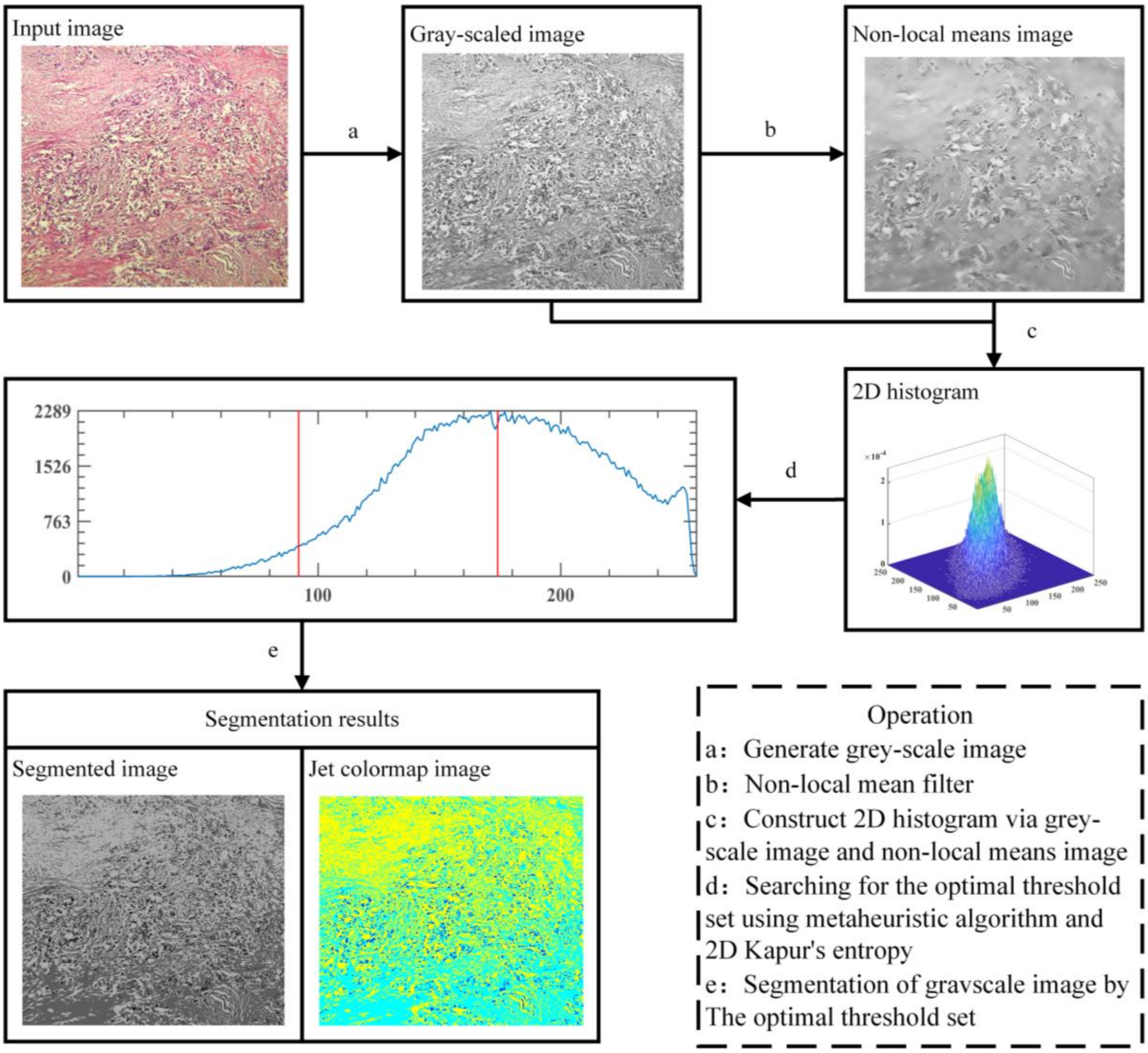


Fig. 16. The flowchart of MTIS method

Two sets of experiments were conducted with low and high threshold levels to thoroughly evaluate PO, and the segmentation results were compared with five other algorithms. Table 18 provides a list of the compared algorithms along with their parameter settings. Our evaluation relied on three key metrics: Peak Signal-to-Noise Ratio (PSNR) [85], Structural Similarity Index (SSIM) [86], and Feature Similarity Index (FSIM) [87]. The mean and variance of these metrics were calculated. It's worth noting that all the compared algorithms were subjected to identical experimental conditions, running for 100 iterations, with an image size of 512×512, a solution space size of 30, and each algorithm was independently run 10 times.

