Knowledge-Based Systems 235 (2022) 107603

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Fractional-order comprehensive learning marine predators algorithm for global optimization and feature selection

Dalia Yousri [a], Mohamed Abd Elaziz [b,g,h], Diego Oliva [c,*], Ajith Abraham [d], Majed A. Alotaibi [e], Md Alamgir Hossain [f]

[a] Department of Electrical Engineering, Faculty of Engineering, Fayoum University, Fayoum, Egypt
[b] Department of Mathematics, Faculty of Science, Zagazig University, Zagazig 44519, Egypt
[c] Depto. de Ciencias Computacionales, Universidad de Guadalajara, CUCEI, Av. Revolución 1500, Guadalajara, Jal, Mexico
[d] Machine Intelligence Research Labs (MIR Labs) Scientific Network for Innovation and Research Excellence 11, 3rd Street NW, P.O. Box 2259, Auburn, Washington 98071, USA
[e] Department of Electrical Engineering, College of Engineering, King Saud University, Riyadh 11421, Saudi Arabia
[f] Queensland Micro- and Nanotechnology Centre, Griffith University, Nathan 4111, Australia
[g] Artificial Intelligence Research Center (AIRC), Ajman University, Ajman 346, United Arab Emirates
[h] Computer Science & Engineering, Galala University, Galala, Egypt

ARTICLE INFO

Article history:
Received 26 September 2020
Received in revised form 30 March 2021
Accepted 13 October 2021
Available online 21 October 2021

Keywords:
Marine Predator Algorithm (MPA)
Fractional-order
Comprehensive learning
Global optimization
Engineering problem
Feature selection

ABSTRACT

The topological structure of the search agents in the swarm is a key factor in diversifying the knowledge between the population and balancing the designs of the exploration and intensification stages. Marine Predator Algorithm (MPA) is a recently introduced algorithm that mimics the interaction between the prey and predator in ocean. MPA has a vital issue in its structure. This drawback related to the number of iterations that is divided into the algorithm phases, hence the agents do not have the adequate number of tries to discover the search landscape and exploit the optimal solutions. This situation affects the search process. Therefore, in this paper, the principle of the comprehensive learning strategy and memory perspective of the fractional calculus have been incorporated into MPA. They help to achieve an efficient sharing for the best knowledge and the historical experiences between the agents with the aim of escaping from the local solutions and avoiding the immature convergence. The developed fractional-order comprehensive learning MPA (FOCLMPA) has been examined with several multidimensional benchmarks from the CEC2017 and CEC2020 as challenging tested functions in the numerical validation part. For real-world applications, four engineering problems have been employed and a set of eighteen UCI datasets have been used to demonstrate the developed performance for feature selection optimization problem. The FOCLMPA has been compared with several well-regarded optimization algorithms via numerous statistical and non-parametric analyses to provide unbiased recommendation. The comparisons confirm the superiority and stability of FOCLMPA in handling the series of experiments with high qualified results and remarkable convergence curves.

1. Introduction

Meta-heuristic algorithms (MA) have been successfully applied for solving complex optimization problems in different domains. MA are part of artificial intelligence, and they are inspired by natural processes that permit to handle the difficulties of the search spaces. MA also contain internal rules that permit to test, improve, and/or discard the candidate solutions. The power of MA is focused on the operators that modify the position of the solutions at each iteration. According to their operators, MA can be classified into two groups called Swarm Algorithms (SA) and Evolutionary Algorithms (EA) [1]. The group of SA is inspired in the collective behaviour of the species, for example, the Particle Swarm Optimization (PSO) [2] that is based on the movement of bird flocks or fish schools. The group of EA takes as a base the evolution theory to formulate new movements; it is possible to find crossover, mutation, and selection. Some examples of EA are the Genetic Algorithms (GA) [3] or the Differential Evolution (DE) [4]. Both SA and EA can find the best solutions depending on the problems. However, MA are governed by the No-Free-Lunch (NFL) theorem [5], it states that there is no algorithm able to solve all the optimization problems, and one algorithm could be

* Corresponding author.
E-mail addresses: day01@fayoum.edu.eg (D. Yousri), abd_el_aziz_m@yahoo.com (M. Abd Elaziz), diego.oliva@cucei.udg.mx (D. Oliva), ajith.abraham@ieee.org (A. Abraham), majedalotaibi@ksu.edu.sa (M.A. Alotaibi), mdalamgir.hossain@griffith.edu.au (M.A. Hossain).

https://doi.org/10.1016/j.knosys.2021.107603
0950-7051/

better than other only if it is designed for a specific problem [6]. Considering such facts, it is essential to develop new methods and strategies that must be adaptive.

Marine Predators Algorithm (MPA) is a recently proposed swarm method that emulates the social behaviour of marine predators [7]. Since MPA is a new SA, its performance has been tested in mathematical benchmark functions and engineering optimization problems. MPA proved its quality due to its simplicity; it also has easy-tuneable parameters and flexibility in the implementation. Nonetheless, the implemented strategy in the MPA is to divide the number of iterations among the algorithm's main processes; this could affect the search process by causing the population to get stuck in a local solutions. This occurs because the candidate solutions do not have enough iterations to find optimal solutions. Therefore, similar to other MA, the MPA has been modified and adapted for solving real problems. An example where MPA has been successfully applied is in image processing; in it, two improved versions have been proposed for the segmentation of COVID-19 medical images [8,9]. Following this domain, the MPA has also been used for forecasting confirmed cases of COVID-19 in different countries [10]. Another interesting enhanced implementation of MPA is for task scheduling in fog computing [11]. In solar energy, the MPA has been employed for large scale photovoltaic array reconfiguration [12] solid oxide fuel cell modelling [13] and for photovoltaic modelling [14].

Motivated by the facts previously explained and NFL theorem [5], this article introduces an enhanced version of the MPA to improve this algorithm's performance in different optimization problems. The topological structure of the swarm search agents plays a vital role in diversifying the knowledge between the population and balancing the exploration and intensification stages. Therefore, modifying the MPA population topological structure can enhance the search process's diversity to avoid the local solutions and exploit the optimal ones. The Comprehensive Learning strategy (CL) [15] is one of the efficient approaches employed to boost and handle the search during the optimization process. The search agents' current positions can learn from previous best locations of the population landscape (exemplars). The selection of the exemplars is performed by a probability coefficient that guarantees the diversity of the population. The use of CL then permits that the search agents discover the optimal solutions avoiding premature convergence. With this strategy, the algorithms could successfully deal with multi-modal and nonlinear optimization problems [16]. Recently, a robust mathematical tool named fractional calculus (FC) has been merged with optimization algorithms to boost the performance of the search agents by including a memory window (called historical terms or memory) into the processes [17,18]. This advantage permits us to obtain more realistic models than integer-based approaches.

The benefits of the CL and FC mechanisms have been studied in benchmark problems and real world applications. Regarding the CL was firstly introduced to enhance the PSO, the method is called CLPSO [15]. A modified version of the CLPSO was proposed in [19] for the operation of multi-reservoir hydropower systems. Also, the hybrid of heterogeneous comprehensive learning strategy and Pigeon-Inspired Optimization (HCLPIO) employs the CL and it is used to model control systems [20]. Another interesting approach was proposed by Yousri et al. [21] called Chaotic Heterogeneous Comprehensive Learning Particle Swarm Optimization for the identification of static and dynamic photovoltaic models' parameters. For the FC also exist a wide range of applications and proposals; for example, the Fractional Lévy Flight Bat Algorithm (FLFBA) that was introduced for solving global optimization problems [22]. In this domain, it has also been proposed the Fractional Order Cuckoo Search (FO-CS) algorithm to identify the parameters of the chaos phenomena in the economic-financial system [23]. Moreover, the fractional-order flower pollination algorithm (FO-FPA) has been proposed for global optimization and image segmentation [24]. The Fractional Order Firefly (FO-FF) is another interesting approach that involves the use of FC to modify an optimization algorithm in order to provide better results [25]. In this case, the FO-FF has been tested for the parameter estimation of chaotic systems. In the field of deep learning the combination of FC with MA, the Cat Swarm Fractional Calculus Optimization (CSFCO) algorithm then is proposed for removing the artifacts present in the EEG signal by using deep learning [26]. Recently, the fractional order Grunwald–Letnikov definition has been merged with the Manta ray foraging optimizer in [27] for image segmentation optimization problem.

According to the FC and CL approach's above-mentioned properties, we were motivated to provide a modified MPA version based on those concepts to enhance the algorithm's performance. The improved MPA is called fractional-order comprehensive learning marine predator optimizer (FOCLMPA). In this method the FC has been combined with MPA during the exploitation phase where the heredity and non-locality properties of the FC operator has adopted to model the aftereffect of the previous locations of marine predators on their future movement directions. Moreover, the CL has adopted in the exploration phase to boost the diversity of the search agents. The sensitivity of the proposed FOCLMPA for the derivative order has been investigated with CEC2017 problems. Besides, the quality and stability of the FOCLMPA with large dimension optimization problems have been assessed with 30 and 50 dimensional CEC2017 functions [28]. With recent challenging benchmark functions from the CEC2020, the proposed FOCLMPA is examined for two dimensional of 5, and 20 [29]. In the experiments the FOCLMPA was compared with similar approaches from the state-of-the-art and they are the basic Marine Predators Algorithm (MPA) [7], Artificial ecosystem-based optimization (AEO) [30], Slime mould algorithm (SMA) [31], Manta ray foraging optimization (MRFO) [32], Sine–cosine algorithm (SCA) [33], Comprehensive learning particle swarm optimizer (CLPSO) [15], as well as the basic Particle swarm optimizer (PSO) [34] and Differential Evolution with composite trial vector generation strategies (CODE) [35]. Besides, the proposed FOCLMPA has also been tested with benchmark engineering optimization problems. The experimental results and comparison over this kind of problems provide evidence of the good performance of the FOCLMPA. This fact is validated by statistical analysis and non-parametric tests. The idea of MA is that they can work over real problems. Thus, the proposed algorithm has also been implemented for Feature Selection (FS), as a complex data-mining problem. In FS, it is necessary to find the optimal subset that contains the relevant information of the entire dataset. By performing this task, it is possible to remove redundant or non-important information. In the FS experiments, they were used eighteen datasets from the UCI repository. The results of the FOCLMPA for FS were compared with different MA to prove that the proposed algorithm can be used in real implementations.

The following points summarize the main contributions of this paper:

- An alternative global optimization and feature selection method is developed that depends on modified the new MA technique named Marine predators algorithm (MPA). Since the traditional MPA still suffers from some limitations that influence its convergence towards the global solution and lead to the degradation of the final solution's quality, mainly when applied to real-world applications.
- A novel variant of the MPA is proposed via using the CL approach to modify its exploration phase to preserve the

swarm particles' diversity. Besides, adopting the FC perspective to enhance the exploitation stage of the MPA through providing it with suitable memory. As a result, Fractional-Order Comprehensive Learning Marine Predators Algorithm (FOCLMPA) is proposed and examined with several series of experiments.
- A set of most challenging test functions of the CEC2017 and CEC2020 is applied with various numerical validation dimensions. The considered numerical functions are used to assess the optimization method's exploration and exploitation abilities during the searching process for global optimization.
- The FOCLMPA has been implemented for real-world optimization problems, including engineering applications and feature selection. Each one of these applications has its target and characteristics. For example, FS aims to reduce the number of features by removing the irrelevant features that affect the quality of classification and the decision-making process. The FS problem is employed as another challenging optimization problem; especially, when the number of features N increased, the search domain increased (2^N).

Finally, several statistical and non-parametric tests are performed over the series of the experiments to provide unbiased comparison aiming to offer an alternative tool for complex optimization problems.

The remainder sections are organized as follows: Section 2 presents the structure of the FOCLMPA and considers the basics of MPA, FC and CL. The Section 3 provides the simulation and experimental results for benchmark problems. Section 3.5 introduces the implementation of the FOCLMPA for FS. Finally Section 4 includes some conclusion and future works.

2. Structure of fractional-order with comprehensive learning marine predators algorithm

This section introduces the basic concepts of the fractional-order with comprehensive learning Marine Predators Algorithm (FOCLMPA). The following subsection describes the structure of the proposed algorithm.

2.1. Marine predators algorithm

The marine predators algorithm (MPA) has been introduced as a novel MA optimizer in Ref. [7]. MPA takes its inspiration from trying the predators in catching the prey and searching for the food source. The algorithm starts by initializing a population of solutions. The initial solutions are identified arbitrarily based on the search space as shown in the following equations:

$$Z_{id} = LB_d + r_1 \times (UB_d - LB_d), i = 1, 2, \dots, n, d = 1, 2, 3, \dots, DIM \tag{1}$$

where, LB_j and UB_j exemplify the upper and the lower bounds for each jth variable of the optimized problem of dimension DIM, respectively. In the search domain, $r_1 \in [0, 1]$ is random number.

The prey and predator in MPA are the agents of searching, thus, there are two principal matrices defined as the best and the elite matrices. The matrices contain the best predators, and the prey locations as mathematically defined as follows:

$$Z = \begin{bmatrix} Z_{11} & Z_{12} & \dots & Z_{1DIM} \\ Z_{21} & Z_{22} & \dots & Z_{2DIM} \\ \dots & \dots & \dots & \dots \\ Z_{n1} & Z_{n2} & \dots & Z_{nDIM} \end{bmatrix}, Elite = \begin{bmatrix} Z^1_{11} & Z^1_{12} & \dots & Z^1_{1DIM} \\ Z^1_{21} & Z^1_{22} & \dots & Z^1_{2DIM} \\ \dots & \dots & \dots & \dots \\ Z^1_{n1} & Z^1_{n2} & \dots & Z^1_{nDIM} \end{bmatrix}, \tag{2}$$

The principal aim is to reach the optimal solutions, so the first random group of solutions can be modernized based on the algorithm's construction. The prey's ratio and the predator's speed is the most critical operator in transferring the algorithm from one phase to another. In MPA, the ratio with the large-speed is the noticed feature in the first stage. In contrast, the unity and the weak ratio are the remarkable properties for the second and the third phases. The main lines of each phase are reviewed as follows:

1. **Phase 1: diversification phase (high-velocity ratio)** This phase is for discovering the search space, also called the exploration stage. This step is performed for the first third of the total number of development (i.e., $\frac{1}{3}t_{max}$). The prey transfers more quickly to look for the food source, whereas the predator pauses without moving. Faramarzi et al. [7] ideally such stage based on the following form:
$$S_i = R_B \bigotimes (Elite_i - R_B \bigotimes Z_i), i = 1, 2, \dots, n \tag{3}$$
$$Z_i = Z_i + P.R \bigotimes S_i \tag{4}$$
where, $R \in [0, 1]$ and $P = 0.5$ is used as a vector containing uniform arbitrary numbers and a fixed constant. R_B is an arbitrary vector represents the Brownian movement. $\bigotimes$ identifies the element-wise multiplications procedure.
2. **Phase 2: Unit velocity ratio** Such a stage is a transferring phase from diversification into intensification. All individuals, including the preys and the predators move at the same speed to search for food sources. This stage is considered as the main phase of the whole algorithm whose implementation starts as $\frac{1}{3}t_{max} < t < \frac{2}{3}t_{max}$. The predator's finest schedule is a tail Brownian, whereas the prey uses the lévy flight through such a stage. Accordingly, Faramarzi et al. [7] partitioned the population into 2 parts and applied Eqs. (5)–(6) for modelling the first section and Eqs. (7)–(8) for the second section as follows:
$$S_i = R_L \bigotimes (Elite_i - R_L \bigotimes Z_i), i = 1, 2, \dots, n/2 \tag{5}$$
$$Z_i = Z_i + P.R \bigotimes S_i \tag{6}$$
where, R_L exemplifies the arbitrary numbers tails lévy distribution. Eqs. (5)–(6) are satisfied to the first part of the population that exemplifies the exploitation. Then the second part of the population can be modelled as follows:
$$S_i = R_B \bigotimes (R_B \bigotimes Elite_i - Z_i), i = n/2, \dots, n \tag{7}$$
$$Z_i = Elite_i + P.CF \bigotimes S_i, \tag{8}$$
$$CF = \left(1 - \frac{t}{t_{max}}\right)^{2\left(\frac{t}{t_{max}}\right)} \tag{9}$$
where, CF is a coefficient that controls the step size of transferring for the predator.
3. **Phase 3: Intensification (low-velocity ratio)** This phase is probably to be the final stage in the procedure of searching. The predator transfers more quickly than the prey, so it tails the lévy through the update. The stage is applied on the final part of the iterations ($t > \frac{2}{3}t_{max}$) defined as:
$$S_i = R_L \bigotimes (R_L \bigotimes Elite_i - Z_i), i = 1, 2, \dots, n \tag{10}$$
$$Z_i = Elite_i + P.CF \bigotimes S_i, \; CF = (1 - \frac{t}{t_{max}})^{2\frac{t}{t_{max}}} \tag{11}$$

4. **Eddy development and fish aggregating devices outcome (FADS)**
The framing milieu has a significant impact on the creature's behaviour. Faramarzi et al. [7] took other effects like the fish aggregating devices (FADs) or eddy formation for averting the local optima of the algorithm. This phase can be summarized and modelled mathematically as:

$$Z_i = \begin{cases} Z_i + CF[Z_{min} + R \bigotimes (Z_{max} - Z_{min})] \bigotimes U \\ \quad r_5 < FAD \\ Z_i + [FAD(1-r) + r](Z_{r1} - Z_{r2}) \\ \quad r_5 > FAD \end{cases} \tag{12}$$

In Eq. (12), $FAD = 0.2$, and U is a bilateral choice which is constructed through creating random solution, after that it is switched into bilateral threshold 0.2. $r \in [0, 1]$ designates a random number. r_1 and r_2 represent the preys list.
5. **Marine memory** Nautical predators register the location that they have high ability in foraging in their memory. They apply this task through keeping the optimal solutions in mind. Then these solutions are modernized after determining the best solutions.

Fig. 1 shows the structure of the MPA algorithm where it starts by generating a set of random solutions using Eq. (1). Those solutions are evaluated, and the best solution vector is determined. The initial set of the solutions are modified via passing through three phases across the number of iterations. For the first third part of the iterations, the agents use Eqs. (3)–(4) while discovering the search space. For the second third of the iterations, the populations are divided into two halves. The first half of the population follows lévy distribution while modifying its location, whereas the second is guided by Brownian movement Using Eqs. (5)–(6), (7)–(8), respectively. For the last part of the iteration numbers, the agents use Eqs. (10)–(11) while updating their locations to keep on a low-velocity ratio between the predator and prey. At each iteration, the FADS of Eq. (12) and local memory are recognized to avoid trapping to local solutions and enhancing the next iteration searching process.

2.2. Comprehensive learning concept

The concept of Comprehensive Learning (CL) depends on the conduct of sharing the information and experiences between individuals. Thus CL helps to have a good balance between the exploration and the exploitation [15,20,21]. For inhibiting the premature convergence, the diversity of the swarm must be preserved for in the multiple dimensions. This can be done by the agents' best experiences which learn to each agent, this can be called exemplar. According to learning probability values (P_{cl}), the exemplar can be chosen for each dimension. The exemplar for each dimension is selected based on learning probability values (P_{cl}). This probability has various values for different agents and can be calculated as follows [15]:

$$P_{Cl_i} = a + b\frac{(exp(10(i-1))/(ps-1)-1)}{(exp(10)-1)} \tag{13}$$

where N is the population size number, and a and b are fixed numbers that have values of 0.05 and 0.25, respectively [15].

