Applied Soft Computing 59 (2017) 596–621

Contents lists available at ScienceDirect

Applied Soft Computing

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

A novel physical based meta-heuristic optimization method known as Lightning Attachment Procedure Optimization

A. Foroughi Nematollahi[a], A. Rahiminejad[b], B. Vahidi[a,*]

[a] Electrical Engineering Department, Amirkabir University of Technology, Tehran, 1591634311, Iran
[b] Department of Electrical and Computer Science, Esfarayen University of Technology, Esfarayen, North Khorasan, 9661998195, Iran

ARTICLE INFO

Article history:
Received 12 December 2016
Received in revised form 22 May 2017
Accepted 16 June 2017
Available online 21 June 2017

Keywords:
Meta-heuristic
Physical based
Optimization algorithm
Constrained optimization
Optimization

ABSTRACT

In this article, A novel nature-inspired optimization algorithm known as Lightning Attachment Procedure Optimization (LAPO) is proposed. The proposed approach mimics the lightning attachment procedure including the downward leader movement, the upward leader propagation, the unpredictable trajectory of lightning downward leader, and the branch fading feature of lightning. Final optimum result would be the lightning striking point. The proposed method is free from any parameter tuning and it is rarely stuck in the local optimum points. To evaluate the proposed algorithm, 29 mathematical benchmark functions are employed and the results are compared to those of 9 high quality well-known optimization methods The results of the proposed method are compared from different points of views, including quality of the results, convergence behavior, robustness, and CPU time consumption. Superiority and high quality performance of the proposed method are demonstrated through comparing the results. Moreover, the proposed method is also tested by five classical engineering design problems including tension/compression spring, welded beam, pressure vessel designs, Gear train design, and Cantilever beam design and a high constraint optimization problem known as Optimal Power Flow (OPF) which is a high constraint electrical engineering problem. The excellence performance of the proposed method in solving the problems with large number of constraints and also discrete optimization problems are also concluded from the results of the six engineering problem.

1. Introduction

Optimization refers to determination of the decision variables of a function so that the function would be in its minimum or maximum value. A majority of problems especially engineering ones are optimization problems in which the decision variables should be determined in a way that the systems operate in their best operation points. Since most engineering problems are non-linear, non-convex, and complicated, sophisticated methods are required to solve these problems optimally. This has resulted in excellence research in the area of optimization methods in recent years [1–3].

The early methods used for solving the optimization problems were the mathematical or numerical methods in which the decision variables are obtained by achieving the point at which the derivative is zero. However, solving the non-linear non-convex problems with lots of variables and constraints using these methods is almost impossible. Moreover, since there may be lots of local optimum points (at which the derivative is also zero), the numerical methods may be stuck at these local optimum points. In other words, there is no guarantee to find the best overall point of the problem using the numerical methods. Other specifications of these methods could be stated as need for initial guess, additional mathematical calculations, divergent issue, complicated implementation, difficult convergence for discrete types of optimization problem, etc. [4–6].

To overcome the drawbacks of numerical methods including derivative, complexity, and being trapped in local optimum points, a bunch of optimization methods known as meta-heuristics methods are introduced and developed in recent decades [7]. These methods, in which random operators are employed, are inspired by simple concepts. On one hand, using the random operators reduces the probability of being stuck in local optimum point. On the other hand, it increases diversity of the final results. Nevertheless, since the meta-heuristics methods 1-are simple methods which could be applied for both continuous and discrete functions, 2-do not need any additional complex mathematical operations such as deriva-

* Corresponding author at: Amirkabir University of Technology, 424 Hafez Ave., Tehran, 15916343111, Iran.
E-mail addresses: amin.forooghi@aut.ac.ir (A.F. Nematollahi), arahiminezhad@aut.ac.ir (A. Rahiminejad), bvahidi@yahoo.com, vahidi@aut.ac.ir (B. Vahidi).

http://dx.doi.org/10.1016/j.asoc.2017.06.033
1568-4946/

tive, and 3-rarely get stuck in local optimum points, they are used widely in different optimization problems.

for initial guess, additional mathematical calculations, divergent issue, complicated implementation, difficult convergence for discrete types of optimization problem, etc. [4–6].