Table 20. The PSNR comparison results

IMAGE		2 thresholds		4 thresholds		10 thresholds	
		Mean	**STD**	**Mean**	**STD**	**Mean**	**STD**
BC-01	PO	**9.486E-01**	**1.464E-04**	9.565E-01	1.491E-04	9.719E-01	3.066E-05
	SSA	9.475E-01	1.565E-04	9.557E-01	2.991E-04	**9.787E-01**	6.348E-05
	MFO	9.347E-01	1.575E-03	9.626E-01	5.173E-05	9.649E-01	4.020E-06
	PSO	9.239E-01	6.818E-04	**9.715E-01**	**1.806E-05**	9.654E-01	**2.418E-06**
	BA	8.676E-01	3.646E-02	9.606E-01	2.280E-04	9.787E-01	1.802E-04
	WOA	9.333E-01	1.588E-03	9.667E-01	1.584E-04	9.675E-01	5.199E-06
BC-02	PO	9.216E-01	8.075E-04	9.429E-01	1.529E-04	9.489E-01	2.667E-04
	SSA	9.108E-01	9.374E-04	9.370E-01	2.380E-04	**9.685E-01**	**8.505E-05**
	MFO	9.213E-01	2.433E-04	9.434E-01	1.001E-04	9.489E-01	3.573E-04
	PSO	**9.273E-01**	**1.245E-04**	9.392E-01	2.090E-04	9.576E-01	4.676E-04
	BA	9.005E-01	6.127E-04	9.420E-01	**4.452E-05**	9.663E-01	3.590E-04
	WOA	9.107E-01	1.107E-03	**9.477E-01**	9.861E-05	9.540E-01	5.279E-04
BC-03	PO	8.231E-01	5.909E-04	8.893E-01	1.166E-03	9.459E-01	2.507E-03
	SSA	8.271E-01	6.036E-04	**8.951E-01**	**2.094E-04**	9.560E-01	1.535E-04
	MFO	8.127E-01	1.408E-03	8.767E-01	1.159E-03	9.683E-01	6.101E-05
	PSO	8.292E-01	6.294E-04	8.718E-01	1.631E-03	**9.695E-01**	2.972E-05
	BA	**8.440E-01**	2.338E-04	8.729E-01	1.019E-03	9.688E-01	**1.119E-05**
	WOA	8.312E-01	**1.971E-04**	8.944E-01	3.800E-04	9.467E-01	7.568E-04
BC-04	PO	9.015E-01	4.284E-04	9.076E-01	2.482E-04	9.178E-01	7.966E-04
	SSA	**9.091E-01**	**8.062E-05**	**9.150E-01**	2.954E-04	9.387E-01	1.164E-03
	MFO	8.937E-01	3.659E-04	9.042E-01	5.125E-04	9.173E-01	**1.783E-04**
	PSO	8.924E-01	5.367E-04	9.083E-01	4.882E-04	9.525E-01	7.668E-04
	BA	8.914E-01	9.582E-04	9.032E-01	**2.391E-04**	**9.631E-01**	7.440E-04
	WOA	8.937E-01	2.808E-04	8.968E-01	6.138E-04	9.202E-01	5.527E-04
BC-05	PO	8.759E-01	3.169E-05	9.193E-01	6.222E-05	9.193E-01	6.222E-05
	SSA	8.777E-01	**1.908E-05**	**9.244E-01**	**1.070E-04**	**9.244E-01**	**1.070E-04**
	MFO	8.728E-01	5.985E-05	9.186E-01	7.256E-05	9.186E-01	7.256E-05
	PSO	8.766E-01	1.548E-05	9.162E-01	7.004E-05	9.162E-01	7.004E-05
	BA	8.748E-01	9.567E-06	9.153E-01	3.136E-05	9.153E-01	3.136E-05
	WOA	**8.782E-01**	4.348E-05	9.178E-01	8.163E-05	9.178E-01	8.163E-05
BC-06	PO	8.051E-01	5.448E-04	8.439E-01	8.432E-04	9.539E-01	3.272E-04
	SSA	**8.067E-01**	**3.294E-04**	8.331E-01	7.307E-04	9.430E-01	2.103E-04
	MFO	8.039E-01	9.720E-05	8.441E-01	5.814E-04	**9.641E-01**	**4.225E-05**
	PSO	8.045E-01	5.264E-04	8.362E-01	**5.492E-04**	9.558E-01	1.215E-03
	BA	7.967E-01	1.673E-03	**8.521E-01**	1.238E-03	9.585E-01	3.700E-04
	WOA	7.816E-01	3.530E-03	8.200E-01	1.204E-03	9.609E-01	5.126E-04