Fig. 2 shows the mechanism of selection of exemplars regarding the CL concept. There is an arbitrary number compared with the stated probability P_{cl_i} for each agent's dimension (d). If this number is greater than P_{cl_i}, then the corresponding dimension of ith agent will learn from its own position for that dimension. On the other hand, if this random number is smaller than P_{cl_i}, then the corresponding dimension of ith agent is guided by another particle chosen regarding the corresponding fitness function where two agents are selected randomly, and the agent with a better fitness is selected for the corresponding dimension. For a minimization problem, the corresponding agent for the least fitness is the selected exemplar and the reverse for a maximization problem.

2.3. Fractional calculus concept

The Fractional calculus (FC) concept depends on several definitions like the Grunwald–Letnikov (GL) definition, and it can be mathematically formulated as [36]:

$$D^{\beta}(Z(t)) = \lim_{h\to 0} \frac{1}{h^{\beta}} \sum_{k=0}^{\infty} (-1)^k \binom{\beta}{k} Z(t-kh), \tag{14}$$

where

$$\binom{\beta}{k} = \frac{\Gamma(\beta+1)}{\Gamma(k+1)\Gamma(\beta-k+1)} = \frac{\beta(\beta-1)(\beta-2)\dots(\beta-k+1)}{k!}, \tag{15}$$

where, $D^{\beta}(x(t))$ exemplifies the Grunwald–Letnikov (GL) fractional derivative of order β. $\Gamma(t)$ identifies gamma function. In discrete-time implementation, GL definition of Eq. (14) can be discussed as:

$$D^{\beta}[Z(t)] = \frac{1}{T^{\beta}} \sum_{k=0}^{r} \frac{(-1)^k \Gamma(\beta+1) Z(t-kT)}{\Gamma(k+1)\Gamma(\beta-k+1)} \tag{16}$$

where, T is the sampling period and r corresponds to the number of terms from the memory. The symbol of β identifies the derivative order coefficient.

In particular, $\beta = 1$, the formula of Eq. (16) can be rewritten as:

$$D^1[Z(t+1)] = Z(t+1) - Z(t) \tag{17}$$

where $D^1[x(t)]$ is the difference between two tailed events.

2.4. Proposed fractional-order comprehensive learning Marine Predators Algorithm

In this section the MPA subjected to two modifications to improve the diversification and intensification abilities based on the CL approach and FC memory property.

- Modification 1: the first step of the modification focuses on the sharing phase of the best information among the particles. This modification plays a vital role in avoiding the local solutions and the guarantee the diversity of the swarm. Therefore, the exploration phase of MPA has been enhanced based on CL approach where the prey uses the best-experienced agents of the previous iteration while it discovering the search landscape. Accordingly, the phase 1 of Eq. (3) in the basic MPA can be reformulated as below:

$$\begin{aligned} &S_i = R_B \bigotimes \left(Elite_i - R_B \bigotimes Z_i + K \cdot rand(d)\right. \\ &\quad \left. \bigotimes \left(Z_{Exemplar_{f_i}} - Z_i\right)\right), i = 1, 2, \dots, n \\ &where \qquad K = \frac{a}{1 + exp^{\left(\frac{-\gamma \cdot t}{T}\right)}} + b \cdot \left(\frac{t}{T} - 1\right)^2 \\ &\quad t = 1, 2, 3, \dots .T \\ &Then \qquad Z_i = Z_i + P.R \bigotimes S_i \end{aligned} \tag{18}$$

where K is an acceleration coefficient that obeys for sigmoid function. The values of a,b, γ are selected as 1.5, 2.2 and 0.0001 to achieve a diversification for the particles, respectively.

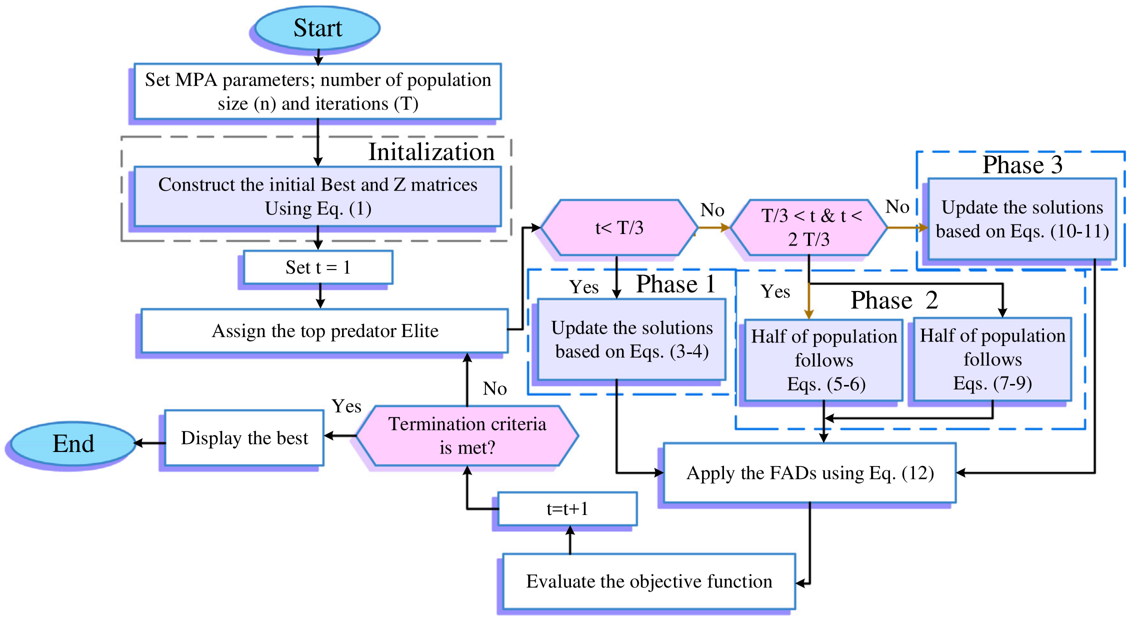

Fig. 1. Flowchart of MPA technique.

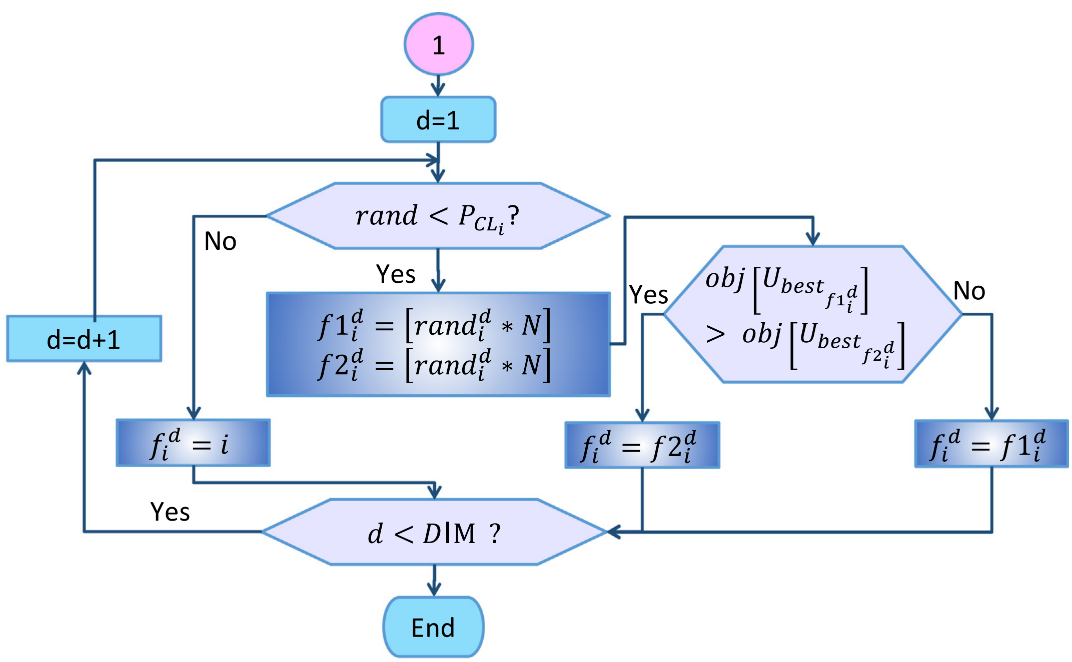

Fig. 2. Flowchart of CL concept..

- Modification 2: the second step of the modification exposes integrating the memory perspective of FC into the CLMPA to boost the second phase of the standard version of the algorithm. Hence, the FC memory is applied during updating the prey location in the second step of the algorithm to enhance the exploitation stage. The details of the applied approach have been explained in detail as below:
The motion of the prey of Eq. (6) can be reformulated to meet the special case of GL definition of Eq. (16) as follows.

$$Z_i(t+1) - Z_i(t) = P.R \bigotimes S_i \tag{19}$$

For general case based on the FC definition, the Eq. (20) can be written as follows:

$$D^{\delta}\left[Z_i(t+1)\right] = P.R \bigotimes S_i \tag{20}$$

By using the discrete form of GL definition of Eq. (16) at $T = 1$, the expression of Eq. (20) can be written as follows:

$$D^{\delta}\left[Z_i(t+1)\right] = Z_i(t+1) + \sum_{k=1}^{m} \frac{(-1)^k \Gamma(\delta+1) Z_i(t+1-k)}{\Gamma(k+1)\Gamma(\delta-k+1)} = P.R \bigotimes S_i. \tag{21}$$

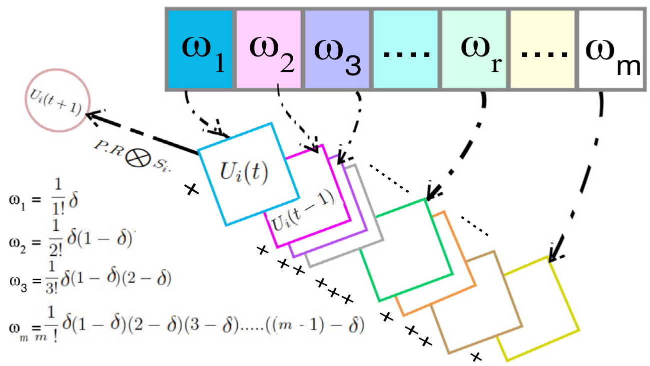

Fig. 3. Agent motion update based on FC concept.

By taking into account the early mentioned relation in Eq. (21), the general formulation for the solutions of FOCLMPA based on FC memory perspective can be written as follows:

$$Z_i(t+1) = -\sum_{k=1}^{m} \frac{(-1)^k \Gamma(\delta+1) Z_i(t+1-k)}{\Gamma(k+1)\Gamma(\delta-k+1)} + P.R \bigotimes S_i. \tag{22}$$

After checking the previous formula, it can be detected that the motion of the prey becomes based on some terms from the previous solutions with a length of (m), as depicted in Fig. 3. With accounting the first four previous events ($m = 4$) from the memory data with derivative order δ, the position of prey can be modified as follow;

$$\begin{aligned} Z_i(t+1) &= \frac{1}{1!}\delta Z_i(t) + \frac{1}{2!}\delta(1-\delta)Z_i(t-1) \\ &+ \frac{1}{3!}\delta(1-\delta)(2-\delta)Z_i(t-2) \\ &+ \frac{1}{4!}\delta(1-\delta)(2-\delta)(3-\delta)Z_i(t-3) + P.R \bigotimes S_i. \end{aligned} \tag{23}$$

Through combining modifications approaches 1 and 2, the FOCLMPA has been introduced. The main steps of the proposed algorithm can be summarized in Algorithm 1 as follows:

3. Simulation and results

The quality of the proposed algorithm FOCLMPA has been assessed and confirmed based on several series of analyses with conducting numerous statistical measures, non-parametric tests including Wilcoxon sum-rank test at 0.05 significant level, and the results ranking. Moreover, three signs of $(+/=/-)$ have been listed referring to the number of the functions in which the FOCLMPA has a superiority performance $(+)$, that the FOCLMPA has the same behaviour in comparison with the other contest $(=)$, and the number of functions where the FOCLMPA has located in the inferior $(-)$. The sensitivity of the proposed FOCLMPA for the derivative order (δ) has been addressed to determine the effective value of the derivative order on the performance of the proposed algorithm. Several dimensional CEC2017 and CEC2020 test functions have been recognized as challenging benchmarks for the numerical validation part. Also two real applications including four engineering optimization problems and eighteen UCI datasets for feature selection optimization problem have been employed for a comprehensive evaluation of the FOCLMPA. The FOCLMPA has been compared with well-regarded optimization algorithms. Seeking for unbiased comparisons, all the algorithms

Algorithm 1 Steps of FOCLMPA

```
Set the FOCLMPA parameters; N agents, derivative order n, number of iterations (T), derivative order (δ) and optimization problem bounds; LB, and UB.
Calculate the initial solutions matrix using Eq. (1).
Saved the initial values for the memory window with length (m).
Set t=1.
while termination criteria are not met do
  Evaluate the objective function (obj).
  Assign the Elite matrix using Eq. (2).
  if t < t_max/3 then
    Calculate the acceleration coefficient K using formula of Eq. (18).
    for each search agent (i = 1, ..., n) do
      Generate random number rand
      Calculate the learning probability value using (P_Cl_i) Eq. (13).
      Select two particles with rand indexes f1_i^d, f2_i^d , d=1,2,3 ,.. DIM.
      if rand > P_Cl_i then
        if obj(Z_f1_i^d) > obj(Z_f2_i^d) then
          Z_Exemplar_fi = Z_f2_i^d.
        else
          Z_Exemplar_fi = Z_f1_i^d.
        end if
      else
        Z_Exemplar_fi = Z_i.
      end if
    end for
    Update the agents' locations using Eq. (18).
  else if t_max/3 < t < 2 × t_max/3 then
    for The first half of the agents (i = 1, ..., n/2) do
      Update the agents' locations based on the memory perspective using Eq. (23).
    end for
    for The second half of the agents (i = n/2, ..., n) do
      Update the agents' locations using Eq. (8).
    end for
  else if t > 2 × t_max/3 then
    Update the agents' locations using Eqs. (10)–(11).
  end if
  Using FADs effect and Eq. (12) to update current agent.
  Update memory window based on first in first out approach.
  t=t+1.
end while
```

have been implemented with the same settings of the population size and the maximum number of evaluation function, furthermore, all the series of analyses have been carried out on Matlab 2018b working on a laptop with Core i7-6500U CPU, 2.5 GHz of speed and 16 GB of RAM.

3.1. Sensitivity analysis

With varying the derivative order δ, the contribution weights of the historical terms (past solutions) are changed as shown in Eq. (23), sequentially that reflects on the updating process of the agents' positions and the obtained solutions. Therefore, the sensitivity of the FOCLMPA for the δ values has been studied in the current section, and the adequate value of δ is recommended. A set of Twenty-eight CEC2017 benchmark functions

Table 1
The CEC2017 benchmark functions [28].

Type	No.	Description	F_{min}	Range
Unimodal functions	3	Shifted and Rotated Zakharov Function	300	[−100,100]
Simple multimodal functions	4	Shifted and Rotated Rosenbrock's Function	400	[−100,100]
	5	Shifted and Rotated Rastrigin's Function	500	[−100,100]
	6	Shifted and Rotated Expanded Scaffer's F6 Function	600	[−100,100]
	7	Shifted and Rotated Lunacek Bi-Rastrigin Function	700	[−100,100]
	8	Shifted and Rotated Non-Continuous Rastrigin's Function	800	[−100,100]
	9	Shifted and Rotated Levy Function	900	[−100,100]
	10	Shifted and Rotated Schwefel's Function	1000	[−100,100]
Hybrid functions	11	Hybrid Function of Zakharov, Rosenbrock and Rastrigin's	1100	[−100,100]
	12	Hybrid Function of High Conditioned Elliptic, Modified Schwefel and Bent Cigar	1200	[−100,100]
	13	Hybrid Function of Bent Ciagr, Rosenbrock and Lunache Bi-Rastrigin	1300	[−100,100]
	14	Hybrid Function of Elliptic, Ackley, Schaffer and Rastrigin	1400	[−100,100]
	15	Hybrid Function of Bent Cigar, HGBat, Rastrigin and Rosenbrock	1500	[−100,100]
	16	Hybrid Function of Expanded Schaffer, HGBat, Rosenbrock and Modified Schwefel	1600	[−100,100]
	17	Hybrid Function of Katsuura, Ackley, Expanded Griewank plus Rosenbrock, Modified Schwefel and Rastrigin	1700	[−100,100]
	18	Hybrid Function of high conditioned Elliptic, Ackley, Rastrigin, HGBat and Discus	1800	[−100,100]
	19	Hybrid Function of Bent Cigar, Rastrigin, Expanded Grienwank plus Rosenbrock, Weierstrass and expanded Schaffer	1900	[−100,100]
	20	Hybrid Function of Happycat, Katsuura, Ackley, Rastrigin, Modified Schwefel and Schaffer	2000	[−100,100]
Composition functions	21	Composition Function of Rosenbrock, High Conditioned Elliptic and Rastrigin	2100	[−100,100]
	22	Composition Function of Rastrigin's, Griewank's and Modified Schwefel's	2200	[−100,100]
	23	Composition Function of Rosenbrock, Ackley, Modified Schwefel and Rastrigin	2300	[−100,100]
	24	Composition Function of Ackley, High Conditioned Elliptic, Girewank and Rastrigin	2400	[−100,100]
	25	Composition Function of Rastrigin, Happycat, Ackley, Discus and Rosenbrock	2500	[−100,100]
	26	Composition Function of Expanded Scaffer, Modified Schwefel, Griewank, Rosenbrock and Rastrigin	2600	[−100,100]
	27	Composition Function of HGBat, Rastrigin, Modified Schwefel, Bent-Cigar, High Conditioned Elliptic and Expanded Scaffer	2700	[−100,100]
	28	Composition Function of Ackley, Griewank, Discus, Rosenbrock, HappyCat, Expanded Scaffer	2800	[−100,100]
	29	Composition Function of shifted and rotated Rastrigin, Expanded Scaffer and Lunacek Bi-Rastrigin	2900	[−100,100]
	30	Composition Function of shifted and rotated Rastrigin, Non-Continuous Rastrigin and Levy Function	3000	[−100,100]

(F3–F30) with dimension 30 have been considered. The regarded test functions composed of four groups, the F3 is a unimodal function, the F4–F10 subset is simple multi-modal functions, the F11–F20 subset is hybrid functions, and the F21 to F30 subset corresponds to composition functions. The specifications of the studied algorithms are listed in Table 1. Ten values of δ are varied from 0.1 to 1, with step 0.1 have been recognized to assign the most suitable derivative order value with the proposed FOCLMPA. The algorithm variants have been implemented considering the maximum of the evaluation function is 10000DIM (DIM is the dimension of the studied functions) and search agents are 30 [28]. The average (AVR) and standard deviation (STD) values of the used functions across the independent execution times (30 runs) have also been computed the ranking of the FOCLMPA variants with changing the δ values have reported in Table 2.