To overcome the drawbacks of numerical methods including derivative, complexity, and trapping in local optimum points, a bunch of optimization methods known as meta-heuristics methods are introduced and developed in the recent decades [7]. These methods, in which random operators are employed, are inspired by simple concepts. In one side, using the random operators reduces the probability of being stuck in local optimum point. On the other side, it increases the diversity of final results. Nevertheless, since the meta-heuristics methods 1-are simple methods which could be applied for both continuous and discrete functions, 2-do not need any additional complex mathematical operations such as derivative, and 3-rarely get stuck in local optimum points, they are used widely in different optimization problems.

The meta-heuristic methods could be classified in two main categories

1. Single solution based methods in which a random solution is generated and improved until the best overall answer is obtained. Simulated Annealing (SA) [8] is one the most well-known single solution based methods.
2. Population based methods in which a set of solutions is generated randomly in the predefined search space. All the solutions are updated iteratively until the best answer is obtained among the population.

In the second group, each solution tries to find the local optimum point around itself based on the information interaction with the other solutions. Moreover, the solutions help others not to get stuck in the local optimum points. Since in these kinds of methods the whole search area is almost investigated, better results are achieved compared to the single solution based methods. Hence, they have attracted attentions of researchers recently.

The population based optimization methods are mostly inspired by the nature and could be categorized in 4 different categories.

1. Evolutionary algorithms: these methods simulate the evolution of the nature. The first generation is produced randomly and evolved gradually. The best answer would be the best solution among the whole population in the last iteration of evolution. Genetic Algorithm [9] is the first and most well-known meta-heuristic method which simulates the Darwin's theory of evolution. Evolution Strategy (ES) [10], Genetic Programming (GP) [11], and Biogeography-Based Optimizer (BBO) [12] are some other evolutionary methods.
2. Physical based optimization algorithm: in this kind of methods, the physical rules are employed for updating the solutions in each iteration. Charged System Search (CSS) [13], Central Force Optimization (CFO) [14], Artificial Chemical Reaction Optimization Algorithm (ACROA) [15], Black Hole (BH) algorithm [16], Ray Optimization (RO) algorithm [17], Small-World Optimization Algorithm (SWOA) [18], Galaxy-based Search Algorithm (GbSA) [19], Water Evaporation Optimization Algorithm [20], Multi-Verse Optimizer(MVO) [21] and Gravitational Search Algorithm(GSA) [22] are classified as the physical methods.
3. The third group is based on movement of a group of animals. This kind of methods mimic the social behavior of animals in order to enhance their knowledge of a goal such as food source. The most well-known method of this group is Particle Swarm Optimization (PSO) [23,24]. Wolf pack search algorithm [25], Cuckoo Search (CS) [26], Firefly Algorithm (FA) [27], Bird Mating Optimizer (BMO)[28], Monkey Search Algorithm (MSA) [29], Coral Reef Optimization Algorithm(CRO) [30,31], Artificial Bee Colony (ABC) algorithm[32], Antlion Optimization Algorithm(ALO) [33], Grey Wolf Optimization algorithm (GWO) [7], Moth-flame Optimization (MFO) algorithm [34], Whale Optimization Algorithm (WOA) [35], Dragonfly algorithm [36], Dolphin Echolocation (DE) [37], and Krill Herd (KH) [38] are some of other methods of this group.
4. The fourth group are the methods which simulate the human behavior. Teaching-Learning-Base Optimization (TLBO) [39] algorithm is one of the most well-known methods of this group which simulates the enhancing procedure of a class grade. The other methods which could be classified in this group are Harmony Search (HS) [40], Tabu Search [41–44], Group Search Optimizer (GSO) [45], Imperialist Competitive Algorithm (ICA) [46], League Championship Algorithm (LCA)[47,48], Firework Algorithm [49–51], Colliding Bodies Optimization (CBO) [52–54], Tug of War Optimization (TWO) [55], and Interior Search Algorithm (ISA) [56].

The optimization algorithms should be capable of searching all the search space. This refers to exploration. The high exploration methods are those with a large diversity of solutions in an iteration [57]. For instance, In GA [57], high probability of crossover causes more combination of individuals and it is the main mechanism for the exploration milestone. Moreover, the methods should enhance quality of the solutions in each iteration. This refers to exploitation which is the characteristics of a method in finding the local optimum point around a solution. The mechanism that brings GA exploitation is the mutation operators. Mutation causes slight random changes in the individuals and local search around the candidate solutions. It is obvious that these two characteristics are in contradiction to each other. In other words, high exploration means a large diversity of solutions which results in not obtaining the exact best global answer [58]. On the contrary, high exploitation may result in being trapped in local optimum point. Thus, a tradeoff should be established between these two features.