Table 21. The SSIM comparison results

IMAGE		2 thresholds		4 thresholds		10 thresholds	
		Mean	STD	Mean	STD	Mean	STD
BC-01	PO	**1.309E+01**	**2.069E-01**	1.507E+01	9.614E-01	2.024E+01	2.549E-01
	SSA	1.295E+01	2.327E-01	1.506E+01	2.048E+00	2.024E+01	9.946E-01
	MFO	1.232E+01	2.075E+00	1.534E+01	4.563E-01	2.037E+01	3.860E-02
	PSO	1.169E+01	1.229E+00	**1.613E+01**	**1.689E-01**	2.036E+01	**2.967E-02**
	BA	1.241E+01	4.583E+00	1.523E+01	1.842E+00	**2.116E+01**	5.539E-01
	WOA	1.232E+01	1.974E+00	1.582E+01	9.581E-01	2.000E+01	7.124E-02
BC-02	PO	1.259E+01	2.158E+00	1.591E+01	7.180E-01	1.982E+01	**5.391E-01**
	SSA	1.256E+01	7.483E-01	1.541E+01	1.484E+00	2.041E+01	9.385E-01
	MFO	1.261E+01	8.341E-01	1.589E+01	3.831E-01	2.036E+01	6.770E-01
	PSO	**1.305E+01**	5.524E-01	1.571E+01	1.047E+00	2.082E+01	1.181E+00
	BA	1.234E+01	1.730E-01	1.600E+01	1.179E-01	**2.129E+01**	8.574E-01
	WOA	1.231E+01	2.224E+00	**1.617E+01**	3.023E-01	1.980E+01	1.850E+00
BC-03	PO	1.203E+01	4.138E-01	1.626E+01	4.028E-01	2.101E+01	5.508E+00
	SSA	1.234E+01	5.377E-01	**1.641E+01**	**2.826E-01**	2.119E+01	5.372E-01
	MFO	1.188E+01	7.973E-01	1.571E+01	1.294E+00	2.213E+01	2.845E-01
	PSO	1.239E+01	5.911E-01	1.592E+01	6.836E-01	2.212E+01	3.125E-01
	BA	**1.277E+01**	4.131E-01	1.583E+01	5.069E-01	**2.215E+01**	**1.637E-01**
	WOA	1.255E+01	**3.897E-01**	1.595E+01	5.800E-01	2.087E+01	2.634E+00
BC-04	PO	1.451E+01	2.017E+00	1.603E+01	2.240E+00	1.968E+01	6.734E-01
	SSA	**1.465E+01**	**1.609E-01**	**1.674E+01**	1.006E+00	1.951E+01	4.353E+00
	MFO	1.443E+01	1.359E+00	1.656E+01	**7.075E-01**	1.986E+01	**1.938E-01**
	PSO	1.450E+01	4.018E-01	1.664E+01	8.720E-01	2.077E+01	5.784E-01
	BA	1.456E+01	4.443E-01	1.548E+01	1.545E+00	**2.180E+01**	1.379E+00
	WOA	1.376E+01	1.621E+00	1.599E+01	1.218E+00	1.938E+01	4.811E-01
BC-05	PO	1.401E+01	1.573E-01	1.653E+01	8.979E-02	1.650E+01	8.979E-02
	SSA	1.401E+01	5.746E-02	1.649E+01	4.464E-02	1.649E+01	4.464E-02
	MFO	1.398E+01	1.235E-01	1.657E+01	4.698E-03	1.657E+01	4.698E-03
	PSO	1.394E+01	1.243E-01	**1.660E+01**	**3.114E-03**	**1.660E+01**	**3.114E-03**
	BA	**1.411E+01**	**8.599E-04**	1.658E+01	2.119E-02	1.658E+01	2.119E-02
	WOA	1.398E+01	5.788E-02	1.660E+01	3.247E-03	1.660E+01	3.247E-03
BC-06	PO	**1.499E+01**	**4.021E-01**	1.617E+01	1.166E+00	2.113E+01	2.683E+00
	SSA	1.455E+01	1.089E+00	1.605E+01	7.916E-01	2.116E+01	1.346E+00
	MFO	1.473E+01	1.393E+00	1.662E+01	1.321E+00	2.161E+01	**3.600E-01**
	PSO	1.430E+01	5.469E-01	1.609E+01	5.944E-01	2.161E+01	4.574E-01
	BA	1.454E+01	7.583E-01	1.688E+01	4.189E-01	2.144E+01	6.603E-01
	WOA	1.478E+01	6.874E-01	**1.707E+01**	**3.739E-01**	**2.182E+01**	6.595E-01