The AVR and STD values of the applied functions across the independent runs by the FOCLMPA variants with changing the derivative orders are reported in Table 2. The presented results show that the proposed FOCLMPA variants attain the closest results for the global values of the majority of the used test functions in comparison with the basic MPA. For the unimodal function of F3, the FOCLMPA at $\delta = 0.2$ is the best variant; therefore, it has the first rank in optimizing this function and is followed by the FOCLMPA at $\delta = 1$, and 0.5 whilst the MPA locates in the fourth rank. For the multimodal test functions set of F4–F10, the FOCLMPA at $\delta = 0.6$ occupies the first rank for four test functions out of this set (F4, F5, F7, F8) as it offers the least AVR values of these functions meanwhile, the MPA has an average rank of 10.25 (where MPA ranks $(8 + 11 + 11 + 11)/4$) over these functions. For the hybrid functions suite of F11–F20, the FOCLMPA variants are located at the first rank except for F17, where MPA has the first rank. The FOCLMPA at derivative orders 0.5 and 0.6 have the first and second ranks in six functions of the hybrid benchmarks set, while the MPA is located in the inferior ranks in four functions of this set. For the composition test functions set of F21 to F30; however, the MPA shows the superior performance for four functions of this set, the FOCLMPA variants attain the best results for the other six functions. The composite test problems are the most difficult functions in the benchmark set. They are highly multimodal, non-separable functions, and have different attributes near a huge number of the local optima. The better performance of the FOCLMPA variants in comparison to MPA across these functions confirms the remarkable and significant modification that fractional calculus and CL concept provided for MPA for enhancing its main phases (exploration and exploitation). To sum up the overall results, the average (AVR) rank values of the proposed variants across the studied twenty-eight problems have been computed at the last row in Table 2. The computed AVR rank values show that the FOCLMPA at $\delta = 0.5$ and 0.6 considered the best variants for optimizing the 30-dimensional CEC2017 benchmarks as they achieve the least average ranks. Consequently, the recommended effective derivative order is the average of these two values $((0.5 + 0.6)/2) = 0.55$.

Table 2
Comparison of simulation results of MPA and FOCLMPA with number of δ changed from 0.1 to 1 for 30 dimensional CEC17.

Fun/meas		MPA	FOCLMPA									
			$\delta = 0.1$	$\delta = 0.2$	$\delta = 0.3$	$\delta = 0.4$	$\delta = 0.5$	$\delta = 0.6$	$\delta = 0.7$	$\delta = 0.8$	$\delta = 0.9$	$\delta = 1$
F3	AVR	3.0000e+02	3.0000e+02	**3.0000e+02**	3.0000e+02	3.0000e+02	3.0000e+02	3.0000e+02	3.0000e+02	3.0000e+02	3.0000e+02	3.0000e+02
	STD	3.6618e−03	3.6698e−03	**1.6174e−03**	3.7413e−03	3.5510e−03	1.9999e−03	2.7480e−03	3.8883e−03	5.6318e−03	3.6837e−03	2.2008e−03
	Rank	4	7	1	6	8	3	5	10	11	9	2
F4	AVR	4.7483e+02	4.7806e+02	4.7000e+02	4.7459e+02	4.7951e+02	4.7176e+02	**4.6658e+02**	4.8252e+02	4.6941e+02	4.7238e+02	4.7283e+02
	STD	2.3588e+01	1.9273e+01	3.2213e+01	3.2841e+01	2.5684e+01	2.5267e+01	**3.0614e+01**	2.6265e+01	3.0841e+01	2.9279e+01	3.0823e+01
	Rank	8	9	3	7	10	4	1	11	2	5	6
F5	AVR	5.6196e+02	5.4242e+02	5.4364e+02	5.4176e+02	5.4463e+02	5.4073e+02	**5.4040e+02**	5.4083e+02	5.4136e+02	5.4515e+02	5.4233e+02
	STD	1.1210e+01	1.0137e+01	1.1886e+01	1.1778e+01	1.4445e+01	1.0440e+01	**9.7261e+00**	1.2941e+01	9.3768e+00	1.2944e+01	1.1012e+01
	Rank	11	7	8	5	9	2	1	3	4	10	6
F6	AVR	6.0014e+02	6.0018e+02	6.0014e+02	6.0015e+02	6.0023e+02	6.0015e+02	6.0022e+02	**6.0011e+02**	6.0020e+02	6.0016e+02	6.0019e+02
	STD	1.4557e−01	1.2943e−01	1.4069e−01	1.5300e−01	1.8585e−01	1.5555e−01	2.3349e−01	**1.3342e−01**	1.8737e−01	1.5316e−01	1.4618e−01
	Rank	2	7	3	4	11	5	10	1	9	6	8
F7	AVR	8.1409e+02	7.9596e+02	7.9806e+02	7.9596e+02	8.0237e+02	7.9621e+02	**7.9401e+02**	8.0012e+02	7.9494e+02	7.9560e+02	7.9447e+02
	STD	2.2336e+01	1.4376e+01	1.4044e+01	1.6926e+01	1.3333e+01	1.6718e+01	**1.5200e+01**	1.4466e+01	1.4585e+01	1.8230e+01	1.9515e+01
	Rank	11	5	8	6	10	7	1	9	3	4	2
F8	AVR	8.6983e+02	8.4686e+02	8.4699e+02	8.4932e+02	8.4477e+02	8.4381e+02	**8.4363e+02**	8.4499e+02	8.4617e+02	8.4773e+02	8.4647e+02
	STD	1.5084e+01	1.2689e+01	1.2190e+01	1.2628e+01	1.2482e+01	1.2697e+01	**1.1632e+01**	1.2629e+01	1.1815e+01	1.3447e+01	1.0363e+01
	Rank	11	7	8	10	3	2	1	4	5	9	6
F9	AVR	1.0430e+03	1.0211e+03	1.0465e+03	1.0339e+03	1.0249e+03	1.0318e+03	1.0332e+03	1.0394e+03	**1.0196e+03**	1.0376e+03	1.0430e+03
	STD	8.1894e+01	4.4817e+01	6.5142e+01	7.0213e+01	4.5850e+01	6.3370e+01	6.2153e+01	6.0235e+01	**5.8011e+01**	7.0813e+01	7.0177e+01
	Rank	9	2	11	6	3	4	5	8	1	7	10
F10	AVR	3.4773e+03	3.5689e+03	3.4665e+03	3.6346e+03	3.5730e+03	3.6030e+03	3.6346e+03	3.5075e+03	3.5195e+03	**3.4210e+03**	3.4872e+03
	STD	3.0609e+02	3.9360e+02	5.1000e+02	4.4311e+02	3.7714e+02	3.7210e+02	4.5532e+02	3.7173e+02	4.2527e+02	**3.9583e+02**	4.6552e+02
	Rank	3	7	2	11	8	9	10	5	6	1	4
F11	AVR	1.1626e+03	1.1669e+03	1.1686e+03	1.1632e+03	1.1664e+03	1.1574e+03	1.1707e+03	1.1652e+03	1.1627e+03	1.1648e+03	**1.1574e+03**
	STD	2.7191e+01	3.1112e+01	3.0189e+01	3.5713e+01	3.1344e+01	2.4094e+01	3.2260e+01	2.8783e+01	2.9226e+01	2.6834e+01	**2.4631e+01**
	Rank	3	9	10	5	8	2	11	7	4	6	1
F12	AVR	5.9847e+04	6.5566e+04	6.6146e+04	6.5906e+04	7.0235e+04	6.1253e+04	6.1821e+04	**4.8397e+04**	7.2248e+04	6.3580e+04	5.7865e+04
	STD	4.7665e+04	6.3918e+04	5.1517e+04	4.3588e+04	6.8848e+04	7.1426e+04	5.5944e+04	**3.3911e+04**	6.2674e+04	5.9866e+04	4.5260e+04
	Rank	3	7	9	8	10	4	5	1	11	6	2
F13	AVR	1.4468e+03	1.4485e+03	**1.4272e**+03	1.4427e+03	1.4403e+03	1.4384e+03	1.4456e+03	1.4313e+03	1.4653e+03	1.4484e+03	1.4658e+03
	STD	5.2269e+01	5.3520e+01	**5.4519e+01**	5.1433e+01	6.8361e+01	5.9574e+01	6.4347e+01	4.7745e+01	1.0048e+02	4.2595e+01	6.8058e+01
	Rank	7	9	1	5	4	3	6	2	10	8	11
F14	AVR	1.4307e+03	1.4344e+03	1.4322e+03	1.4312e+03	1.4350e+03	**1.4293e+03**	1.4296e+03	1.4328e+03	1.4320e+03	1.4323e+03	1.4295e+03
	STD	1.1467e+01	8.4718e+00	1.0819e+01	9.4488e+00	6.9137e+00	**1.0805e+01**	1.3237e+01	9.0631e+00	8.6331e+00	8.6330e+00	1.1525e+01
	Rank	4	10	7	5	11	1	3	9	6	8	2
F15	AVR	1.5573e+03	**1.5356e+03**	1.5401e+03	1.5418e+03	1.5439e+03	1.5481e+03	1.5410e+03	1.5376e+03	1.5420e+03	1.5430e+03	1.5526e+03
	STD	3.0839e+01	**2.1224e+01**	2.2568e+01	2.3977e+01	2.2210e+01	2.2863e+01	1.7578e+01	1.3704e+01	2.0407e+01	2.2085e+01	3.0445e+01
	Rank	11	1	3	5	8	9	4	2	6	7	10
F16	AVR	1.9930e+03	2.0421e+03	2.0203e+03	2.0142e+03	2.0354e+03	1.9937e+03	**1.9835e+03**	1.9995e+03	2.0086e+03	1.9869e+03	2.0252e+03
	STD	1.5728e+02	1.3105e+02	1.1378e+02	1.4453e+02	1.4568e+02	1.8854e+02	**1.4366e+02**	1.4934e+02	1.6287e+02	1.3514e+02	1.7950e+02
	Rank	3	11	8	7	10	4	1	5	6	2	9
F17	AVR	**1.7461e+03**	1.7604e+03	1.7648e+03	1.7783e+03	1.7602e+03	1.7546e+03	1.7546e+03	1.7531e+03	1.7552e+03	1.7495e+03	1.7766e+03
	STD	**1.4228e+01**	2.5378e+01	4.1972e+01	4.3310e+01	3.9118e+01	2.6133e+01	2.0215e+01	2.6151e+01	3.5688e+01	2.5403e+01	4.9777e+01
	Rank	1	8	9	11	7	4	5	3	6	2	10

(*continued on next page*)

Table 2 (*continued*).

Fun/meas		MPA	FOCLMPA									
			$\delta=0.1$	$\delta=0.2$	$\delta=0.3$	$\delta=0.4$	$\delta=0.5$	$\delta=0.6$	$\delta=0.7$	$\delta=0.8$	$\delta=0.9$	$\delta=1$
F18	AVR	1.8439e+03	1.8432e+03	1.8418e+03	1.8457e+03	1.8423e+03	1.8403e+03	**1.8387e+03**	1.8489e+03	1.8430e+03	1.8434e+03	1.8411e+03
	STD	1.4642e+01	1.3469e+01	1.0497e+01	1.4899e+01	1.2593e+01	1.0982e+01	**1.0584e+01**	1.8318e+01	1.3553e+01	1.7826e+01	1.1840e+01
	Rank	9	7	4	10	5	2	1	11	6	8	3
F19	AVR	1.9140e+03	1.9138e+03	1.9154e+03	1.9164e+03	1.9157e+03	1.9148e+03	1.9138e+03	1.9157e+03	1.9158e+03	1.9152e+03	**1.9134e+03**
	STD	5.1073e+00	6.6201e+00	4.6604e+00	7.6117e+00	6.5928e+00	4.7328e+00	4.3963e+00	6.1989e+00	6.4665e+00	6.0369e+00	**4.9193e+00**
	Rank	4	3	7	11	9	5	2	8	10	6	1
F20	AVR	2.1143e+03	2.0796e+03	2.0861e+03	2.0853e+03	2.0915e+03	2.0990e+03	2.0920e+03	2.0999e+03	2.0881e+03	**2.0726e+03**	2.0958e+03
	STD	5.7427e+01	5.8461e+01	5.9638e+01	5.7795e+01	6.2264e+01	6.8593e+01	7.7316e+01	7.2517e+01	6.2920e+01	**5.8326e+01**	6.5142e+01
	Rank	11	2	4	3	6	9	7	10	5	1	8
F21	AVR	2.3432e+03	2.3340e+03	2.3371e+03	2.3316e+03	**2.3261e+03**	2.3363e+03	2.3296e+03	2.3409e+03	2.3388e+03	2.3411e+03	2.3340e+03
	STD	2.9699e+01	2.6740e+01	2.7071e+01	3.7102e+01	**4.3784e+01**	2.7748e+01	3.6690e+01	1.1735e+01	2.8955e+01	1.4051e+01	2.6987e+01
	Rank	11	4	7	3	1	6	2	9	8	10	5
F22	AVR	**2.3012e+03**	2.3020e+03	2.3028e+03	2.3026e+03	2.3022e+03	2.3017e+03	2.3029e+03	2.3016e+03	2.3037e+03	2.3026e+03	2.3026e+03
	STD	**2.0066e+00**	3.4278e+00	3.9562e+00	3.9972e+00	3.1721e+00	2.9523e+00	3.8492e+00	3.2315e+00	3.8606e+00	4.2214e+00	3.5613e+00
	Rank	1	4	9	6	5	3	10	2	11	8	7
F23	AVR	2.6923e+03	2.6875e+03	2.6873e+03	2.6896e+03	2.6893e+03	**2.6828e+03**	2.6892e+03	2.6873e+03	2.6885e+03	2.6911e+03	2.6921e+03
	STD	1.6353e+01	1.6411e+01	1.3198e+01	1.7460e+01	1.1388e+01	**4.7605e+01**	1.1342e+01	1.3767e+01	1.3461e+01	1.1131e+01	1.4549e+01
	Rank	11	4	2	8	7	1	6	3	5	9	10
F24	AVR	2.8757e+03	2.8658e+03	2.8665e+03	2.8671e+03	2.8681e+03	2.8691e+03	2.8713e+03	2.8708e+03	2.8673e+03	2.8716e+03	**2.8657e+03**
	STD	1.4745e+01	1.2503e+01	1.2271e+01	1.1451e+01	1.3855e+01	1.1328e+01	1.2674e+01	1.2704e+01	1.5504e+01	1.4533e+01	**1.2146e+01**
	Rank	11	2	3	4	6	7	9	8	5	10	1
F25	AVR	2.8905e+03	2.8953e+03	**2.8888e+03**	2.8913e+03	2.8937e+03	2.8910e+03	2.8920e+03	2.8941e+03	2.8930e+03	2.8898e+03	2.8974e+03
	STD	1.3997e+01	1.5050e+01	**9.9018e+00**	1.2264e+01	1.5501e+01	1.2657e+01	1.1900e+01	1.3801e+01	1.5867e+01	8.8030e+00	1.8624e+01
	Rank	3	10	1	5	8	4	6	9	7	2	11
F26	AVR	**2.9000e+03**	3.1341e+03	3.1542e+03	3.2875e+03	3.1323e+03	3.2522e+03	3.1706e+03	3.0167e+03	3.0545e+03	3.2213e+03	3.1608e+03
	STD	**8.7330e−04**	4.8293e+02	4.8878e+02	6.1330e+02	4.8578e+02	5.5272e+02	5.1347e+02	3.9252e+02	4.1282e+02	5.3814e+02	5.0549e+02
	Rank	1	5	6	11	4	10	8	2	3	9	7
F27	AVR	3.2106e+03	3.2068e+03	3.2053e+03	3.2079e+03	3.2068e+03	**3.2045e+03**	3.2070e+03	3.2054e+03	3.2088e+03	3.2082e+03	3.2078e+03
	STD	1.1532e+01	9.9117e+00	1.1836e+01	9.8907e+00	1.2103e+01	**1.0577e+01**	9.0193e+00	1.2105e+01	9.9765e+00	1.1763e+01	1.0635e+01
	Rank	11	4	2	8	5	1	6	3	10	9	7
F28	AVR	**3.1477e+03**	3.1733e+03	3.1930e+03	3.1702e+03	3.1810e+03	3.1695e+03	3.1976e+03	3.1953e+03	3.2024e+03	3.1733e+03	3.1817e+03
	STD	**5.4723e+01**	4.8940e+01	4.6249e+01	4.7892e+01	4.9633e+01	4.8031e+01	5.1812e+01	4.7571e+01	3.8326e+01	4.9182e+01	4.5198e+01
	Rank	1	5	8	3	6	2	10	9	11	4	7
F29	AVR	**3.3771e+03**	3.3897e+03	3.3949e+03	3.3779e+03	3.3828e+03	3.4030e+03	3.3779e+03	3.4015e+03	3.3978e+03	3.3992e+03	3.4021e+03
	STD	**5.1486e+01**	4.9281e+01	7.9493e+01	4.6117e+01	6.1445e+01	7.5602e+01	6.7724e+01	6.7785e+01	4.5913e+01	7.4764e+01	6.2021e+01
	Rank	1	5	6	2	4	11	3	9	7	8	10
F30	AVR	5.3946e+03	5.4466e+03	5.3748e+03	5.3764e+03	5.4520e+03	5.4090e+03	**5.2930e+03**	5.3684e+03	5.4422e+03	5.3354e+03	5.3013e+03
	STD	3.6332e+02	5.0208e+02	3.9761e+02	3.1982e+02	3.5016e+02	3.5111e+02	**2.7515e+02**	3.8509e+02	4.3920e+02	3.3494e+02	3.3724e+02
	Rank	7	10	5	6	11	8	1	4	9	3	2
	AVR rank	6.1786	6.1071	5.5357	6.4643	7.0357	**4.7143**	**5**	5.9643	6.6786	6.3214	6

3.2. Stability test

In this section, the stability of the proposed FOCLMPA at the recommended $\delta = 0.55$ (viewed as FOCLMPA) has been evaluated while handling 50 dimensional CEC2017 benchmark functions. Moreover, the FOCLMPA has been compared with well-regarded state-of-the-art algorithms including the basic MPA [7], Artificial ecosystem-based optimization (AEO) [30], Slime mould algorithm (SMA) [31], Manta ray foraging optimization (MRFO) [32], Sine–cosine algorithm (SCA) [33], comprehensive learning particle swarm optimizer [15], as well as the basic Particle swarm optimizer [34] and Differential evolution with composite trial vector generation strategies (CODE) [35]. The algorithms have executed with the maximum number of evaluation function is 10000DIM (DIM $= 50$) and the search agents is 30 for independent runs of 30 [28]. The average and STD of the utilized problems and ranking values of the implemented algorithms have been illustrated as in the Table 3. The p-values, and the null hypothesis test based on non-parametric Wilcoxon sum-rank test at 0.05 significant level have been performed in Table 4 to show the pairwise comparison among the FOCLMPA and the other contests. Moreover, the three signs of $(+/-/=)$ have been listed at the last row in Table 4 to show the number of the functions which the FOCLMPA has a superiority $(+)$ performance, the FOCLMPA has the same behaviour in comparison with the other contest $(=)$ and the number of functions where the FOCLMPA has located in the inferior.