Although a large number of optimization algorithms are introduced in the literature, based on No-Free-Lunch (NFL) theorem [59], it cannot be claimed that an optimization algorithm could solve all the problems. In other words, a method may have acceptable results for some problems, but not for some others. Thus, new methods are introduced in order to solve a wider range of problems. This is the motivation of this study in which a new optimization method inspired by the attachment procedure of lightning is introduced. It should be mentioned that an optimization algorithm known as Lightning Search Algorithm (LSA) which is also inspired by lightning phenomena was proposed by Shareef et al., [60]. The inspiration of both LSA and proposed LAPO are the same; however, the view and the equations by which the solutions are updated, are completely different. In LSA the solutions are the atoms traveling through the atmosphere by their kinetic energy and ionizing the nearby space by collision with other molecules and atoms. In this method, only the downward movement of the ionized channel is considered. In the proposed LAPO, the solutions are the jumping points of the lightning. In this model both the upward and downward leader are considered. A full description of the proposed method is explained in Section 2. In order to distinguish the two methods, the results of the proposed LAPO for some benchmark functions are compared to those of LSA.

2. Lightning Attachment Procedure Optimization (LAPO)

In this section, the proposed algorithm and its inspiration are illustrated.

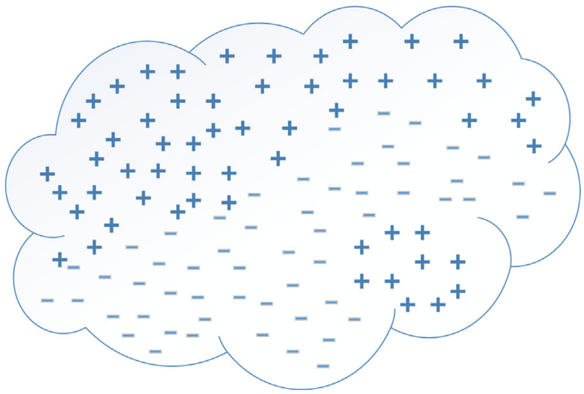

Fig. 1. The charge distribution in a cloud.

Fig. 2. Different starting points of lightning from the cloud.

2.1. Inspiration

The method proposed in this paper is inspired by the nature of lightning attachment process. This process consists of 4 important phases; 1- air breakdown on cloud surface, 2- downward movement of lightning channel, 3- upward leader inception and propagation from the ground (or earthed objects), and 4- final jump. Herein, these stages are explained.

2.1.1. Air breakdown on cloud edge

The charge of cloud could be considered in three parts as depicted in Fig. 1 [61]. a huge value of negative charge which is placed in the lower part of cloud, a huge value of positive charge located in the upper part of cloud, and a small positive charge in the lower part of cloud. As the quantities of these charges increase, the potential between the charge centers increase and the breakdown may occur between the negative charges and huge positive charge part or small positive charge part. Following this breakdown, the voltage gradient at the edge of the cloud increases and the lightning is formed and a huge value of electrical charge (mostly negative charge) moves toward the ground. As the high speed photographs of actual lightning reveal, starting point of lightning could be more than one point (see Fig. 2).

2.1.2. Downward leader movement toward the ground

As the air breakdown occurs at the edge of the cloud, the lightning approaches the ground in a stepwise movement. After each step, the lightning stops, then moves to one or more other directions towards the ground. In order to understand this procedure, after each step, a hemisphere is considered beneath the leader tip with the center of leader tip and the radius of next step length (see Fig. 3) [62–64]. There are lots of potential points on the surface of this hemisphere which could be selected as the next jump point. The next jump point is selected randomly; however, a point with higher value of electrical field between the line connecting the leader tip and the corresponding point is more probable to be considered as the next jump[65].

2.1.3. Branch fading

In the case where there are more than one points for the next jump of lightning, the charge of upper branch is divided into new branches. The same procedure is repeated for all of the new branches and new branches are formed. Whenever, the charge of a branch becomes lower than a critical value (1 μC), no air breakdown occurs and as a result no further movement occurs. Thus, this branch would disappear as shown in Fig. 4.