Table 22. The FSIM comparison results

IMAGE		2 thresholds		4 thresholds		10 thresholds	
		Mean	STD	Mean	STD	Mean	STD
BC-01	PO	**7.334E-01**	9.058E-04	8.100E-01	1.664E-03	9.375E-01	6.902E-05
	SSA	7.299E-01	**5.268E-04**	8.129E-01	3.412E-03	9.337E-01	3.879E-04
	MFO	6.844E-01	7.197E-03	8.241E-01	7.058E-04	9.436E-01	1.048E-05
	PSO	6.510E-01	3.581E-03	**8.543E-01**	**2.286E-04**	9.433E-01	**9.781E-06**
	BA	6.756E-01	4.256E-02	8.188E-01	3.004E-03	**9.506E-01**	7.026E-05
	WOA	6.828E-01	7.372E-03	8.420E-01	1.343E-03	9.371E-01	2.896E-05
BC-02	PO	6.654E-01	6.141E-03	8.216E-01	9.448E-04	9.182E-01	1.857E-04
	SSA	6.632E-01	2.867E-03	8.000E-01	2.512E-03	9.177E-01	3.615E-04
	MFO	6.731E-01	1.772E-03	8.205E-01	5.943E-04	9.280E-01	**7.489E-05**
	PSO	**6.935E-01**	**8.595E-04**	8.130E-01	1.761E-03	9.324E-01	1.144E-04
	BA	6.688E-01	1.003E-03	8.232E-01	**2.941E-04**	**9.375E-01**	7.797E-05
	WOA	6.476E-01	7.794E-03	**8.314E-01**	4.138E-04	9.072E-01	5.254E-04
BC-03	PO	5.064E-01	1.080E-03	6.957E-01	1.005E-03	8.095E-01	5.452E-03
	SSA	5.251E-01	**3.195E-04**	**7.017E-01**	**5.798E-04**	8.486E-01	5.561E-04
	MFO	4.998E-01	1.499E-03	6.703E-01	2.446E-03	8.738E-01	3.064E-04
	PSO	5.279E-01	2.169E-03	6.741E-01	1.441E-03	**8.770E-01**	1.476E-04
	BA	**5.638E-01**	3.594E-03	6.765E-01	8.855E-04	8.741E-01	**9.361E-05**
	WOA	5.169E-01	7.779E-04	6.912E-01	1.100E-03	8.516E-01	1.119E-03
BC-04	PO	7.080E-01	3.258E-03	7.533E-01	2.689E-03	8.336E-01	1.204E-03
	SSA	**7.203E-01**	**1.814E-04**	**7.855E-01**	**1.378E-03**	8.587E-01	3.198E-03
	MFO	7.078E-01	2.053E-03	7.730E-01	1.423E-03	8.560E-01	**2.844E-04**
	PSO	7.066E-01	1.096E-03	7.799E-01	1.232E-03	8.929E-01	8.787E-04
	BA	7.076E-01	1.409E-03	7.427E-01	2.019E-03	**9.136E-01**	1.151E-03
	WOA	6.858E-01	2.379E-03	7.536E-01	1.909E-03	8.476E-01	8.598E-04
BC-05	PO	6.745E-01	2.613E-04	7.798E-01	1.067E-04	7.798E-01	1.067E-04
	SSA	6.785E-01	9.640E-05	7.807E-01	4.828E-05	7.807E-01	4.828E-05
	MFO	6.769E-01	2.044E-04	7.829E-01	7.520E-06	7.829E-01	7.520E-06
	PSO	6.757E-01	1.972E-04	7.844E-01	4.202E-06	7.844E-01	4.202E-06
	BA	**6.823E-01**	**1.890E-06**	7.834E-01	2.264E-05	7.834E-01	2.264E-05
	WOA	6.772E-01	1.136E-04	**7.843E-01**	**3.665E-06**	**7.843E-01**	**3.665E-06**
BC-06	PO	**6.257E-01**	**5.205E-04**	6.871E-01	1.006E-03	8.285E-01	3.053E-03
	SSA	6.161E-01	1.613E-03	6.804E-01	7.684E-04	8.556E-01	1.069E-03
	MFO	6.160E-01	1.407E-03	7.049E-01	2.033E-03	**8.682E-01**	**2.474E-04**
	PSO	6.033E-01	1.212E-03	6.835E-01	**6.954E-04**	8.665E-01	4.110E-04
	BA	6.006E-01	4.395E-03	**7.136E-01**	9.594E-04	8.627E-01	5.516E-04
	WOA	5.912E-01	7.749E-03	6.989E-01	1.655E-03	8.682E-01	4.419E-04

We employed different threshold levels (2, 4, and 10) and conducted a comparative analysis between PO and the five algorithms for segmenting the same set of six images. The results are presented in Table 20 – Table 22.

In the evaluation using PSNR (Table 20), the PO algorithm consistently exhibits competitive performance across diverse thresholds, frequently demonstrating comparable or superior mean values relative to alternative methodologies. Its robustness becomes evident through the consistently lowest standard deviation, signifying consistent performance across various iterations and image scales. SSA and WOA also reveal acceptable efficacy in certain cases.

In SSIM (Table 21), the PO algorithm consistently excels across a range of thresholds, particularly in mean values, underscoring its proficiency in preserving structural details and similarity between the segmented and original images. SSA and MFO methods also reveal good performance but are less consistent over different images and thresholds.

In the assessment of FSIM (Table 22), the PO algorithm sustains competitive mean scores and relatively reduced standard deviations across different thresholds, highlighting its competence in preserving image features compared to rival algorithms. SSA and MFO also expose competitive results but with more variability over various images and thresholds.

Overall, the PO algorithm showcases consistent and competitive performance across the assessed metrics and threshold variations, positioning it as a robust option for image segmentation tasks. Nevertheless, its effectiveness may be contingent upon specific image characteristics and the thresholds employed in the segmentation process.