The attained AVR, STD and rank values in Table 3 show that the proposed FOCLMPA offers the highest qualified results in the majority of these functions especially while dealing with the multimodal (F5), hybrid (F14,F15,F16,F17,F18,F20), and composition functions (F21, F23, F24, F28, F29, F30). Therefore, the FOCLMPA has the least average rank whilst the MPA occupied the second position with achieving the best performance for only two functions (F22, F12) out of these testbed functions. This observation confirms the efficient exploration and exploitation abilities of the proposed FOCLMPA in discovering the search domain, exploiting the optimal solutions accurately and avoiding the many local optima such in the high dimensional composition functions. Table 4 displays pairwise comparison among the FOCLPMA and the other peers in the basis of the p-values, and the null hypothesis test using non-parametric Wilcoxon sum-rank test at 0.05 significant level. The pairwise comparison approves the significant difference between the proposed FOCLMPO and the other counterparts in favour of the proposed FOCLMPA as the p-values are less than 0.05 in most of the cases. The FOCLMPA considered the best variant in 16 functions in comparing with MPA, and shows a high success in comparing with AEO, SMA, MRFO, SCA, CLPSO, PSO and CODE as it recognized as the best algorithm for more than 22 functions with respect to these techniques. This observation states that FOCLMPA can be recognized as one of the reliable and the stable algorithms.

The second part of the analysis is the convergence speed of the proposed FOCLMPA, therefore the mean convergence curves of the FOCLMPA have been appraised in comparison with the other state-of-the-art techniques as in Fig. 4 while dealing with 50 dimensional CEC2017 functions. By inspecting the curves, one can see that the SCA exposes an observable stagnation over all the implemented functions. For the AEO, SMA, and MRFO techniques, they trapped in a local solution of the majority of the functions. The FOCLMPA exhibits a high balance among the exploration and exploitation phases as it has an ability to discover and exploit the optimal solution in fastest response in comparing with the other peers in the cases of the multimodal, hybrid, and composition functions as in functions of F5, F8, F16, F21, F23, F27 and F30. Accordingly, we can detect that, combining the CL approach in the exploration stage enhanced the diversity of the search agents and boosts their ability in discovering the search landscape meanwhile, integrating the memory properties of FC boosts exploitation the best solutions and helped in avoiding trapping in the local solutions.

3.3. Valuation FOCLMPA with utilizing CEC2020

In this section the recommended variant of FOCLMPA has been examined with the recent suite of the CEC that include ten functions of CEC2020. The studied dimensions of this section are 5 and 20, the specifications of the considered functions set are reported in Table 5. The number of evaluation function is tuned based on the CEC2020 report where for dimensions 5 and 20, the utilized number of evaluations function (NEF) are 50 000 and 10 000 000, respectively [29], the utilized population size (30). The FOCLMPA$_{0.55}$ has been compared with MPA, MRFO, AEO, SMA, HHO, PSO, and CODE. Each algorithm has been implemented for 30 independent runs. The best, AVR, worst, and STD values as well as algorithms ranking values have been carried out as presented in Tables 6 and 7 to provide a meaningful statistical comparison between the FOCLMPA$_{0.55}$ and the other considered techniques. The best, AVR, Worst and STD results (see Table 6) illustrate that the FOCLMPA offers the least mean values for seven functions of 5 dimensional benchmarks and five functions of 20 dimensional benchmarks whereas the MPA has the same feature for only one function in the cases of 5 and 20 dimensional benchmarks. Moreover, the results of the other counterparts confirmed the quality of the FOCLMPA while dealing with these set of the test functions. To assign the FOCLMPA final rank across the considered functions, Table 7 exhibits the rank values of the implemented algorithms and the average value of the occupied ranks have been computed. The rank values state that the FOCLMPA has an efficient performance on comparison with the other peers.

3.4. Real application: Engineering optimization problems

In this section, the developed model is applied to solve four real-world engineering problems including welded beam design, tension/compression spring design, pressure vessel design, and rolling element bearing design. However, these are constraint optimization problems, for that reason in this article is used the following formula (following [37]) to handle with the constraints.

$$f(X) = f(X)\sum_{j=1}^{m} Pe_j max\{g_i(X), 0\} + \sum_{k=1}^{n} Pe_k max\{|h_k(X) - \varepsilon|, 0\} \quad (24)$$

In Eq. (24), Pe_j and Pe_k denote the cost functions. $\varepsilon = 1e-6$ denotes the error of equilibrium constraints. This method is simple and less computationally than other methods.

3.4.1. Welded beam design (WBD) problem

In this minimization problem for the fabrication of a welded bean, the objective function is the fabrication costs. Moreover, several constraints are considered to find the value of the parameters that reduce this cost (see Fig. 5). The constraints include : (I) Shear stress (τ) (II) Bending stress in the beam (σ) (III) Buckling load on the bar (P_c) (IV) End deflection of the beam (γ) (V) Side constraints. The WBD has four unknown parameters that are the thickness of the bar (b), the length of the attached part of the bar (l), thickness of weld (h), and the height of the bar (t) [38]. The

Table 3
Comparison of simulation results of MPA and FOCLMPA recommended variant for 50 dimensional CEC17.

Fun	meas	FOCLMPA	MPA	AEO	SMA	MRFO	SCA	CLPSO	PSO	CODE
F3	AVR	3.0029e+02	3.0044e+02	3.0000e+02	3.0052e+02	3.0061e+02	9.9394e+04	7.2894e+04	1.1823e+04	1.1823e+04
	STD	1.5104e−01	3.9042e−01	2.5415e−03	3.3306e−01	1.4242e+00	1.1904e+04	1.1351e+04	3.6550e+03	3.6550e+03
	Rank	2	3	1	4	5	9	8	6	7
F4	AVR	4.8174e+02	5.0861e+02	5.1206e+02	5.4940e+02	4.6152e+02	6.1270e+03	4.6808e+02	6.9121e+02	6.9121e+02
	STD	4.3720e+01	4.0752e+01	4.5122e+01	5.3517e+01	4.5205e+01	1.0654e+03	2.4191e+01	7.7460e+01	7.7460e+01
	Rank	3	4	5	6	1	9	2	7	8
F5	AVR	5.9991e+02	6.9803e+02	8.0105e+02	7.0992e+02	8.2425e+02	1.0504e+03	6.1135e+02	6.3764e+02	6.3764e+02
	STD	1.1838e+01	3.1293e+01	4.4076e+01	4.3070e+01	3.9911e+01	3.6468e+01	1.2489e+01	2.9924e+01	2.9924e+01
	Rank	1	5	7	6	8	9	2	3	4
F6	AVR	6.0323e+02	6.0313e+02	6.2416e+02	6.0575e+02	6.4765e+02	6.6731e+02	6.0000e+02	6.0121e+02	6.0121e+02
	STD	8.8872e−01	1.1792e+00	5.6979e+00	2.2689e+00	1.1227e+01	4.6816e+00	2.2041e−13	7.6182e−01	7.6182e−01
	Rank	5	4	7	6	8	9	1	2	3
F7	AVR	9.8600e+02	1.0328e+03	1.3479e+03	9.8578e+02	1.4977e+03	1.6168e+03	8.6817e+02	9.4995e+02	9.4995e+02
	STD	3.3573e+01	5.9938e+01	1.1239e+02	4.6819e+01	1.3581e+02	6.7756e+01	1.4195e+01	4.9071e+01	4.9071e+01
	Rank	5	6	7	4	8	9	1	2	3
F8	AVR	9.0052e+02	9.9675e+02	1.0999e+03	9.8827e+02	1.1382e+03	1.3551e+03	9.1739e+02	9.4820e+02	9.4820e+02
	STD	1.1531e+01	3.9440e+01	4.7281e+01	4.5826e+01	4.4933e+01	2.9863e+01	1.3941e+01	2.3206e+01	2.3206e+01
	Rank	1	6	7	5	8	9	2	3	4
F9	AVR	2.1533e+03	2.5333e+03	1.0008e+04	8.8045e+03	1.0262e+04	2.1617e+04	2.0965e+03	1.7962e+03	1.7962e+03
	STD	2.7197e+02	4.2415e+02	1.9496e+03	4.1619e+03	2.1798e+03	3.6272e+03	4.6092e+02	1.0520e+03	1.0520e+03
	Rank	3	4	6	5	7	8	2	1	1
F10	AVR	6.0252e+03	5.8260e+03	7.3697e+03	7.3335e+03	7.4781e+03	1.4244e+04	5.1440e+03	6.9984e+03	6.9984e+03
	STD	4.4183e+02	7.6416e+02	1.0055e+03	8.1754e+02	8.6425e+02	4.5982e+02	3.9942e+02	1.3399e+03	1.3399e+03
	Rank	3	2	7	6	8	9	1	4	5
F11	AVR	1.2831e+03	1.2869e+03	1.2541e+03	1.3906e+03	1.2504e+03	5.7885e+03	1.2075e+03	1.3255e+03	1.3255e+03
	STD	3.8052e+01	4.3040e+01	3.7379e+01	7.2417e+01	2.8364e+01	1.2348e+03	3.2684e+01	6.8586e+01	6.8586e+01
	Rank	4	5	3	8	2	9	1	6	7
F12	AVR	1.7695e+06	3.0396e+06	2.3559e+05	5.6105e+06	1.1986e+05	1.1583e+10	3.0759e+06	2.7271e+06	2.7271e+06
	STD	8.2152e+05	1.9795e+06	1.3700e+05	3.3639e+06	5.4176e+04	2.2134e+09	1.6352e+06	2.5234e+06	2.5234e+06
	Rank	3	6	2	8	1	9	7	4	5
F13	AVR	3.2148e+03	6.0259e+03	8.0892e+03	3.5581e+04	5.4121e+03	2.6048e+09	1.7941e+03	7.0926e+03	7.0926e+03
	STD	1.3561e+03	4.8481e+03	5.4013e+03	9.0619e+03	4.8026e+03	9.6863e+08	1.5857e+02	6.8538e+03	6.8538e+03
	Rank	2	4	7	8	3	9	1	5	6
F14	AVR	1.4886e+03	1.5037e+03	4.9568e+03	1.2329e+05	6.2703e+03	2.0975e+06	4.3151e+05	8.7627e+04	8.7627e+04
	STD	1.3916e+01	2.1113e+01	3.2941e+03	8.1386e+04	5.0191e+03	9.9705e+05	2.7439e+05	8.2944e+04	8.2944e+04
	Rank	1	2	3	7	4	9	8	5	6
F15	AVR	1.6851e+03	1.7732e+03	7.5949e+03	2.6560e+04	1.0394e+04	2.9286e+08	1.8515e+03	8.1477e+03	8.1477e+03
	STD	4.8928e+01	1.1364e+02	6.0860e+03	7.0281e+03	6.3665e+03	1.0403e+08	4.6601e+02	7.2640e+03	7.2640e+03
	Rank	1	2	4	8	7	9	3	5	6
F16	AVR	2.4880e+03	2.6431e+03	3.6353e+03	3.2517e+03	3.3603e+03	5.3052e+03	2.7548e+03	2.9107e+03	2.9107e+03
	STD	1.7604e+02	2.9166e+02	4.1674e+02	3.0762e+02	4.3163e+02	3.0528e+02	1.8429e+02	3.5018e+02	3.5018e+02
	Rank	1	2	8	6	7	9	3	4	5
F17	AVR	2.2815e+03	2.4302e+03	3.1564e+03	3.1007e+03	3.2171e+03	4.1876e+03	2.4892e+03	2.7561e+03	2.7561e+03
	STD	1.3080e+02	1.3871e+02	3.6787e+02	3.8834e+02	3.5632e+02	2.6119e+02	1.4235e+02	3.4257e+02	3.4257e+02
	Rank	1	2	7	6	8	9	3	4	5
F18	AVR	4.4436e+03	5.4092e+03	1.1698e+04	6.5200e+05	6.2117e+04	1.2694e+07	7.3209e+05	1.4151e+06	1.4151e+06
	STD	1.3821e+03	1.8671e+03	6.4595e+03	3.7436e+05	3.1235e+04	5.8121e+06	4.1677e+05	1.5271e+06	1.5271e+06
	Rank	1	2	3	5	4	9	6	7	8
F19	AVR	1.9843e+03	2.0109e+03	1.5851e+04	1.1180e+04	1.6594e+04	2.2395e+08	2.1751e+03	1.1636e+04	1.1636e+04
	STD	2.5499e+01	6.1318e+01	1.0797e+04	1.4291e+04	8.6516e+03	9.1237e+07	5.1407e+02	1.3499e+04	1.3499e+04
	Rank	1	2	7	4	8	9	3	5	6
F20	AVR	2.3643e+03	2.3737e+03	3.0632e+03	2.9847e+03	3.2555e+03	3.8064e+03	2.6232e+03	2.7640e+03	2.7640e+03
	STD	8.0457e+01	1.4721e+02	2.5864e+02	2.6650e+02	2.8232e+02	1.6369e+02	1.8683e+02	3.5288e+02	3.5288e+02
	Rank	1	2	7	6	8	9	3	4	5
F21	AVR	2.3938e+03	2.4244e+03	2.5967e+03	2.5004e+03	2.5924e+03	2.8595e+03	2.4096e+03	2.4284e+03	2.4284e+03
	STD	1.5349e+01	2.5095e+01	5.1341e+01	4.8130e+01	5.9981e+01	3.2544e+01	2.3310e+01	2.6054e+01	2.6054e+01
	Rank	1	3	8	6	7	9	2	4	5
F22	AVR	5.8277e+03	3.2491e+03	9.1708e+03	8.5906e+03	9.7202e+03	1.5891e+04	7.1608e+03	8.6909e+03	8.6909e+03
	STD	2.7532e+03	2.1767e+03	1.5673e+03	8.8234e+02	1.6942e+03	4.7132e+02	9.6858e+02	1.5935e+03	1.5935e+03
	Rank	2	1	7	4	8	9	3	5	6
F23	AVR	2.8301e+03	2.8523e+03	3.1845e+03	2.9374e+03	3.1530e+03	3.4842e+03	2.8600e+03	2.8589e+03	2.8589e+03
	STD	1.8998e+01	3.2472e+01	9.9116e+01	3.7229e+01	1.2025e+02	4.8709e+01	1.8621e+01	2.3081e+01	2.3081e+01
	Rank	1	2	8	6	7	9	5	3	4
F24	AVR	3.0161e+03	3.0266e+03	3.4211e+03	3.0837e+03	3.3588e+03	3.6662e+03	3.1490e+03	3.1199e+03	3.1199e+03
	STD	1.7191e+01	3.0253e+01	1.1719e+02	3.3313e+01	1.2714e+02	5.4666e+01	3.0986e+01	1.2851e+02	1.2851e+02
	Rank	1	2	8	3	7	9	6	4	5

(continued on next page)

Table 3 (*continued*).

Fun	meas	FOCLMPA	MPA	AEO	SMA	MRFO	SCA	CLPSO	PSO	CODE
F25	AVR	3.0208e+03	3.0427e+03	3.0580e+03	3.0272e+03	3.0624e+03	5.9049e+03	3.0284e+03	3.1182e+03	3.1182e+03
	STD	3.6641e+01	3.7884e+01	3.8696e+01	3.9176e+01	4.0919e+01	5.0403e+02	1.4688e+01	4.6437e+01	4.6437e+01
	Rank	1	4	5	2	6	9	3	7	8
F26	AVR	5.0210e+03	3.7282e+03	7.1655e+03	5.6094e+03	7.1147e+03	1.1454e+04	4.6057e+03	5.2261e+03	5.2261e+03
	STD	4.3838e+02	1.2167e+03	3.4366e+03	8.4608e+02	3.9328e+03	4.7080e+02	7.6932e+02	3.5855e+02	3.5855e+02
	Rank	3	1	8	6	7	9	2	4	5
F27	AVR	3.3358e+03	3.3815e+03	3.8471e+03	3.3638e+03	3.7284e+03	4.3405e+03	3.3199e+03	3.4086e+03	3.4086e+03
	STD	4.9625e+01	8.5921e+01	1.7701e+02	6.6761e+01	2.0288e+02	1.6679e+02	2.4788e+01	6.8591e+01	6.8591e+01
	Rank	2	4	8	3	7	9	1	5	6
F28	AVR	3.2868e+03	3.2996e+03	3.3056e+03	3.3093e+03	3.2984e+03	6.2943e+03	3.3087e+03	3.3430e+03	3.3430e+03
	STD	1.9098e+01	3.2796e+01	2.4694e+01	2.2196e+01	3.4640e+01	4.6024e+02	1.1005e+01	4.2886e+01	4.2886e+01
	Rank	1	3	4	6	2	9	5	7	8
F29	AVR	3.5597e+03	3.6236e+03	4.6092e+03	4.3452e+03	4.6241e+03	7.1443e+03	3.6548e+03	3.8221e+03	3.8221e+03
	STD	1.1291e+02	1.6515e+02	2.7624e+02	2.3220e+02	3.1504e+02	5.9274e+02	1.4236e+02	2.4027e+02	2.4027e+02
	Rank	1	2	7	6	8	9	3	4	5
F30	AVR	6.5570e+05	6.8247e+05	1.1993e+06	1.6586e+06	9.0097e+05	5.6755e+08	7.2837e+05	2.4672e+06	2.4672e+06
	STD	4.2720e+04	1.0067e+05	3.1607e+05	3.0622e+05	1.2016e+05	1.8583e+08	5.5434e+04	6.2579e+05	6.2579e+05
	Rank	1	2	5	6	4	9	3	7	8
	Average rank	1.7667	2.9000	5.5333	5.2000	5.6000	8.3667	3.0000	4.2333	5.1333

WBD problem is formulated as:

$$
\begin{aligned}
\textit{Consider}\quad & \vec{u} = [u_1\ u_2\ u_3\ u_4] = [h\ l\ t\ b],\\
\textit{Minimize}\quad & f(\vec{u}) = 1.10471u_1^2u_2 + 0.04811u_3u_4(14.0+u_2),\\
\textit{Subject to}\quad & g_1(\vec{u}) = \tau(\vec{u}) - \tau_{max} \leqslant 0,\\
& g_2(\vec{u}) = \sigma(\vec{u}) - \gamma_{max} \leqslant 0,\\
& g_3(\vec{u}) = \gamma(\vec{u}) - \gamma_{max} \leqslant 0,\\
& g_4(\vec{u}) = u_1 - u_4 \leqslant 0,\\
& g_5(\vec{u}) = P - P_c(\vec{u}) \leqslant 0,\\
& g_6(\vec{u}) = 0.125 - u_1 \leqslant 0,\\
& g_7(\vec{u}) = 1.10471u_1^2 + 0.04811u_3u_4(14.0+u_2)\\
& \quad - 5.0 \leqslant 0\\
\textit{Variables range}\quad & 0.1 \leqslant u_1 \leqslant 2,\\
& 0.1 \leqslant u_2 \leqslant 10,\\
& 0.1 \leqslant u_3 \leqslant 10,\\
& 0.1 \leqslant u_4 \leqslant 2\\
\textit{where}\quad & \tau(\vec{u}) = \sqrt{(\tau)^2 + 2\tau\tau\frac{u_2}{2R} + (\tau)^2},\\
& \tau = \frac{p}{\sqrt{2u_1u_2}},\quad \tau = \frac{MR}{J},\\
& M = P(L + \frac{u_2}{2}),\\
& R = \sqrt{\frac{u_2^2}{4} + (\frac{u_1+u_3}{2})^2},\\
& J = 2\left\{\sqrt{2u_1u_2}\left[\frac{u_2^2}{4} + (\frac{u_1+u_3}{2})^2\right]\right\},\\
& \sigma(\vec{u}) = \frac{6PL}{u_4u_3^2},\quad \gamma(\vec{u}) = \frac{6PL^3}{Eu_3^2u_4}\\
& P_c(\vec{u}) = \frac{4.013E\sqrt{\frac{u_3^2u_4^6}{36}}}{L^2}(1 - \frac{u_3}{2L}\sqrt{\frac{E}{4G}}),\\
& P = 6000\ \text{lb}, L = 14\ \text{in.},\quad \gamma_{max} = 0.25\ \text{in.},\\
& E = 30 \times 1^6\ \text{psi},\quad G = 12 \times 10^6\ \text{psi},\\
& \tau_{max} = 13600\ \text{psi},\quad \sigma_{max} = 30000\ \text{psi}
\end{aligned} \tag{25}
$$

The results of the developed FOCLMPA are compared with other MA collected from literature. For example, MVO [39], Ray optimization (RO) [40], Harmony search (HS) [41], Coevolutionary particle swarm optimization (CPSO) [42], Genetic algorithm (GA) [43], Davidon–Fletcher–Powell (DAVID) and Simplex method (SIMPLEX) [44], Whale optimization algorithm (WOA) [45], Gravitational search algorithm (GSA) [46], and Improved grasshopper optimization algorithm using opposition-based learning (OBLGOA) [47].

The comparison results are given in Table 8. It can be seen from this table that the FOCLMPA , with standard deviation 3.69e−07, has high ability to find the optimal parameters that reduce the fabrication costs overall other MA techniques. Followed by OBLGOA that allocates the second rank.

3.4.2. Tension/compression spring design problem

The tension/compression spring design (TCSD) problem aims to find the value of three parameters named the wire diameter (d), mean coil diameter (D) of a spring that achieves minimum spring weight, and number of active coils (N) (see Fig. 6). The TCSD optimization problem is subjected to some constraints such as surge frequency, shear stress, and minimum deflection [50]. The mathematical definition of TCSD problem is given as:

$$
\begin{aligned}
\textit{Consider}\quad & \vec{u} = [u_1\ u_2\ u_3] = [d\ D\ N],\\
\textit{Minimize}\quad & f(\vec{u}) = (u_3+2)u_2u_1^2,\\
\textit{Subject to}\quad & g_1(\vec{u}) = 1 - \frac{u_2^3u_3}{71785u_1^4} \leqslant 0,\\
& g_2(\vec{u}) = \frac{4u_2^2 - u_1u_2}{12566(u_2u_1^3 - u_1^4)} + \frac{1}{5108u_1^2} \leqslant 0,\\
& g_3(\vec{u}) = 1 - \frac{140.45u_1}{u_2^2u_3} \leqslant 0,\\
& g_4(\vec{u}) = \frac{u_1+u_2}{1.5} - 1 \leqslant 0,\\
\textit{Variables range}\quad & 0.05 \leqslant u_1 \leqslant 2\\
& 0.25 \leqslant u_2 \leqslant 1.30\\
& 2.00 \leqslant u_3 \leqslant 15
\end{aligned} \tag{26}
$$

The developed FOCLMPA is compared with Improved grasshopper optimization algorithm using opposition-based learning (OBLGOA) [47], CPSO [42], Evolutionary strategies (ES)

Table 4
The P-values based on Wilcoxon rank-sum for FOCLMPA versus the other algorithms 50 dimensional CEC17.

Fun	meas	MPA	AEO	SMA	MRFO	SCA	CLPSO	PSO	CODE
F3	p-value	0.29047	3.0199e−11	0.00039881	0.37904	3.0199e−11	3.018e−11	3.0199e−11	3.0199e−11
	h_0	0	1	1	0	1	1	1	1
F4	p-value	0.036439	0.019112	2.0023e−06	0.061452	3.018e−11	0.099253	7.3891e−11	7.3891e−11
	h_0	1	1	1	0	1	0	1	1
F5	p-value	3.0199e−11	3.0199e−11	3.0199e−11	3.0199e−11	3.0199e−11	0.0035012	1.3594e−07	1.3594e−07
	h_0	1	1	1	1	1	1	1	1
F6	p-value	0.71719	3.0199e−11	7.695e−08	3.0199e−11	3.018e−11	6.4328e−12	4.998e−09	4.998e−09
	h_0	0	1	1	1	1	1	1	1
F7	p-value	0.00238	3.0199e−11	0.9	3.0199e−11	3.018e−11	3.018e−11	0.00072951	0.00072951
	h_0	1	1	0	1	1	1	1	1
F8	p-value	4.0772e−11	3.0199e−11	4.5043e−11	3.0199e−11	3.0199e−11	5.8569e−06	8.891e−10	8.891e−10
	h_0	1	1	1	1	1	1	1	1
F9	p-value	0.00018916	3.0199e−11	3.0199e−11	3.0199e−11	3.018e−11	0.68432	1.0188e−05	1.0188e−05
	h_0	1	1	1	1	1	0	1	1
F10	p-value	0.082357	1.5964e−07	8.3486e−08	6.722e−10	3.0161e−11	2.6015e−08	9.2113e−05	9.2113e−05
	h_0	0	1	1	1	1	1	1	1
F11	p-value	0.59969	0.0083146	7.695e−08	0.0017666	3.018e−11	7.1152e−09	0.0030339	0.0030339
	h_0	0	1	1	1	1	1	1	1
F12	p-value	0.0044272	1.2057e−10	1.6057e−06	3.3384e−11	3.018e−11	0.0027545	0.29727	0.29727
	h_0	1	1	1	1	1	1	0	0
F13	p-value	0.055546	2.154e−06	3.0199e−11	0.6204	3.0199e−11	3.8202e−10	0.029205	0.029205
	h_0	0	1	1	0	1	1	1	1
F14	p-value	0.0038481	3.0199e−11	3.018e−11	3.018e−11	3.0161e−11	3.018e−11	3.0199e−11	3.0199e−11
	h_0	1	1	1	1	1	1	1	1
F15	p-value	0.00033679	3.0199e−11	3.0199e−11	1.6132e−10	3.018e−11	0.66273	8.0927e−10	8.0927e−10
	h_0	1	1	1	1	1	0	1	1
F16	p-value	0.039167	6.0658e−11	4.0772e−11	1.7769e−10	3.0199e−11	1.3853e−06	1.4918e−06	1.4918e−06
	h_0	1	1	1	1	1	1	1	1
F17	p-value	0.00042175	6.6955e−11	3.3384e−11	3.0199e−11	3.018e−11	8.838e−07	2.6015e−08	2.6015e−08
	h_0	1	1	1	1	1	1	1	1
F18	p-value	0.05012	1.8567e−09	3.0199e−11	3.0199e−11	3.018e−11	3.0199e−11	3.0199e−11	3.0199e−11
	h_0	0	1	1	1	1	1	1	1
F19	p-value	0.099258	3.0199e−11	3.018e−11	3.0199e−11	3.018e−11	1.0185e−05	1.0105e−08	1.0105e−08
	h_0	0	1	1	1	1	1	1	1
F20	p-value	0.8883	9.9186e−11	9.9186e−11	3.0199e−11	3.018e−11	7.5965e−07	1.4727e−07	1.4727e−07
	h_0	0	1	1	1	1	1	1	1
F21	p-value	5.462e−06	3.0199e−11	1.0937e−10	3.0199e−11	3.0161e−11	1.5292e−05	2.5721e−07	2.5721e−07
	h_0	1	1	1	1	1	1	1	1
F22	p-value	0.0011143	3.352e−08	1.729e−06	1.287e−09	3.0161e−11	0.61001	2.7726e−05	2.7726e−05
	h_0	1	1	1	1	1	0	1	1
F23	p-value	0.007959	3.0199e−11	3.018e−11	3.018e−11	3.0199e−11	2.1959e−07	8.8829e−06	8.8829e−06
	h_0	1	1	1	1	1	1	1	1
F24	p-value	0.21702	3.0199e−11	8.097e−10	3.018e−11	3.018e−11	3.018e−11	0.0055699	0.0055699
	h_0	0	1	1	1	1	1	1	1
F25	p-value	0.045146	9.7917e−05	0.33285	2.7726e−05	3.018e−11	0.42039	2.9215e−09	2.9215e−09
	h_0	1	1	0	1	1	0	1	1
F26	p-value	0.00095207	0.077272	9.0632e−08	0.379	3.0199e−11	0.043578	0.05012	0.05012
	h_0	1	0	1	0	1	1	0	0
F27	p-value	0.042067	3.0199e−11	0.12235	5.4941e−11	3.018e−11	0.17612	5.9706e−05	5.9706e−05
	h_0	1	1	0	1	1	0	1	1
F28	p-value	0.18577	0.0036709	1.9957e−05	0.64142	3.018e−11	1.4298e−05	1.8608e−06	1.8608e−06
	h_0	0	1	1	0	1	1	1	1
F29	p-value	0.079782	3.0199e−11	3.0199e−11	3.0199e−11	3.018e−11	0.021504	1.5292e−05	1.5292e−05
	h_0	0	1	1	1	1	1	1	1
F30	p-value	1	3.0199e−11	3.0199e−11	3.4742e−10	3.018e−11	3.3221e−06	3.0199e−11	3.0199e−11
	h_0	0	1	1	1	1	1	1	1
	$+/=/-$	16/0/12	27/0/1	25/0/3	23/0/5	28/0/0	22/0/6	2/0/2	26/0/2

Where $+/=/-$ are best/not significantly differ from the best/significantly differ from the best.

[51], MVO [39], Ray–Saini method [52], GA [38], WOA [45], CSCA [49], and mathematical method by Belegundu and Arora [53].

Table 9 shows the results of FOCLMPA and other MA techniques. One can be observed from these results that the FOCLMPA , with standard deviation 8.44e−09, has smallest cost,

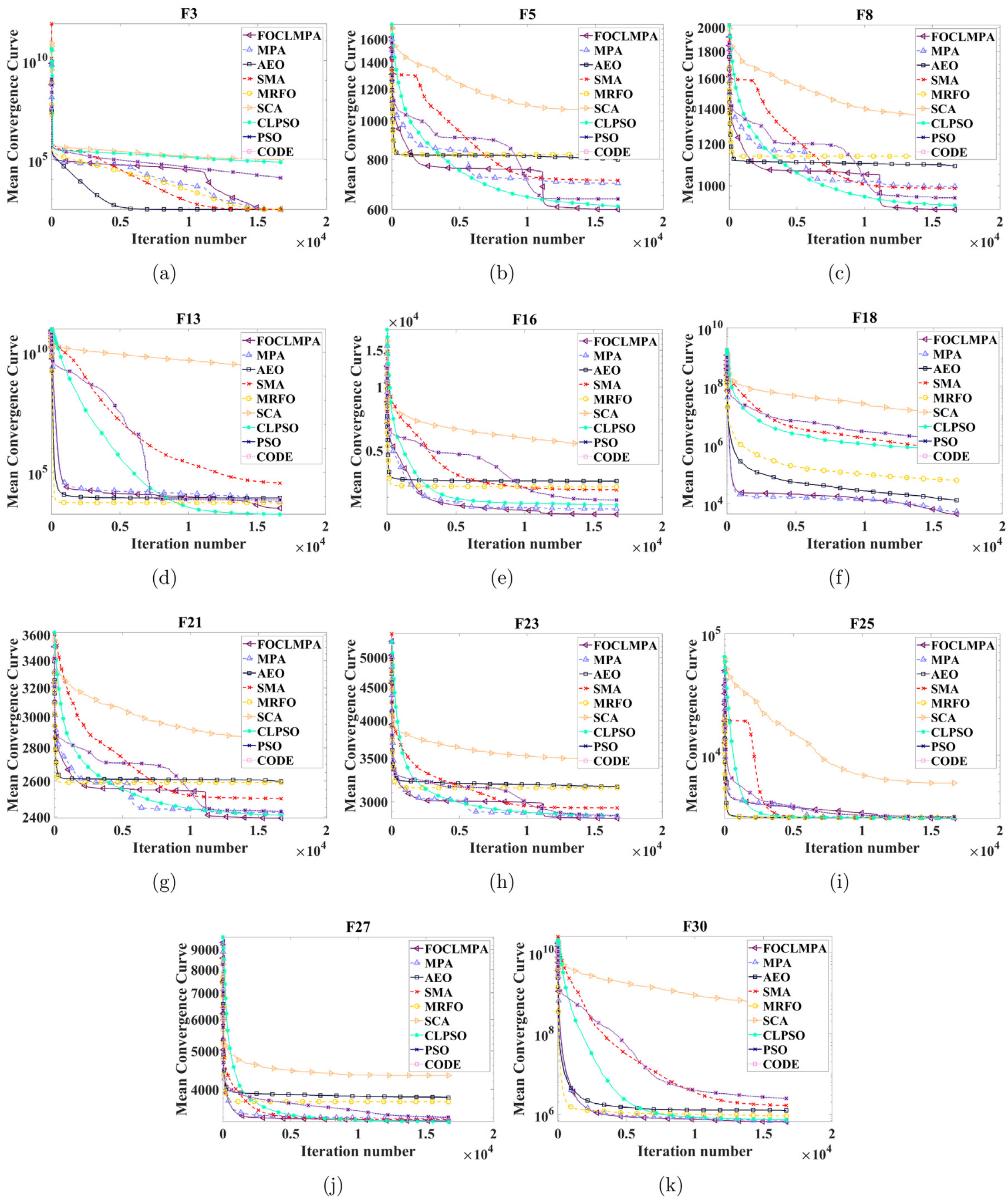

Fig. 4. Mean convergence curves by FOCLMPA, and other counterparts for 50D CEC2017 benchmarks] (a) $F3$, (b) $F5$, (c) $F8$, (d) $F13$, (e) $F16$, (f) $F18$, (g) $F21$, (h) $F23$, (i) $F25$, (j) $F27$, and (k) $F30$.

Table 5
CEC2020 benchmark functions [29].

Type	No.	Description	Fi*
Unimodal Function	1	Shifted and Rotated Bent Cigar Function (CEC 2017 F1)	100
Basic functions	2	Shifted and Rotated Schwefel's Function (CEC 2014 F11)	1100
	3	Shifted and Rotated Lunacek bi-Rastrigin Function (CEC 2017 F7)	700
	4	Expanded Rosenbrock's plus Griewangk's Function (CEC2017 f19)	1900
Hybrid functions	5	Hybrid Function 1 (N = 3) (CEC 2014 F17)	1700
	6	Hybrid Function 2 (N = 4) (CEC 2017 F16)	1600
	7	Hybrid Function 3 (N = 5) (CEC 2014 F21)	2100
Composition functions	8	Composition Function 1 (N = 3) (CEC 2017[4] F22)	2200
	9	Composition Function 2 (N = 4) (CEC 2017 F24)	2400
	10	Composition Function 3 (N = 5) (CEC 2017 F25)	2500

Table 6
Comparative analysis of FOCLMPA$_{\delta=0.55}$ for 5 and 20 dimensional CEC2020 test functions.