It is evident that PO, in overall, surpasses the other five algorithms in terms of both mean and standard deviation across almost all the images at various threshold levels. MFO, PSO, and BA expose moderate to good efficacy but might lag behind PO, SSA, and WOA in terms of consistency or average efficacy across multiple images and metrics. The statistical analysis repeats the competitiveness of the proposed PO methods when dealing with diverse segmentation accuracy requirements, especially in solving low threshold problems.

4 Conclusions and future works

In conclusion, the parrot optimizer (PO) introduced in this paper stands out as an efficient algorithm. With its efficient on-the-fly architecture, PO adeptly balances exploration and exploitation, steering clear of local optima without distinct phases. Our comprehensive experiments involved rigorous comparisons with eight well-established algorithms across classical and IEEE CEC 2022 test functions. PO consistently emerged as the top-performing algorithm, demonstrating exceptional exploration-exploitation balance. Across five selected performance metrics, PO consistently excelled or maintained stability. A subsequent parameter sensitivity analysis highlighted the influence of PO's parameters on optimization results, emphasizing the potential for enhancing its capabilities through thoughtful parameter adjustments.

Moreover, PO exhibited remarkable real-world applicability by successfully tackling five optimization problems, especially in the medical field. Three key factors contribute to its robust performance: efficient exploration and exploitation through stochastic states, a multi-strategy search approach ensuring algorithmic stochasticity and population diversity, and incorporating four comprehensive strategies inspired by domesticated Pyrrhura Molinae's behaviors.

Despite PO's impressive potential, future research should focus on determining the optimal parameter ratios, presenting an ongoing challenge. Additionally, exploring hybrid approaches by combining PO with established metaheuristics holds promise for creating more potent and versatile optimization techniques. These endeavors may open new possibilities and significantly advance the optimization field.

CRediT authorship contribution statement

Junbo Lian: Investigation, Resources, Conceptualization, Methodology, Software, Data curation, Funding acquisition, Formal analysis, Visualization, Writing-original draft.

Guohua Hui: Writing - review & editing, Funding acquisition, Supervision, Project administration.

Ling Ma: Writing - review & editing, Data curation, Software, Validation.

Ting Zhu: Writing - review & editing, Data curation, Software, Validation.

Xincan Wu: Writing - review & editing, Visualization, Software.

Ali Asghar Heidari: Formal analysis, Visualization, Resources, Software, Investigation, Supervision, Writing - review & editing.

Yi Chen: Writing - review & editing, Methodology, Software, Visualization.

Huiling Chen: Formal analysis, Resources, Investigation, Methodology, Supervision, Writing - review & editing.

Acknowledgments

This research is financially supported by the Scientific Research Project of Zhejiang Province (Grant No. 2019C02075, LGG19F010012), the Natural Science Foundation of Zhejiang Province (Grant No. LZ22F020005), National College Student Research Programme (Grant No. 202310341080X), College Student Research Programme of Zhejiang Province (Grant No.2023R412012) and Zhejiang A&F University, National Natural Science Foundation of China (Grant No. 62076185).

Data availability

Data will be made available on request.

Declaration of AI and AI-assisted technologies in the writing process

During the revision stage of this work the author(s) used ChatGPT as a grammar checker in order to double check and proofread the English of the paper. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

References

[1] R. Li, X. Wu, H. Tian, N. Yu, and C. Wang, "Hybrid memetic pretrained factor analysis-based deep belief networks for transient electromagnetic inversion," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 60, pp. 1-20, 2022.

[2] S. Li and Z. Geng, "Bicriteria scheduling on an unbounded parallel-batch machine for minimizing makespan and maximum cost," *Information Processing Letters,* vol. 180, p. 106343, 2023.

[3] H. Wang, Q. Gao, H. Li, H. Wang, L. Yan, and G. Liu, "A structural evolution-based anomaly detection method for generalized evolving social networks," *The Computer Journal,* vol. 65, no. 5, pp. 1189-1199, 2022.

[4] A. Qi *et al.*, "Directional mutation and crossover boosted ant colony optimization with application to COVID-19 X-ray image segmentation," *Computers in biology and medicine,* vol. 148, p. 105810, 2022.

[5] J. Xia *et al.*, "Evolving kernel extreme learning machine for medical diagnosis via a disperse foraging sine cosine algorithm," *Computers in Biology and Medicine,* vol. 141, p. 105137, 2022.

[6] K. Tafakkori, R. Tavakkoli-Moghaddam, and A. Siadat, "Sustainable negotiation-based nesting and scheduling in additive manufacturing systems: A case study and multi-objective meta-heuristic algorithms," *Engineering Applications of Artificial Intelligence,* vol. 112, p. 104836, 2022.

[7] J. Lian *et al.*, "Visualized pattern recognition optimization for apple mechanical damage by laser relaxation spectroscopy," *International Journal of Food Properties,* vol. 26, no. 1, pp. 1566-1578, 2023.

[8] W. Zhou, J. Lian, J. Zhang, Z. Mei, Y. Gao, and G. Hui, "Tomato storage quality predicting method based on portable electronic nose system combined with WOA-SVM model," *Journal of Food Measurement and Characterization,* pp. 1-11, 2023.

[9] G. Divsalar, A. Divsalar, A. Jabbarzadeh, and H. Sahebi, "An optimization approach for green tourist trip design," *Soft Computing,* vol. 26, no. 9, pp. 4303-4332, 2022.

[10] X. Shen, S.-C. Du, Y.-N. Sun, P. Z. Sun, R. Law, and E. Q. Wu, "Advance scheduling for chronic care under online or offline revisit uncertainty," *IEEE Transactions on Automation Science and Engineering,* 2023.

[11] E. Van Den Berg and M. P. Friedlander, "Probing the Pareto frontier for basis pursuit solutions," *Siam journal on scientific computing,* vol. 31, no. 2, pp. 890-912, 2009.

[12] W. Rodi and R. L. Mackie, "Nonlinear conjugate gradients algorithm for 2-D magnetotelluric inversion," *Geophysics,* vol. 66, no. 1, pp. 174-187, 2001.

[13] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science,* vol. 313, no. 5786, pp. 504-507, 2006.

[14] B. Cao *et al.*, "Large-scale many-objective deployment optimization of edge servers," *IEEE Transactions on Intelligent Transportation Systems,* vol. 22, no. 6, pp. 3841-3849, 2021.

[15] B. Cao, J. Zhao, Z. Lv, and P. Yang, "Diversified personalized recommendation optimization based on mobile data," *IEEE Transactions on Intelligent Transportation Systems,* vol. 22, no. 4, pp. 2133-2139, 2020.

[16] R. Dong, H. Chen, A. A. Heidari, H. Turabieh, M. Mafarja, and S. Wang, "Boosted kernel search: Framework, analysis and case studies on the economic emission dispatch problem," *Knowledge-Based Systems,* vol. 233, p. 107529, 2021.

[17] T. Guo, H. Yuan, L. Wang, and T. Wang, "Rate-distortion optimized quantization for geometry-based point cloud

compression," *Journal of Electronic Imaging,* vol. 32, no. 1, pp. 013047-013047, 2023.

[18] N. Zeng, H. Qiu, Z. Wang, W. Liu, H. Zhang, and Y. Li, "A new switching-delayed-PSO-based optimized SVM algorithm for diagnosis of Alzheimer's disease," *Neurocomputing,* vol. 320, pp. 195-202, 2018.

[19] N. Zeng, Z. Wang, W. Liu, H. Zhang, K. Hone, and X. Liu, "A dynamic neighborhood-based switching particle swarm optimization algorithm," *IEEE Transactions on Cybernetics,* vol. 52, no. 9, pp. 9290-9301, 2020.

[20] I. Ahmadianfar, A. A. Heidari, A. H. Gandomi, X. Chu, and H. Chen, "RUN beyond the metaphor: an efficient optimization algorithm based on Runge Kutta method," *Expert Systems with Applications,* vol. 181, p. 115079, 2021.

[21] I. Ahmadianfar, A. A. Heidari, S. Noshadian, H. Chen, and A. H. Gandomi, "INFO: An efficient optimization algorithm based on weighted mean of vectors," *Expert Systems with Applications,* vol. 195, p. 116516, 2022.

[22] C. Zhang, L. Zhou, and Y. Li, "Pareto Optimal Reconfiguration Planning and Distributed Parallel Motion Control of Mobile Modular Robots," *IEEE Transactions on Industrial Electronics,* pp. 1-10, 2023, doi: 10.1109/TIE.2023.3321997.

[23] V. Kumar, J. K. Chhabra, and D. Kumar, "Parameter adaptive harmony search algorithm for unimodal and multimodal optimization problems," *Journal of Computational Science,* vol. 5, no. 2, pp. 144-155, 2014.

[24] C. Wang, J. M. Koh, T. Yu, N. G. Xie, and K. H. Cheong, "Material and shape optimization of bi-directional functionally graded plates by GIGA and an improved multi-objective particle swarm optimization algorithm," *Computer Methods in Applied Mechanics and Engineering,* vol. 366, p. 113017, 2020.

[25] W. Zhao, C. Du, and S. Jiang, "An adaptive multiscale approach for identifying multiple flaws based on XFEM and a discrete artificial fish swarm algorithm," *Computer Methods in Applied Mechanics and Engineering,* vol. 339, pp. 341-357, 2018.

[26] H. Chen, C. Li, M. Mafarja, A. A. Heidari, Y. Chen, and Z. Cai, "Slime mould algorithm: A comprehensive review of recent variants and applications," *International Journal of Systems Science,* vol. 54, no. 1, pp. 204-235, 2023.

[27] H. Li, J. Li, P. Wu, Y. You, and N. Zeng, "A ranking-system-based switching particle swarm optimizer with dynamic learning strategies," *Neurocomputing,* vol. 494, pp. 356-367, 2022.

[28] E. K. Burke, E. K. Burke, G. Kendall, and G. Kendall, *Search methodologies: introductory tutorials in optimization and decision support techniques*. Springer, 2014.