Function(global)		FOCLMPA	MPA	MRFO	AEO	SMA	HHO	PSO	CODE
Dimension 5									
F1(100)	Best	**1.0000e+02**	1.0000e+02	1.0000e+02	1.0000e+02	1.0001e+02	6.8106e+02	1.0235e+02	1.0000e+02
	AVR	**1.0000e+02**	1.0000e+02	1.0773e+02	1.0000e+02	3.7442e+03	8.0570e+03	7.8983e+02	2.1217e+08
	Worst	**1.0000e+02**	1.0000e+02	1.7417e+02	1.0000e+02	1.3202e+04	3.8623e+04	3.3391e+03	2.7767e+09
	STD	**0.0000e+00**	0.0000e+00	1.4397e+01	6.8374e−09	4.3661e+03	6.9096e+03	8.2179e+02	6.0121e+08
F2(1100)	Best	**1.1000e+03**	1.1000e+03	1.1001e+03	1.1000e+03	1.1071e+03	1.1072e+03	1.1068e+03	1.1001e+03
	AVR	1.1172e+03	**1.1042e+03**	1.2376e+03	1.2118e+03	1.2122e+03	1.4052e+03	1.1769e+03	1.3935e+03
	Worst	1.2260e+03	**1.1618e+03**	1.6339e+03	1.4561e+03	1.4052e+03	1.8705e+03	1.4629e+03	2.3142e+03
	STD	3.1185e+01	**1.1994e+01**	9.7613e+01	8.5331e+01	6.7687e+01	1.8715e+02	1.0799e+02	3.6942e+02
F3(700)	Best	7.0061e+02	7.0099e+02	7.0264e+02	7.0605e+02	7.0000e+02	7.0755e+02	**7.0000e+02**	7.0099e+02
	AVR	**7.0431e+02**	7.0482e+02	7.1017e+02	7.0803e+02	7.0664e+02	7.1678e+02	7.0576e+02	7.2397e+02
	Worst	7.0715e+02	**7.0604e+02**	7.1817e+02	7.1148e+02	7.1001e+02	7.2844e+02	7.0756e+02	8.4088e+02
	STD	1.9986e+00	1.5925e+00	3.6984e+00	**1.5034e+00**	2.1537e+00	5.1286e+00	1.6649e+00	2.9399e+01
F4(1900)	Best	**1.9000e+03**	1.9000e+03	1.9000e+03	1.9001e+03	1.9001e+03	1.9001e+03	1.9000e+03	1.9000e+03
	AVR	**1.9001e+03**	1.9001e+03	1.9002e+03	1.9003e+03	1.9003e+03	1.9009e+03	1.9002e+03	2.2472e+03
	Worst	**1.9002e+03**	1.9003e+03	1.9005e+03	1.9007e+03	1.9005e+03	1.9029e+03	1.9005e+03	1.1742e+04
	STD	**7.3877e−02**	7.6281e−02	1.1880e−01	1.5994e−01	1.1162e−01	7.3801e−01	1.2381e−01	1.7947e+03
F5(1700)	Best	**1.7000e+03**	1.7000e+03	1.7000e+03	1.7006e+03	1.7010e+03	1.7074e+03	1.7000e+03	1.7000e+03
	AVR	**1.7000e+03**	1.7000e+03	1.7085e+03	1.7193e+03	1.7209e+03	2.2713e+03	1.7091e+03	1.7615e+03
	Worst	**1.7000e+03**	1.7000e+03	1.7425e+03	1.7653e+03	1.7427e+03	3.5855e+03	1.8194e+03	3.4068e+03
	STD	**3.5374e-12**	1.4965e−11	1.0961e+01	1.9173e+01	1.0924e+01	4.6474e+02	3.0010e+01	3.1142e+02
F6(1600)	Best	1.6000e+03	1.6000e+03	**1.6000e+03**	1.6000e+03	1.6000e+03	1.6001e+03	1.6000e+03	1.6000e+03
	AVR	**1.6000e+03**	1.6000e+03	1.6003e+03	1.6004e+03	1.6005e+03	1.6144e+03	1.6002e+03	1.6943e+03
	Worst	**1.6000e+03**	1.6000e+03	1.6008e+03	1.6011e+03	1.6010e+03	1.6588e+03	1.6005e+03	2.2649e+03
	STD	**1.1612e−06**	3.2688e−06	2.9278e−01	3.3065e−01	2.8154e−01	1.1439e+01	1.7664e−01	1.9633e+02
F8(2200)	Best	**2.2000e+03**	2.2000e+03	2.2000e+03	2.2000e+03	2.2000e+03	2.2001e+03	2.2000e+03	2.2000e+03
	AVR	2.2034e+03	2.2167e+03	2.2082e+03	2.2014e+03	2.2020e+03	2.2747e+03	**2.2000e+03**	2.3020e+03
	Worst	2.3006e+03	2.3009e+03	2.2227e+03	2.2117e+03	2.2103e+03	2.3132e+03	**2.2000e+03**	3.3520e+03
	STD	1.8362e+01	3.8070e+01	7.2091e+00	3.6153e+00	4.1007e+00	4.0969e+01	**2.8007e−13**	2.7484e+02
F9(2400)	Best	**2.4000e+03**	2.4000e+03	2.4000e+03	2.4000e+03	2.4000e+03	2.4043e+03	2.4000e+03	2.5000e+03
	AVR	**2.4433e+03**	2.4633e+03	2.4900e+03	2.4900e+03	2.5608e+03	2.5500e+03	2.5667e+03	2.5650e+03
	Worst	**2.5000e+03**	2.5000e+03	2.5000e+03	2.5000e+03	2.6032e+03	2.6122e+03	2.7000e+03	3.4089e+03
	STD	5.0401e+01	4.9013e+01	3.0513e+01	**3.0513e+01**	5.6905e+01	5.9711e+01	1.3731e+02	1.9982e+02
F10(2500)	Best	**2.5000e+03**	2.5000e+03	2.8000e+03	2.8000e+03	2.8474e+03	2.8004e+03	2.8000e+03	2.8000e+03
	AVR	**2.7654e+03**	2.8016e+03	2.8442e+03	2.8458e+03	2.8474e+03	2.8501e+03	2.8442e+03	2.8592e+03
	Worst	**2.8474e+03**	2.8474e+03	2.8475e+03	2.8474e+03	2.8474e+03	2.8918e+03	2.8474e+03	3.1028e+03
	STD	1.1228e+02	8.6063e+01	1.2024e+01	8.6515e+00	**1.3044e−02**	1.6000e+01	1.2022e+01	5.5486e+01
Dimension 20									
F1(100)	Best	1.0000e+02	1.0000e+02	1.0000e+02	1.0000e+02	1.6574e+02	2.7958e+05	1.0001e+02	**1.0000e+02**
	AVR	2.7465e+03	1.5865e+03	**1.0000e+02**	1.0007e+02	7.0214e+03	5.9800e+05	1.6732e+02	1.4721e+09
	Worst	1.1811e+04	1.1774e+04	**1.0000e+02**	1.0216e+02	1.2154e+04	8.3406e+05	3.3223e+02	2.2815e+10
	STD	4.1385e+03	2.9752e+03	**3.6051e−04**	3.9501e−01	3.4661e+03	1.3748e+05	7.3804e+01	5.3638e+09
F2(1100)	Best	1.1052e+03	**1.1035e+03**	1.7867e+03	1.2286e+03	1.1262e+03	1.2481e+03	1.1203e+03	1.1059e+03
	AVR	**1.2032e+03**	1.2211e+03	2.6240e+03	1.4665e+03	1.5103e+03	1.7718e+03	1.4822e+03	2.6597e+03
	Worst	**1.3729e+03**	1.3807e+03	3.6937e+03	1.8578e+03	2.0288e+03	2.7874e+03	2.0631e+03	7.1923e+03
	STD	**7.7577e+01**	9.4419e+01	5.0980e+02	1.6675e+02	2.3456e+02	3.4411e+02	2.2322e+02	1.4102e+03
F3(700)	Best	7.2322e+02	7.2277e+02	7.5771e+02	7.2770e+02	7.2316e+02	7.7160e+02	**7.0266e+02**	7.2330e+02
	AVR	7.2716e+02	7.2796e+02	8.4981e+02	7.4079e+02	7.2929e+02	7.9422e+02	**7.2179e+02**	7.7085e+02
	Worst	7.3473e+02	7.3427e+02	9.2576e+02	7.5687e+02	7.3664e+02	8.3302e+02	**7.3323e+02**	1.2448e+03
	STD	2.7712e+00	**2.6296e+00**	3.8601e+01	7.1011e+00	3.6765e+00	1.6670e+01	1.0121e+01	1.1676e+02
F4(1900)	Best	1.9007e+03	1.9007e+03	1.9017e+03	1.9013e+03	1.9007e+03	1.9042e+03	**1.9005e+03**	1.9009e+03
	AVR	**1.9011e+03**	1.9011e+03	1.9043e+03	1.9036e+03	1.9012e+03	1.9138e+03	1.9015e+03	1.4661e+05
	Worst	1.9020e+03	1.9024e+03	1.9072e+03	1.9097e+03	**1.9016e+03**	1.9216e+03	1.9025e+03	4.0424e+06
	STD	3.2910e−01	4.4455e−01	1.2803e+00	1.7908e+00	**2.3298e−01**	4.5587e+00	4.0505e−01	7.3673e+05
F5(1700)	Best	1.7040e+03	1.7031e+03	1.8919e+03	2.0110e+03	3.7210e+03	3.9280e+03	2.6626e+03	**1.7024e+03**
	AVR	**1.8105e+03**	1.8180e+03	2.3989e+03	2.5930e+03	1.3273e+04	2.8422e+04	1.2921e+04	4.5830e+07
	Worst	**1.9573e+03**	2.0417e+03	3.0040e+03	3.2733e+03	1.6802e+04	1.1803e+05	1.4404e+05	6.8282e+08
	STD	**8.0593e+01**	9.0339e+01	2.7387e+02	3.2353e+02	4.0755e+03	2.7367e+04	2.5004e+04	1.5857e+08
F6(1600)	Best	1.6761e+03	**1.6025e+03**	1.6025e+03	1.6025e+03	1.6025e+03	1.7387e+03	1.6025e+03	1.6025e+03
	AVR	**1.6762e+03**	1.8192e+03	1.8192e+03	1.8192e+03	1.8192e+03	1.7805e+03	1.8192e+03	1.8192e+03
	Worst	**1.6763e+03**	2.3267e+03	2.3267e+03	2.3267e+03	2.3267e+03	1.8282e+03	2.3267e+03	2.3267e+03
	STD	**1.3236e−01**	2.1374e+02	2.1374e+02	2.1374e+02	2.1374e+02	4.5402e+01	2.1374e+02	2.1374e+02
F7(2100)	Best	**2.1002e+03**	2.1002e+03	2.1809e+03	2.1718e+03	2.3186e+03	2.8731e+03	2.1388e+03	2.1023e+03
	AVR	**2.1243e+03**	2.1417e+03	2.6459e+03	2.5438e+03	3.8725e+03	5.4349e+03	2.5657e+03	9.1516e+05
	Worst	**2.2310e+03**	2.2837e+03	3.2587e+03	3.1602e+03	6.4946e+03	1.1114e+04	3.8441e+03	1.1396e+07
	STD	**3.3499e+01**	4.6867e+01	2.4382e+02	2.4659e+02	1.3768e+03	1.7990e+03	3.5521e+02	2.8460e+06

(*continued on next page*)

Table 6 (*continued*).

Function(global)		FOCLMPA	MPA	MRFO	AEO	SMA	HHO	PSO	CODE
Dimension 5									
F8(2200)	Best	**2.2353e+03**	2.3017e+03	2.3000e+03	2.3000e+03	2.3000e+03	2.2915e+03	2.3000e+03	2.3000e+03
	AVR	2.3029e+03	2.3054e+03	**2.3011e+03**	2.3015e+03	3.9746e+03	2.5945e+03	2.9396e+03	2.5699e+03
	Worst	2.3160e+03	2.3182e+03	**2.3028e+03**	2.3043e+03	4.4908e+03	5.0117e+03	3.8809e+03	6.6236e+03
	STD	1.3403e+01	4.0757e+00	**6.9236e−01**	1.2170e+00	4.3257e+02	7.5132e+02	4.8483e+02	8.3311e+02
F9(2400)	Best	2.5000e+03	2.5000e+03	**2.4379e+03**	2.8355e+03	2.8289e+03	2.9401e+03	2.8125e+03	2.8016e+03
	AVR	2.8064e+03	**2.7880e+03**	2.9098e+03	2.9084e+03	2.8507e+03	3.1215e+03	2.8207e+03	2.8633e+03
	Worst	**2.8297e+03**	2.8346e+03	3.0148e+03	2.9735e+03	2.8727e+03	3.3312e+03	2.8470e+03	3.0822e+03
	STD	5.8223e+01	9.7981e+01	1.1079e+02	3.4895e+01	1.2047e+01	9.5900e+01	**9.6248e+00**	8.9213e+01
F10(2500)	Best	2.8998e+03	2.8994e+03	2.9114e+03	2.9010e+03	2.9100e+03	2.9142e+03	**2.8992e+03**	2.9137e+03
	AVR	2.9177e+03	2.9178e+03	2.9654e+03	2.9643e+03	**2.9132e+03**	2.9739e+03	2.9193e+03	2.9412e+03
	Worst	2.9642e+03	2.9936e+03	3.0078e+03	3.0133e+03	**2.9139e+03**	3.0159e+03	2.9605e+03	3.1645e+03
	STD	1.4871e+01	1.8842e+01	3.2780e+01	4.0891e+01	**1.4034e+00**	2.7257e+01	1.3624e+01	7.0671e+01

Black bold font refers to the best result (minimum AVR).

Table 7
Ranking analysis of FOCLMPA and other peers for 5 and 20 dimensional CEC2020 test functions.

Function	FOCLMPA	MPA	MRFO	AEO	SMA	HHO	PSO	CODE
Dimension 5								
F1	1	2	4	3	6	7	5	8
F2	2	1	6	4	5	8	3	7
F3	1	2	6	5	4	7	3	8
F4	1	2	3	6	5	7	4	8
F5	1	2	3	5	6	8	4	7
F6	1	2	4	5	6	7	3	8
F8	4	6	5	2	3	7	1	8
F9	1	2	3	4	6	5	8	7
F10	1	2	4	5	6	7	3	8
Average rank	**1.6000**	2.1000	4.1000	4.3000	5.2000	6.9000	4.1000	7.7000
Dimension 20								
F1	5	4	1	2	6	7	3	8
F2	1	2	7	3	5	6	4	8
F3	2	3	8	5	4	7	1	6
F4	1	2	6	5	3	7	4	8
F5	1	2	3	4	6	7	5	8
F6	1	3	4	5	6	2	7	8
F7	1	2	5	3	6	7	4	8
F8	3	4	1	2	8	6	7	5
F9	2	3	7	6	1	8	4	5
Average rank	**1.9991**	2.5008	4.9000	4.1000	4.9000	6.5000	4.2000	6.9000

Table 8
Comparison of optimization results for the WBD problem.

Algorithm	H	L	t	b	Optimized cost
FOCLMPA	0.2057293	3.470495	9.03662	0.2057296	1.7248533
HHO [48]	0.204039	3.531061	9.027463	0.206147	1.7319905
OBLGOA [47]	0.205769	3.471135	9.032728	0.2059072	1.7257
RO [40]	0.203687	3.528467	9.004233	0.207241	1.735344
HS [41]	0.2442	6.2231	8.2915	0.2443	2.3807
DAVID [44]	0.2434	6.2552	8.2915	0.2444	2.3841
SIMPLEX [44]	0.2792	5.6256	7.7512	0.2796	2.5307
CPSO [42]	0.202369	3.544214	9.04821	0.205723	1.72802
MVO [39]	0.205463	3.473193	9.044502	0.205695	1.72645
GA [43]	0.205986	3.471328	9.020224	0.20648	1.728226
GSA [46]	0.182129	3.856979	10	0.202376	1.87995
CSCA [49]	0.203137	3.542998	9.033498	0.206179	1.733461
WOA [45]	0.205396	3.484293	9.037426	0.206276	1.730499

followed by HHO, and MFO that allocate second and third rank, respectively.

3.4.3. Pressure vessel design (PVD) problem

In this engineering problem known as pressure vessel design (PVD), the main target is to find the parameters that reduce total cost of welding, material and forming (see Fig. 7). There are four parameters including the thickness of the head *Th*, the inner radius *R*, the thickness *Ts*, and the length of the cylindrical section of the vessel *L*. The PVD is subjected to four constraints [50].

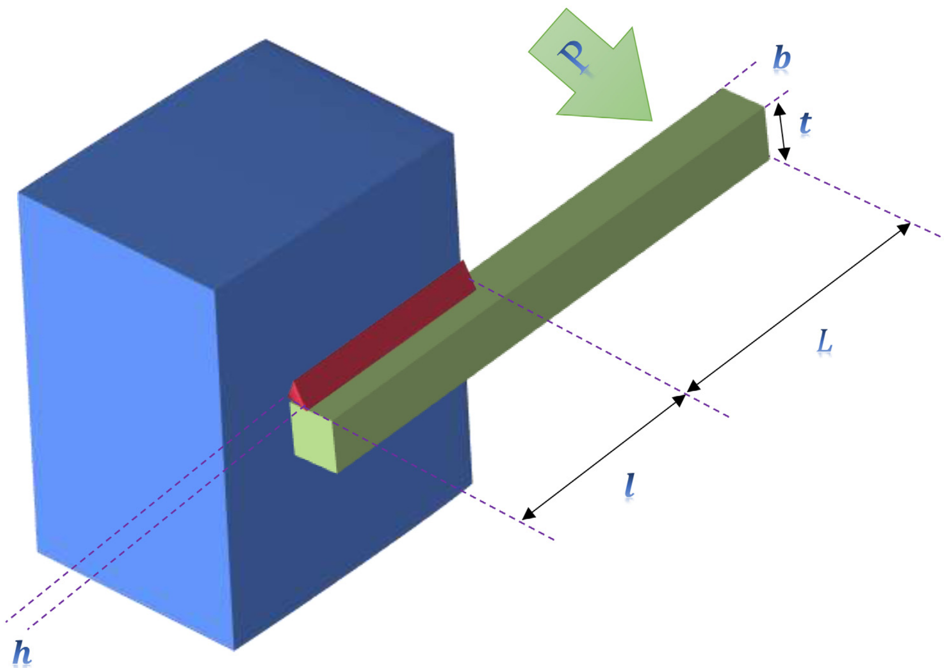

Fig. 5. Structure of WBD problem.

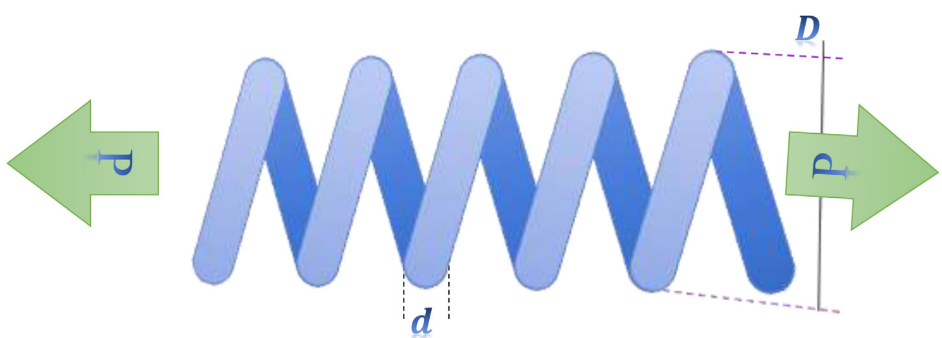

Fig. 6. Design of TCSD problem.

Table 9
Comparison of optimization results for the TCSD problem.

Algorithm	d	D	N	Optimized cost
FOCLMPA	0.05164743	0.355717	11.34788	0.01266527
HHO [48]	0.051796393	0.359305355	11.138859	0.012665443
MFO [54]	0.051994457	0.36410932	10.868421862	0.0126669
OBLGOA [47]	0.0530178	0.38953229	9.6001616	0.01270136
Belegundu–Arora method [53]	0.0500	0.3177	14.026	0.012730
GA [38]	0.05148	0.351661	11.632201	0.01270478
WOA [45]	0.051207	0.345215	12.004032	0.0126763
CPSO [42]	0.051728	0.357644	11.244543	0.0126747
ES [51]	0.051643	0.35536	11.397926	0.012698
MVO [39]	0.05251	0.37602	10.33513	0.012790
GSA [45]	0.050276	0.323680	13.525410	0.0127022
Ray–Saini method [52]	0.321532	0.050417	13.979915	0.013060

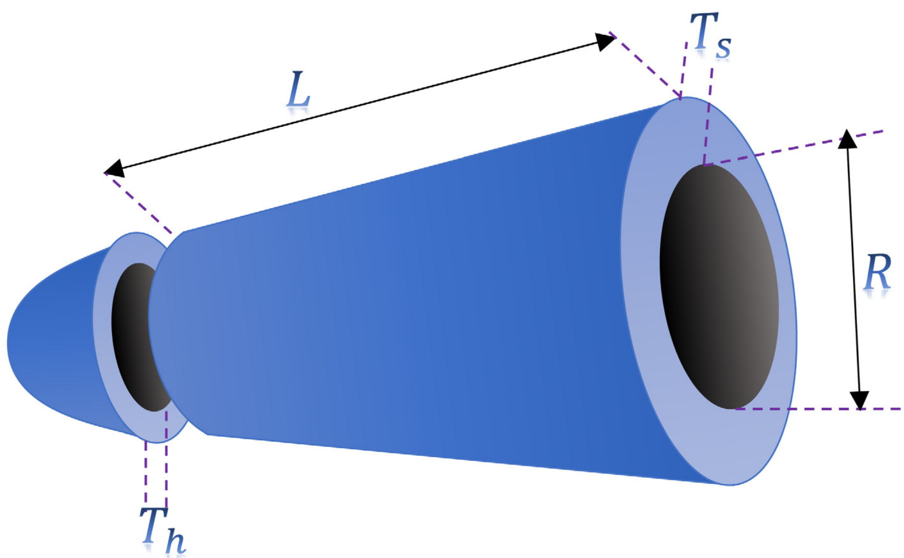

Fig. 7. Schematic of the PVD problem.