[29] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization,* vol. 11, pp. 341-359, 1997.

[30] E. Mezura-Montes and C. A. C. Coello, "A simple multimembered evolution strategy to solve constrained optimization problems," *IEEE Transactions on Evolutionary computation,* vol. 9, no. 1, pp. 1-17, 2005.

[31] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary computation,* vol. 3, no. 2, pp. 82-102, 1999.

[32] L. Ren *et al.*, "Shale gas load recovery modeling and analysis after hydraulic fracturing based on genetic expression programming: A case study of southern Sichuan Basin shale," *Journal of Natural Gas Science and Engineering,* vol. 107, p. 104778, 2022.

[33] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, 1995, vol. 4: IEEE, pp. 1942-1948.

[34] Z. Xiao *et al.*, "Multi-objective parallel task offloading and content caching in D2D-aided MEC networks," *IEEE Transactions on Mobile Computing,* 2022.

[35] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Generation Computer Systems,* vol. 97, pp. 849-872, 2019/08/01/ 2019, doi: https://doi.org/10.1016/j.future.2019.02.028.

[36] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Generation Computer Systems,* vol. 111, pp. 300-323, 2020/04/03/ 2020, doi: https://doi.org/10.1016/j.future.2020.03.055.

[37] Y. Yang, H. Chen, A. A. Heidari, and A. H. Gandomi, "Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts," *Expert Systems with Applications,* vol. 177, p. 114864, 2021.

[38] C. Blum, "Ant colony optimization: Introduction and recent trends," *Physics of Life reviews,* vol. 2, no. 4, pp. 353-373, 2005.

[39] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE computational intelligence magazine,* vol. 1, no.

4, pp. 28-39, 2006.

[40] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Technical report-tr06, Erciyes university, engineering faculty, computer …, 2005.

[41] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowledge-based systems,* vol. 89, pp. 228-249, 2015.

[42] X. S. Yang and A. Hossein Gandomi, "Bat algorithm: a novel approach for global engineering optimization," *Engineering computations,* vol. 29, no. 5, pp. 464-483, 2012.

[43] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information sciences,* vol. 179, no. 13, pp. 2232-2248, 2009.

[44] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: a nature-inspired algorithm for global optimization," *Neural Computing and Applications,* vol. 27, pp. 495-513, 2016.

[45] H. Su *et al.*, "RIME: A physics-based optimization," *Neurocomputing,* vol. 532, pp. 183-214, 2023.

[46] J. Lian and G. Hui, "Human Evolutionary Optimization Algorithm," *Expert Systems with Applications,* p. 122638, 2023.

[47] E. Atashpaz-Gargari and C. Lucas, "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition," in *2007 IEEE congress on evolutionary computation*, 2007: Ieee, pp. 4661-4667.

[48] R. V. Rao, V. J. Savsani, and D. Vakharia, "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-aided design,* vol. 43, no. 3, pp. 303-315, 2011.

[49] S. C. Satapathy and A. Naik, "Modified Teaching–Learning-Based Optimization algorithm for global numerical optimization—A comparative study," *Swarm and Evolutionary Computation,* vol. 16, pp. 28-37, 2014.

[50] S. P. Adam, S.-A. N. Alexandropoulos, P. M. Pardalos, and M. N. Vrahatis, "No free lunch theorem: A review," *Approximation and optimization: Algorithms, complexity and applications,* pp. 57-82, 2019.

[51] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE transactions on evolutionary computation,* vol. 1, no. 1, pp. 67-82, 1997.

[52] P. N. Suganthan *et al.*, "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," *KanGAL report,* vol. 2005005, no. 2005, p. 2005, 2005.

[53] J. J. Liang, B. Y. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization," *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore,* vol. 635, no. 2, 2013.

[54] J. Ragusa-Netto, "Feeding ecology of the Green-cheeked parakeet (Pyrrhura molinae) in dry forests in western Brazil," *Brazilian Journal of Biology,* vol. 67, pp. 243-249, 2007.

[55] C. S. van Hoek and C. Ten Cate, "Abnormal behavior in caged birds kept as pets," *Journal of Applied Animal Welfare Science,* vol. 1, no. 1, pp. 51-64, 1998.

[56] R. Nos and M. C. Llull, "Conducta de alimentación de cinco especies de cotorritas (Aves, Psittacidae)," *Miscel· lània Zoològica,* pp. 245-252, 1984.

[57] A. Tygesen and B. Forkman, "The Parrot–Owner Relationship and Problem Behaviors in Parrots," *Anthrozoös,* pp. 1-13, 2023.

[58] D. G. Homberger, "Classification and the status of wild populations of parrots," *Manual of parrot behavior,* pp. 3-11, 2006.

[59] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-based systems,* vol. 96, pp. 120-133, 2016.

[60] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in engineering software,* vol. 69, pp. 46-61, 2014.

[61] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: theory and application," *Advances in Engineering Software,* vol. 105, pp. 30-47, 2017.

[62] L. Liu *et al.*, "Performance optimization of differential evolution with slime mould algorithm for multilevel breast cancer image segmentation," *Computers in Biology and Medicine,* vol. 138, p. 104910, 2021.

[63] J. Xia *et al.*, "Performance optimization of support vector machine with oppositional grasshopper optimization for acute appendicitis diagnosis," *Computers in Biology and Medicine,* vol. 143, p. 105206, 2022.

[64] S. Zhao *et al.*, "Multilevel threshold image segmentation with diffusion association slime mould algorithm and Renyi's

entropy for chronic obstructive pulmonary disease," *Computers in Biology and Medicine,* vol. 134, p. 104427, 2021.

[65] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in engineering software,* vol. 95, pp. 51-67, 2016.

[66] H. Jia, X. Peng, and C. Lang, "Remora optimization algorithm," *Expert Systems with Applications,* vol. 185, p. 115665, 2021.

[67] M. Azizi, S. Talatahari, and A. H. Gandomi, "Fire Hawk Optimizer: A novel metaheuristic algorithm," *Artificial Intelligence Review,* vol. 56, no. 1, pp. 287-363, 2023.

[68] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The arithmetic optimization algorithm," *Computer methods in applied mechanics and engineering,* vol. 376, p. 113609, 2021.

[69] A. Kumar, G. Wu, M. Z. Ali, R. Mallipeddi, P. N. Suganthan, and S. Das, "A test-suite of non-convex constrained optimization problems from the real-world and some baseline results," *Swarm and Evolutionary Computation,* vol. 56, p. 100693, 2020.

[70] Q. He and L. Wang, "An effective co-evolutionary particle swarm optimization for constrained engineering design problems," *Engineering applications of artificial intelligence,* vol. 20, no. 1, pp. 89-99, 2007.

[71] J. S. Arora, *Introduction to optimum design*. Elsevier, 2004.

[72] A. Kaveh and M. Khayatazad, "A new meta-heuristic method: ray optimization," *Computers & structures,* vol. 112, pp. 283-294, 2012.

[73] C. A. C. Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Computers in Industry,* vol. 41, no. 2, pp. 113-127, 2000.

[74] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," *Advances in engineering software,* vol. 114, pp. 163-191, 2017.

[75] A. W. Mohamed, "A novel differential evolution algorithm for solving constrained engineering optimization problems," *Journal of Intelligent Manufacturing,* vol. 29, pp. 659-692, 2018.

[76] B. Adarsh, T. Raghunathan, T. Jayabarathi, and X.-S. Yang, "Economic dispatch using chaotic bat algorithm," *Energy,* vol. 96, pp. 666-675, 2016.

[77] H. Eskandar, A. Sadollah, A. Bahreininejad, and M. Hamdi, "Water cycle algorithm–A novel metaheuristic optimization method for solving constrained engineering optimization problems," *Computers & Structures,* vol. 110, pp. 151-166, 2012.

[78] P. Savsani and V. Savsani, "Passing vehicle search (PVS): A novel metaheuristic algorithm," *Applied Mathematical Modelling,* vol. 40, no. 5-6, pp. 3951-3978, 2016.

[79] B. He *et al.*, "A new method for CTC images recognition based on machine learning," *Frontiers in Bioengineering and Biotechnology,* vol. 8, p. 897, 2020.

[80] Y. Zhuang, N. Jiang, and Y. Xu, "Progressive Distributed and Parallel Similarity Retrieval of Large CT Image Sequences in Mobile Telemedicine Networks," *Wireless Communications and Mobile Computing,* vol. 2022, p. 6458350, 2022/07/14 2022, doi: 10.1155/2022/6458350.

[81] K. Jin *et al.*, "MSHF: A Multi-Source Heterogeneous Fundus (MSHF) Dataset for Image Quality Assessment," *Scientific Data,* vol. 10, no. 1, p. 286, 2023.

[82] Y. Zhu, R. Huang, Z. Wu, S. Song, L. Cheng, and R. Zhu, "Deep learning-based predictive identification of neural stem cell differentiation," *Nature communications,* vol. 12, no. 1, p. 2614, 2021.

[83] G. Aresta *et al.*, "Bach: Grand challenge on breast cancer histology images," *Medical image analysis,* vol. 56, pp. 122-139, 2019.

[84] D. Oliva, S. Hinojosa, V. Osuna-Enciso, E. Cuevas, M. Pérez-Cisneros, and G. Sanchez-Ante, "Image segmentation by minimum cross entropy using evolutionary methods," *Soft Computing,* vol. 23, pp. 431-450, 2019.

[85] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *2010 20th international conference on pattern recognition*, 2010: IEEE, pp. 2366-2369.

[86] W. Zhou, "Image quality assessment: from error measurement to structural similarity," *IEEE transactions on image processing,* vol. 13, pp. 600-613, 2004.

[87] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE transactions on Image Processing,* vol. 20, no. 8, pp. 2378-2386, 2011.