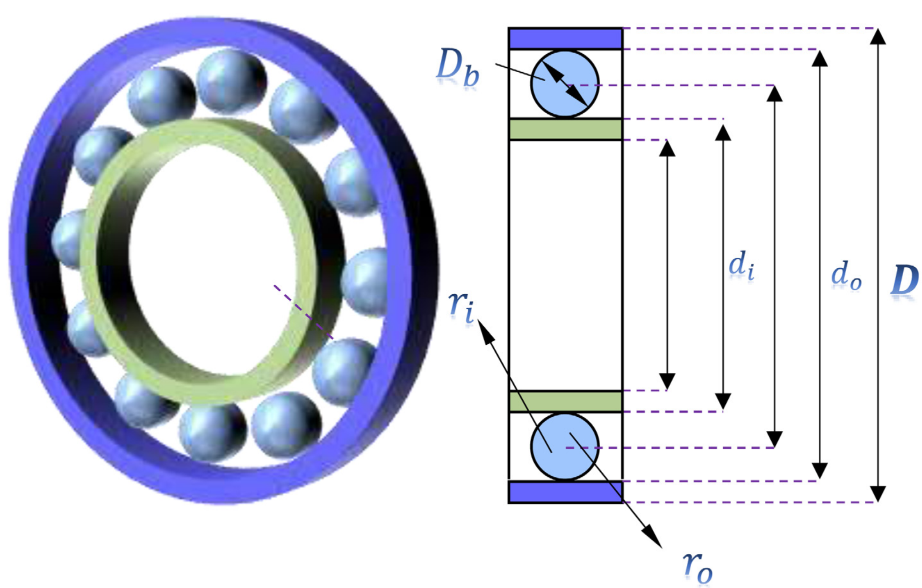

Fig. 8. Structure of the REBD design.

The mathematical formula of PVD problem and constraints are exposed in Eq. (27) [55].

$$
\begin{aligned}
\textit{Consider} \quad & \vec{u} = [u_1\ u_2\ u_3\ u_4] = [T_s\ T_h\ R\ L], \\
\textit{Minimize} \quad & 0.6224 u_1 u_3 u_4 + 1.7781 u_2 u_3^2 \\
& + 3.1661 u_1^2 u_4 + 19.84 u_1^2 u_3, \\
\textit{Subject to} \quad & g_1(\vec{u}) = -u_1 + 0.0193 u_3 \leqslant 0, \\
& g_2(\vec{u}) = -u_2 + 0.00954 u_3 \leqslant 0, \\
& g_3(\vec{u}) = -\pi u_3^2 u_4 - \frac{4}{3}\pi u_3^3 + 1296000 \leqslant 0, \\
& g_4(\vec{u}) = u_4 - 240 \leqslant 0, \\
\textit{Variables range} \quad & 0 \leqslant u_1 \leqslant 99, \\
& 0 \leqslant u_2 \leqslant 99, \\
& 10 \leqslant u_3 \leqslant 200, \\
& 10 \leqslant u_4 \leqslant 200
\end{aligned}
\tag{27}
$$

The results of FOCLMPA are compared with OBLGOA [47], HPSO [56], GSA [45], PSO-DE [57], ACO [58], CDE [49], GA [38], and ES [51]. Table 10 shows the results of FOCLMPA algorithm and other approaches. It can be observed that FOCLMPA , with standard deviation 0.0015, outperforms the other methods OBLGOA [47].

3.4.4. Rolling element bearing design (REBD) problem

In the rolling element bearing design (REBD) problem, there are ten variables that required to determine their values for maximizing the dynamic load carrying capacity (as in Fig. 8). The REBD is subject to nine constraints that are considered for assembly and geometric-based restrictions. The considered objective function is maximizing the dynamic capacity (dynamic load rating) (C_d) that directly formed the basis for longest fatigue life of a bearing. The mathematical definition of REBD problem and constraints are formulated as in Eq. (28) [59].

$$
\begin{aligned}
\textit{Maximize}\quad & C_d = f_c U^{2/3} D_b^{1.8},\ \textit{if } D \le 25.4\text{ mm},\\
& C_d = 3.647 f_c U^{2/3} D_b^{1.4},\ \textit{if } D \le 25.4\text{ mm},\\
\textit{Subject to}\quad & g_1(\vec{c}) = \frac{\phi_0}{2sin^{-1}(D_b/D_m)} - U + 1 \leqslant 0,\\
& g_2(\vec{u}) = 2D_b - K_{Dmin}(D-d) > 0,\\
& g_3(\vec{u}) = K_{Dmax}(D-d) - 2D_b > 0,\\
& g_4(\vec{u}) = \xi B_w - D_b \leqslant 0,\\
& g_5(\vec{u}) = D_m - 0.5(D-d) \ge 0,\\
& g_6(\vec{u}) = (0.5+e)(D+d) - D_m \ge 0,\\
& g_7(\vec{u}) = 0.5(D - D_m - D_b) - \epsilon D_b \ge 0,\\
& g_8(\vec{u}) = f_i \ge 0.515,\\
& g_9(\vec{u}) = f_0 \ge 0.515,\\
\textit{where}\quad & f_c = 37.91\left[1 + \left\{1.64\left(\frac{1-\gamma}{1-\gamma}\right)^{1.72}\right.\right.\\
& \left.\left.\left(\frac{f_i(2f_0-1)}{f_0(2f_i-1)}^{0.41}\right)\right\}^{10/3}\right]^{-0.3}\\
& \times\left[\frac{\gamma^{0.3}(1-\gamma)^{0.39}}{(1-\gamma)^{1/3}}\right]\left[\frac{2f_i}{2f_i-1}\right]^{0.41},\\
& x = [\{(D-d)/2 - 3(T/4)\}^2 + \{D/2 - T/4 - D_b\}^2\\
& \quad - \{d/2 + T/4\}^2],\\
& y = 2\{(D-d)/2 - 3(T/4)\}\{D/2 - T/4 - D_b\},\\
& \phi_0 = 2\pi - cos^{-1}\left(\frac{x}{y}\right),\\
& \gamma = \frac{D_b}{D_m}, f_i = \frac{r_i}{D_b}, f_0 = \frac{r_0}{D_b}, T = D - d - 2D_b,\\
& D = 160, d = 90,\\
& B = 30, r_i = r_0 = 11.033, 0.5(D+d) \leqslant D_m\\
& \quad \leqslant 0.6(D+d),\\
& 0.15(D-d) \leqslant D_b \leqslant 0.45(D-d), 4 \leqslant Z \leqslant 50,\\
& 0.515 \leqslant f_i and f_o \leqslant 0.6,\\
& 0.4 \leqslant K_{Dmin} \leqslant 0.5,\quad 0.6 \leqslant K_{Dmax} \leqslant 0.7,\\
& 0.3 \leqslant \epsilon \leqslant 0.4,\quad 0.02 \leqslant e \leqslant 0.1,\quad 0.6 \leqslant \xi \leqslant 0.85
\end{aligned} \tag{28}
$$

The experimental results obtained by FOCLMPA are compared with other methods including MVO, PVs [60], CMVOHHO [61], HHO [48], and TLBO [62]. Table 11 shows the results obtained to solve REBD problems, by inspecting the obtained objective function values, one can conclude that the FOCLMPA achieves the highest loading carrying capacity, with standard deviation of 0.0034507, in comparison to the other techniques. Hence, it should be considered the best optimizer for this problem, followed by CMVOHHO and MVO in the second and third ranks, respectively.

3.5. Real-world application: Feature selection

To test the performance of the developed FOCLMPA using real-world application, it is implemented as feature selection method. In this application, the main goal is to reduce the number of features by removing the irrelevant and redundant features with influence on the quality of the classification.

The proposed FOCLMPA as FS method is different from its version discussed in previous experiments. Since, FS is a discrete problem, so the current solution is converted into binary solution before computing the fitness value and this achieved by using the following equation.

$$BZ_{ij} = \begin{cases} 1 & if\ Z_{ij} > 0.5 \\ 0 & otherwise \end{cases} \tag{29}$$

Then the following equation is used to compute the fitness value for each solution.

$$Fit_i = \lambda \times \gamma_i + \mu \times \left(\frac{|BZ_i|}{Dim}\right) \tag{30}$$

where Dim denotes the number of features in the tested dataset and $|BZ_i|$ is the number of selected features. The λ and μ ($\lambda + \mu = 1$) denote parameters applied to balance between number of selected features and error of classification (γ_i). In this study, the K-nearest neighbour (KNN) classifier is used since it does not need extra parameters. In addition, the dataset is divided into training set and testing set which represent 80% and 20% from the dataset, respectively.

3.5.1. Datasets description

In the experiments of FS, a set of eighteen UCI [63] are used to assess the performance of the developed method. The general description of those datasets are given in Table 12 and it can be seen that these datasets are collected from different real-life areas. In addition, they contains different number of classes and variants samples and number of features.

3.5.2. Comparative algorithms

The quality of the selected features obtained by the developed FOCLMPA is compared with other well-known MA techniques that have been applied as FS methods. These methods including AEO, MRFO, MPA, Harris hawks optimization (HHO) [48], Henry gas solubility optimization (HGSO) [64], Whale optimization algorithm (WOA) [45], and Grey wolf optimizer (GWO) [65], Genetic Algorithm (GA) [66], PSO, Salp Swarm Algorithm (SSA) [67], and TLBO. We used these methods since their performances have been evaluated in literature and they established high performance. The parameters of those methods are set according to original implementation. However, the common parameters such as number of populations and total number of iterations are 20 and 30, respectively. For achieving an unbiased comparison, each method conducted 25 independent times.

3.5.3. Performance measures

There are several measures are used to assess the performance of the FS methods. These measures are accuracy (Avg_{acc}), Standard deviation (STD), Average of selected features (($AVG_{|BX_{Best}|}$)), and Average of fitness value ((AVG_{Fit})). The definition of these measures are given as follows:

$$Avg_{acc} = \frac{1}{N_r}\sum_{k=1}^{N_r} Acc_{Best}^k,\ Acc_{Best}^k = \frac{TP+TN}{TP+FN+FP+TN} \tag{31}$$

where TP, TN, FN, and FP denotes the true positive, true negative, false negative, and false positive, respectively. N_r total number of runs.

$$STD = \sqrt{\frac{1}{N_r}\sum_{k=1}^{N_r}\left(Acc_{Best}^k - Avg_{acc}\right)^2} \tag{32}$$

$$AVG_{|BX_{Best}|} = \frac{1}{N_r}\sum_{k=1}^{N_r}\left|BX_{Best}^k\right| \tag{33}$$

where $|.|$ is the number of elements in BX_{Best}^k at run k.

$$AVG_{Fit} = \frac{1}{N_r}\sum_{k=1}^{N_r} Fit_{Best}^k \tag{34}$$

Table 10
Comparison of optimization results for the PVD problem.

	Optimal values for variables				
Algorithm	*Th*	*Ts*	*R*	*L*	Optimized cost
FOCLMPA	0.7781719	0.38465	40.31978	199.99770	5885.34045
HHO [48]	0.81758383	0.4072927	42.09174576	176.7196352	6000.46259
MFO [54]	0.8125	0.4375	42.098445	176.636596	6059.7143
OBLGOA [47]	0.81622	0.40350	42.291138	174.811191	5966.67160
PSO-DE [57]	0.8125	0.4375	42.098446	176.6366	6059.71433
HPSO [56]	0.8125	0.4375	42.0984	176.6366	6059.7143
ACO [58]	0.8125	0.4375	42.098353	176.637751	6059.7258
CDE [49]	0.8125	0.4375	42.098411	176.63769	6059.734
ES [51]	0.8125	0.4375	42.098087	176.640518	6059.7456
GA [38]	0.8125	0.4375	42.097398	176.65405	6059.94634
GSA [45]	1.125	0.625	55.9886598	84.4542025	8538.8359

Table 11
Comparison results between variants of FOCLMPA and others methods for REBD problem.

Variable	CMVHHO	MVO	HHO	TLBO	PVS	FOCLMPA
D_m	125.01812	126.1035	125.00000	125.7191	125.7191	125.7225
D_b	21.27836	21.03794	21.000000	21.42559	21.42559	21.4233
Z	10.86415	11.13977	11.092073	11.00000	11.00000	11.001
f_i	0.515000	0.51500	0.515000	0.515000	0.515000	0.515
f_o	0.518045	0.52286	0.515000	0.515000	0.515000	0.515
K_{dmin}	0.456659	0.44355	0.400000	0.424266	0.400430	0.48975
K_{dmax}	0.639656	0.67521	0.600000	0.633948	0.680160	0.694310
ϵ	0.301777	0.30108	0.300000	0.300000	0.300000	0.30000081
e	0.021893	0.08852	0.050474	0.068858	0.079990	0.06102046
ξ	0.646298	0.61468	0.600000	0.799498	0.700000	0.65383051
Dynamic load carrying capacity	83 800.078	83 535.147	83 011.883	81 859.74	81 859.741	85 539.0405265724

Table 12
Datasets description.

Datasets	Number of features	Number of instances	Number of classes	Data category
Breastcancer (DS1)	9	699	2	Biology
BreastEW (DS2)	30	569	2	Biology
CongressEW (DS3)	16	435	2	Politics
Exactly (DS4)	13	1000	2	Biology
Exactly2(DS5)	13	1000	2	Biology
HeartEW (DS6)	13	270	2	Biology
IonosphereEW (DS7)	34	351	2	Electromagnetic
KrvskpEW (DS8)	36	3196	2	Game
Lymphography (DS9)	18	148	2	Biology
M-of-n (DS10)	13	1000	2	Biology
PenglungEW (DS11)	325	73	2	Biology
SonarEW (DS12)	60	208	2	Biology
SpectEW (DS13)	22	267	2	Biology
Tic-tac-toc (DS14)	9	958	2	Game
Vote (DS15)	16	300	2	Politics
WaveformEW (DS16)	40	5000	3	Physics
WineEW (DS17)	13	178	3	Chemistry
Zoo (DS18)	16	101	6	Artificial

Table 13
Average of accuracy obtained by each algorithm.

fn	FOCLMPA	AEO	MRFO	MPA	HHO	HGSO	WOA	GWO	GA	PSO	SSA
Breastcancer4	0.976	**0.982**	0.962	0.956	0.964	0.975	0.948	0.957	0.946	0.971	0.979
BreastEW	0.973	0.975	0.976	0.981	0.973	0.933	0.943	0.942	0.935	**0.985**	0.972
CongressEW	0.974	0.966	0.972	**0.992**	0.964	0.985	0.945	0.916	0.961	0.972	0.959
Exactly	**1.000**	**1.000**	0.999	0.999	0.972	0.974	0.896	0.899	0.867	0.980	0.959
Exactly2	0.756	0.736	0.765	0.744	0.745	0.726	0.770	**0.790**	0.713	0.740	0.765
HeartEW	0.886	0.857	0.865	0.905	**0.906**	0.891	0.799	0.827	0.875	0.853	0.898
IonosphereEW	**0.991**	0.962	0.968	0.941	0.927	0.912	0.917	0.938	0.958	0.940	0.970
KrvskpEW	0.976	0.974	**0.978**	0.972	0.973	0.949	0.951	0.958	0.962	0.964	0.966
Lymphography	0.963	0.966	0.929	0.954	0.913	0.932	0.882	0.876	0.884	0.889	**0.969**
M-of-n	**1.000**	0.999	0.999	**1.000**	0.995	0.985	0.950	0.963	0.948	**1.000**	0.964
PenglungEW	**1.000**	**1.000**	**1.000**	0.938	0.956	**1.000**	0.969	0.982	0.867	**1.000**	**1.000**
SonarEW	**1.000**	0.948	0.957	0.968	0.957	0.956	0.983	0.938	0.994	0.938	0.919
SpectEW	**0.956**	0.847	0.851	0.879	0.931	0.915	0.759	0.772	0.858	0.896	0.810
Tic-tac-toe3	0.839	0.838	0.823	**0.856**	0.818	0.810	0.781	0.782	0.825	0.807	0.799
Vote	0.987	0.966	**0.998**	0.962	0.967	0.984	0.962	0.970	0.960	0.997	0.957
WaveformEW	0.762	**0.776**	0.766	0.759	0.729	0.728	0.728	0.719	0.753	0.753	0.738
WineEW	**1.000**	**1.000**	**1.000**	**1.000**	0.998	0.998	0.976	0.983	0.983	**1.000**	**1.000**
Zoo	**1.000**	0.968	**1.000**	**1.000**	**1.000**	**1.000**	0.997	0.984	**1.000**	**1.000**	**1.000**

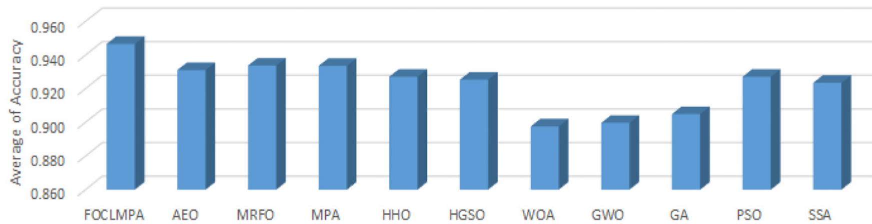

Fig. 9. Average of accuracy.

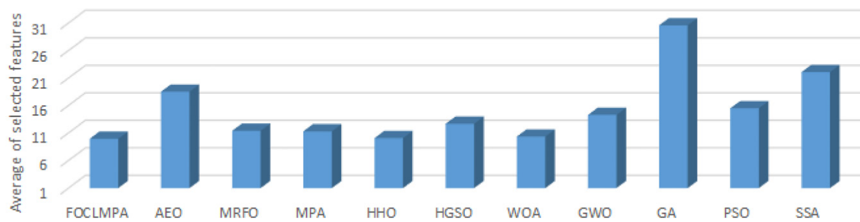

Fig. 10. Average of selected features.

Table 14
Average of selected features.

fn	FOCLMPA	AEO	MRFO	MPA	HHO	HGSO	WOA	GWO	GA	PSO	SSA
Breastcancer4	2.867	3.733	3.067	2.933	**1.800**	3.400	2.400	2.600	4.800	3.000	3.267
BreastEW	14.000	14.467	7.200	6.800	8.600	9.333	**5.667**	8.467	20.867	10.267	15.733
CongressEW	5.100	5.933	3.533	4.933	6.733	**2.733**	3.867	5.067	10.667	5.933	8.467
Exactly	**6.300**	6.400	6.933	6.400	6.733	8.067	8.600	6.533	9.400	6.500	8.267
Exactly2	2.567	5.533	2.500	4.000	2.500	5.667	2.333	**1.400**	9.400	2.500	8.200
HeartEW	**3.533**	8.133	5.667	5.733	5.800	4.200	4.533	6.267	10.867	5.467	6.867
IonosphereEW	14.367	19.533	**8.067**	9.400	9.467	8.933	8.467	8.800	28.067	11.000	19.067
KrvskpEW	**14.267**	22.400	19.667	14.533	19.067	17.800	19.000	20.667	28.867	16.200	22.933
Lymphography	11.600	10.067	12.200	9.133	9.600	7.133	**4.067**	8.000	14.000	7.400	9.933
M-of-n	6.400	6.600	6.533	6.467	8.467	8.267	9.400	8.600	9.200	**6.133**	8.733
PenglungEW	50.300	128.000	48.667	58.133	67.200	77.733	**40.533**	106.800	267.267	134.600	185.333
SonarEW	**24.233**	35.200	26.933	29.467	24.800	25.933	30.933	24.467	50.000	28.667	35.333
SpectEW	8.900	12.067	4.667	8.067	8.333	7.667	**3.733**	6.533	16.933	7.867	12.600
Tic-tac-toe3	7.000	5.800	6.467	6.133	5.933	**4.667**	5.400	5.267	6.333	5.000	5.867
Vote	4.433	6.067	5.733	4.733	4.867	5.200	6.867	4.200	11.067	**3.067**	6.933
WaveformEW	**15.467**	30.133	25.933	18.733	18.533	19.867	22.000	19.733	34.200	17.333	26.800
WineEW	**4.133**	6.467	5.133	5.000	6.200	5.267	6.267	5.467	9.467	4.333	6.600
Zoo	3.667	6.667	6.267	**2.667**	5.600	6.200	8.067	8.267	9.000	3.733	6.867

Table 15
Average of fitness value obtained by each algorithm.

fn	FOCLMPA	AEO	MRFO	MPA	HHO	HGSO	WOA	GWO	GA	PSO	SSA
Breastcancer4	**0.053**	0.058	0.068	0.072	0.071	0.060	0.074	0.068	0.102	0.059	0.055
BreastEW	0.071	0.071	0.046	**0.040**	0.064	0.091	0.070	0.081	0.128	0.047	0.078
CongressEW	0.055	0.068	0.048	0.038	0.036	**0.030**	0.074	0.108	0.102	0.062	0.089
Exactly	0.048	0.049	0.054	0.050	0.061	0.085	0.160	0.141	0.192	0.066	0.101
Exactly2	0.240	0.280	0.219	0.261	0.216	0.290	0.217	**0.200**	0.331	0.242	0.274
HeartEW	0.168	0.191	0.165	**0.130**	0.155	0.130	0.216	0.204	0.196	0.174	0.145
IonosphereEW	0.051	0.091	0.052	0.081	0.042	0.106	0.099	0.082	0.121	0.086	0.083
KrvskpEW	0.067	0.086	0.075	**0.066**	0.074	0.095	0.097	0.095	0.115	0.077	0.095
Lymphography	0.097	0.087	0.132	0.092	0.106	0.101	0.129	0.156	0.182	0.141	0.084
M-of-n	0.049	0.051	0.051	0.050	0.054	0.077	0.118	0.100	0.118	**0.047**	0.100
PenglungEW	0.026	0.039	0.015	0.074	**0.012**	0.024	0.041	0.049	0.202	0.041	0.057
SonarEW	**0.050**	0.106	0.083	0.078	0.069	0.083	0.067	0.096	0.089	0.103	0.132
SpectEW	**0.080**	0.193	0.156	0.146	0.158	0.112	0.234	0.235	0.205	0.129	0.228
Tic-tac-toe3	0.223	0.211	0.232	**0.198**	0.236	0.223	0.257	0.255	0.228	0.229	0.246
Vote	0.039	0.069	0.038	0.064	0.039	0.047	0.046	0.053	0.105	**0.022**	0.082
WaveformEW	0.270	0.277	0.275	0.264	0.266	0.294	0.300	0.303	0.308	0.266	0.303
WineEW	0.039	0.050	0.039	0.038	0.043	0.042	0.070	0.057	0.088	0.033	0.051
Zoo	0.023	0.070	0.039	**0.017**	0.027	0.039	0.053	0.066	0.056	0.023	0.043

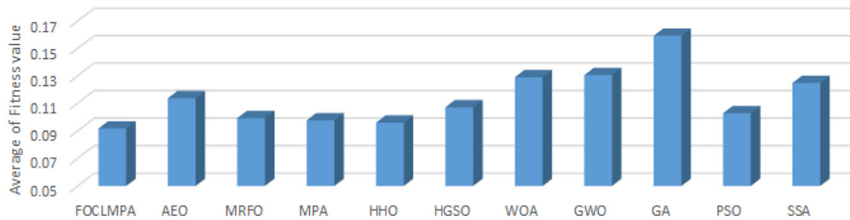

Fig. 11. Average of fitness value.

3.5.4. Results and discussion

The comparison of results between the developed FOCLMPA with other methods are given in Tables 13–15. Table 13 shows the average of the accuracy values obtained by each algorithm overall the total number of runs. From these recorded results it can be observed that developed FOCLMPA has the higher accuracy value at three datasets, namely, IonosphereEW, SpectEW, and SonarEW. In addition, it has the same accuracy with other algorithms at five datasets, namely, Exactly, M-of-n, PenglungEW, WineEW, and Zoo. However, AEO, MRFO, and MPA allocates the second rank in terms of accuracy which has two high accuracy values. In addition, by observed the average of each algorithm overall the tested dataset, as in Fig. 9, it can be noticed the superiority of FOCLMPA over the comparison algorithms.

To validate the ability of the developed FOCLMPA to reduce the number of features by removing the irrelevant features, the average of selected features is given Table 14. From this table, it can be noticed that FOCLMPA finds the smallest number of features at six datasets including Exactly, HeartEW, KrvskpEW, SonarEW, WaveformEW, and WineEW. Followed by WOA which has smallest number features at four datasets BreastEW, Lymphography, PenglungEW, and SpectEW. Also, PSO and HGSO have ability to find the smallest number of features at two datasets. Fig. 10 shows the average of number of selected features overall the tested datasets. One can be seen that FOCLMPA has better average among all the algorithms followed by HHO. This indicates the high ability of FOCLMPA to select the relevant features and remove the irrelevant features without influence on the quality of the original data.

Table 15 shows the average of fitness value for each algorithm overall the given datasets. It can be seen that the FOCLMPA has the smallest fitness value at three datasets. Also, the traditional MPA has better fitness value at five datasets. However, by analysis the average of fitness value overall datasets , as in Fig. 11, it can be observed that FOCLMPA has best average followed by HHO and MPA, respectively.

4. Conclusion and future works

In this paper, a modified Marine Predator Algorithm (MPA) is introduced based on the concept of comprehensive learning strategy and a memory perspective of the fractional calculus. This modification aims to enhance the ability of MPA to balance exploration and exploitation. In addition, sharing the history of the best knowledge support of MPA by a suitable tool to avoid attraction to the local point. To validate the performance of developed fractional-order comprehensive learning MPA (FOCLMPA), different experiments were conducted. For example, a set of global optimization functions from CEC2017 and CEC2020 have been used to assess its ability to find comparable solutions in different landscapes and with different characteristics. In addition, the applicability of FOCLMPA has been tested using real-world engineering problems and in feature selection using complex datasets. The experimental results provide evidence of the high performance of the developed FOCLMPA over other MA techniques via using several statistical and nun-parametric analysis.

According to the wonderful results obtained by using the developed FOCLMPA, it open a new trend in future works to combine the comprehensive learning strategy and memory perspective of the fractional calculus. In addition, the proposed FOCLMPA can be applied in other fields such as image segmentation, task scheduling in fog computing, security, data clustering, and power applications.

Declaration of competing interest

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

Acknowledgements

The authors extend their appreciation to the researcher supporting project at King Saud University, Riyadh, Saudi Arabia for funding this research work through the project number RSP-2020/278.

References

[1] K. Hussain, M.N.M. Salleh, S. Cheng, Y. Shi, Metaheuristic research: a comprehensive survey, Artif. Intell. Rev. 52 (4) (2019) 2191–2233.
[2] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95-International Conference on Neural Networks, Vol. 4, IEEE, 1995, pp. 1942–1948.
[3] D.E. Goldberg, Genetic Algorithms, Pearson Education India, 2006.
[4] R. Storn, K. Price, Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (4) (1997) 341–359.
[5] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1) (1997) 67–82.
[6] Y.-C. Ho, D.L. Pepyne, Simple explanation of the no free lunch theorem of optimization, Cybernet. Systems Anal. 38 (2) (2002) 292–298.
[7] A. Faramarzi, M. Heidarinejad, S. Mirjalili, A.H. Gandomi, Marine predators algorithm: A nature-inspired Metaheuristic, Expert Syst. Appl. (2020) 113377.
[8] M. Abd Elaziz, A.A. Ewees, D. Yousri, H.S.N. Alwerfali, Q.A. Awad, S. Lu, M.A. Al-qaness, An improved marine predators algorithm with fuzzy entropy for multi-level thresholding: Real world example of COVID-19 CT image segmentation, IEEE Access (2020).
[9] M. Abdel-Basset, R. Mohamed, M. Elhoseny, R.K. Chakrabortty, M. Ryan, A hybrid COVID-19 detection model using an improved marine predators algorithm and a ranking-based diversity reduction strategy, IEEE Access 8 (2020) 79521–79540.
[10] M.A. Al-Qaness, A.A. Ewees, H. Fan, L. Abualigah, M. Abd Elaziz, Marine predators algorithm for forecasting confirmed cases of COVID-19 in Italy, USA, Iran and Korea, Int. J. Environ. Res. Public Health 17 (10) (2020) 3520.
[11] M. Abdel-Basset, R. Mohamed, M. Elhoseny, A.K. Bashir, A. Jolfaei, N. Kumar, Energy-aware marine predators algorithm for task scheduling in IoT-based fog computing applications, IEEE Trans. Ind. Inf. (2020).
[12] D. Yousri, T.S. Babu, E. Beshr, M. Eteiba, D. Allam, A robust strategy based on marine predators algorithm for large scale photovoltaic array reconfiguration to mitigate the partial shading effect on the performance of PV system, IEEE Access (2020).

[13] D. Yousri, H.M. Hasanien, A. Fathy, Parameters identification of solid oxide fuel cell for static and dynamic simulation using comprehensive learning dynamic multi-swarm marine predators algorithm, Energy Convers. Manage. 228 (2021) 113692.
[14] D. Yousri, M. Abd Elaziz, D. Oliva, L. Abualigah, M.A. Al-qaness, A.A. Ewees, Reliable applied objective for identifying simple and detailed photovoltaic models using modern metaheuristics: Comparative study, Energy Convers. Manage. 223 (2020) 113279.
[15] J.J. Liang, A.K. Qin, P.N. Suganthan, S. Baskar, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions, IEEE Trans. Evol. Comput. 10 (3) (2006) 281–295.
[16] A. Lin, W. Sun, H. Yu, G. Wu, H. Tang, Adaptive comprehensive learning particle swarm optimization with cooperative archive, Appl. Soft Comput. 77 (2019) 533–546.
[17] J. Sabatier, O.P. Agrawal, J.T. Machado, Advances in Fractional Calculus, Vol. 4, Springer, 2007.
[18] K. Oldham, J. Spanier, The Fractional Calculus Theory and Applications of Differentiation and Integration to Arbitrary Order, Elsevier, 1974.
[19] X. Zhang, X. Yu, H. Qin, Optimal operation of multi-reservoir hydropower systems using enhanced comprehensive learning particle swarm optimization, J. Hydro-Environ. Res. 10 (2016) 50–63.
[20] Y. Sun, H. Duan, N. Xian, Fractional-order controllers optimized via heterogeneous comprehensive learning pigeon-inspired optimization for autonomous aerial refueling hose–drogue system, Aerosp. Sci. Technol. 81 (2018) 1–13.
[21] D. Yousri, D. Allam, M. Eteiba, P.N. Suganthan, Static and dynamic photovoltaic models' parameters identification using chaotic heterogeneous comprehensive learning particle swarm optimizer variants, Energy Convers. Manage. 182 (2019) 546–563.
[22] R. Boudjemaa, D. Oliva, F. Ouaar, Fractional Lévy flight bat algorithm for global optimisation, Int. J. Bio-Inspired Comput. 15 (2) (2020) 100–112.
[23] D. Yousri, S. Mirjalili, Fractional-order cuckoo search algorithm for parameter identification of the fractional-order chaotic, chaotic with noise and hyper-chaotic financial systems, Eng. Appl. Artif. Intell. 92 (2020) 103662.
[24] D. Yousri, M. Abd Elaziz, S. Mirjalili, Fractional-order calculus-based flower pollination algorithm with local search for global optimization and image segmentation, Knowl.-Based Syst. (2020) 105889.
[25] Y. Mousavi, A. Alfi, Fractional calculus-based firefly algorithm applied to parameter estimation of chaotic systems, Chaos Solitons Fractals 114 (2018) 202–215.
[26] J. Anem, G.S. Kumar, R. Madhu, Cat swarm fractional calculus optimization-based deep learning for artifact removal from EEG signal, J. Exp. Theor. Artif. Intell. (2019) 1–20.
[27] M. Abd Elaziz, D. Yousri, M. A.A. Al-qaness, A. M. AbdelAty, A. G. Radwan, A. A. Ewees, A Grunwald–Letnikov based Manta ray foraging optimizer for global optimization and image segmentation, Eng. Appl. Artif. Intell. (2020) 104105.
[28] N. Awad, M. Ali, J. Liang, B. Qu, P. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2017 Special Sessionand Competition on Single Objective Real-Parameter Numerical Optimization. Nanyang Technologial University, Jordan University of Science and Technology and Zhengzhou University, Singapore and Zhenzhou, Tech. Rep 201611, Nanyang Technological University, Jordan University of Science and Technology and Zhengzhou University, Singapore and Zhenzhou, China, 2016.
[29] C. Yue, K. Price, P. Suganthan, J. Liang, M. Ali, B. Qu, N. Awad, P. Biswas, Problem Definitions and Evaluation Criteria for the CEC 2020 Special Session and Competition on Single Objective Bound Constrained Numerical Optimization, Tech. Rep 201911, Comput. Intell. Lab., Zhengzhou Univ., Zhengzhou, China, 2019.
[30] W. Zhao, L. Wang, Z. Zhang, Artificial ecosystem-based optimization: a novel nature-inspired meta-heuristic algorithm, Neural Comput. Appl. (2019) 1–43.
[31] S. Li, H. Chen, M. Wang, A.A. Heidari, S. Mirjalili, Slime mould algorithm: A new method for stochastic optimization, Future Gener. Comput. Syst. (2020).
[32] W. Zhao, Z. Zhang, L. Wang, Manta ray foraging optimization: An effective bio-inspired optimizer for engineering applications, Eng. Appl. Artif. Intell. 87 (2020) 103300.
[33] S. Mirjalili, SCA: a sine cosine algorithm for solving optimization problems, Knowl.-Based Syst. 96 (2016) 120–133.
[34] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360), IEEE, 1998, pp. 69–73.
[35] Y. Wang, Z. Cai, Q. Zhang, Differential evolution with composite trial vector generation strategies and control parameters, IEEE Trans. Evol. Comput. 15 (1) (2011) 55–66.
[36] I. Podlubny, Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of their Solution and Some of their Applications, Vol. 198, Elsevier, 1998.
[37] S. Gholizadeh, E. Salajegheh, Optimal design of structures subjected to time history loading by swarm intelligence and an advanced metamodel, Comput. Methods Appl. Mech. Engrg. 198 (37–40) (2009) 2936–2949.
[38] C.A.C. Coello, Use of a self-adaptive penalty approach for engineering optimization problems, Comput. Ind. 41 (2) (2000) 113–127.
[39] S. Mirjalili, S.M. Mirjalili, A. Hatamlou, Multi-verse optimizer: a nature-inspired algorithm for global optimization, Neural Comput. Appl. 27 (2) (2016) 495–513.
[40] A. Kaveh, M. Khayatazad, A new meta-heuristic method: ray optimization, Comput. Struct. 112 (2012) 283–294.
[41] K.S. Lee, Z.W. Geem, A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice, Comput. Methods Appl. Mech. Engrg. 194 (36) (2005) 3902–3933.
[42] R.A. Krohling, L. dos Santos Coelho, Coevolutionary particle swarm optimization using Gaussian distribution for solving constrained optimization problems, IEEE Trans. Syst. Man Cybern. B 36 (6) (2006) 1407–1416.
[43] K. Deb, Optimal design of a welded beam via genetic algorithms, AIAA J. 29 (11) (1991) 2013–2015.
[44] K. Ragsdell, D. Phillips, Optimal design of a class of welded structures using geometric programming, J. Eng. Ind. 98 (3) (1976) 1021–1025.
[45] S. Mirjalili, A. Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67.
[46] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, GSA: a gravitational search algorithm, Inform. Sci. 179 (13) (2009) 2232–2248.
[47] A.A. Ewees, M.A. Elaziz, E.H. Houssein, Improved grasshopper optimization algorithm using opposition-based learning, Expert Syst. Appl. 112 (2018) 156–172.
[48] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: Algorithm and applications, Future Gener. Comput. Syst. 97 (2019) 849–872.
[49] F.-z. Huang, L. Wang, Q. He, An effective co-evolutionary differential evolution for constrained optimization, Appl. Math. Comput. 186 (1) (2007) 340–356.
[50] C.A.C. Coello, E.M. Montes, Constraint-handling in genetic algorithms through the use of dominance-based tournament selection, Adv. Eng. Inform. 16 (3) (2002) 193–203.
[51] E. Mezura-Montes, C.A.C. Coello, An empirical study about the usefulness of evolution strategies to solve constrained optimization problems, Int. J. Gen. Syst. 37 (4) (2008) 443–473.
[52] T. Ray, P. Saini, Engineering design optimization using a swarm with an intelligent information sharing among individuals, Eng. Optim. 33 (6) (2001) 735–748.
[53] A.D. Belegundu, J.S. Arora, A study of mathematical programming methods for structural optimization. Part II: Numerical results, Internat. J. Numer. Methods Engrg. 21 (9) (1985) 1601–1623.
[54] S. Mirjalili, Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm, Knowl.-Based Syst. 89 (2015) 228–249.
[55] B. Kannan, S.N. Kramer, An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design, 1994.
[56] Q. He, L. Wang, A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization, Appl. Math. Comput. 186 (2) (2007) 1407–1422.
[57] H. Liu, Z. Cai, Y. Wang, Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization, Appl. Soft Comput. 10 (2) (2010) 629–640.
[58] A. Kaveh, S. Talatahari, An improved ant colony optimization for constrained engineering design problems, Eng. Comput. 27 (1) (2010) 155–182.
[59] S. Gupta, R. Tiwari, S.B. Nair, Multi-objective design optimisation of rolling bearings using genetic algorithms, Mech. Mach. Theory 42 (10) (2007) 1418–1443.
[60] P. Savsani, V. Savsani, Passing vehicle search (PVS): A novel metaheuristic algorithm, Appl. Math. Model. 40 (5–6) (2016) 3951–3978.
[61] A.A. Ewees, M.A. Elaziz, Performance analysis of chaotic multi-verse harris hawks optimization: A case study on solving engineering problems, Eng. Appl. Artif. Intell. 88 (2020) 103370.
[62] R.V. Rao, V.J. Savsani, D. Vakharia, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, Comput. Aided Des. 43 (3) (2011) 303–315.
[63] D. Dua, C. Graff, UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, 2017, URL http://archive.ics.uci.edu/ml.

[64] F.A. Hashim, E.H. Houssein, M.S. Mabrouk, W. Al-Atabany, S. Mirjalili, Henry gas solubility optimization: A novel physics-based algorithm, Future Gener. Comput. Syst. 101 (2019) 646–667.
[65] R.A. Ibrahim, M. Abd Elaziz, S. Lu, Chaotic opposition-based grey-wolf optimization algorithm based on differential evolution and disruption operator for global optimization, Expert Syst. Appl. 108 (2018) 1–27.
[66] S. Oreski, G. Oreski, Genetic algorithm-based heuristic for feature selection in credit risk assessment, Expert Syst. Appl. 41 (4) (2014) 2052–2064.
[67] R.A. Ibrahim, A.A. Ewees, D. Oliva, M. Abd Elaziz, S. Lu, Improved salp swarm algorithm based on particle swarm optimization for feature selection, J. Ambient Intell. Humaniz. Comput. 10 (8) (2019) 3155–3169.