2.1.4. Upward leader propagation

Presence of cloud means presence of a huge negative charge above the ground. This results in aggregation of positive charges on the ground surface or earthed object beneath the cloud. In the sharp points, the high electric field causes air breakdown; thus, upward leader starts from these sharp points and propagates through the air (see Fig. 5). As the downward leader approaches the ground, these upward leaders go toward the downward leader faster. The branching procedure and branch fading also occur for the upward leaders.

2.1.5. Final jump (striking point determination)

Whenever an upward leader reaches to a downward leader, the final jump occurs and the striking point would be the point from which the upward leader has started. In this situation, all the other branches disappear and charge of the cloud is naturalized through this channel.

2.2. Mathematical model of proposed algorithm

In this part the proposed model inspired by the nature of lightning phenomena is illustrated.

2.2.1. Initialization (test points)

For all the population based algorithms, an initial population is required. In other words, a number of decision variable set is defined randomly in the predefined range. These solutions are considered as the test points located at the cloud and the ground surface. In fact, some of these tests points are the emanating point of lightning, and some of them are the points from which the upward leaders start. For all the test points, the objective function is calculated and considered as the electric field of these test points. The test points are defined as follows (Eq. (1)):

$$X^{i}_{testpoint} = X^{i}_{min} + (X^{i}_{max} - X^{i}_{min}) \times rand \quad (1)$$

Where X_{min} and X_{max} are the lower and upper bound of decision variables, and *rand* is a random variable selected uniformly in the range of [0,1]. The electric field of a test point which is the fitness of the solution is calculated based on the objective function (Eq. (2)).

$$F^{i}_{testpoint} = obj(X^{i}_{testpoint}) \quad (2)$$

2.2.2. Next jump determination

The averages of all test points are obtained and the values of fitness of these points are calculated as follows (Eqs. (3) and (4)):

$$X_{ave} = mean(X_{testpoint}) \quad (3)$$

$$F_{ave} = obj(X_{ave}) \quad (4)$$

There are lots of potential points for a test point, which the lightning could go through. For mathematical simulation of the method, for a specific test point, all the other test points are considered

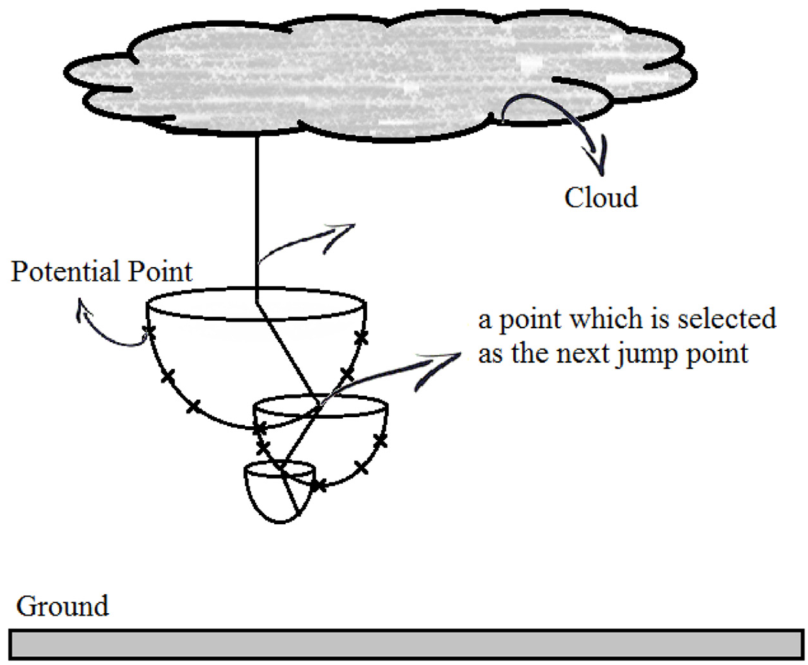

Fig. 3. The procedure of stepwise movement of downward lightning leader.

Fig. 4. Fading branches in high speed camera photographs of lightning.

as the potential next jump points. Since the lightning has a random behavior, for test point *i*, a random point *j* is selected among the population ($i \neq j$). If the electric field of the point *j* is higher than that of the average value (the fitness of point *j* is better than that of the average point), the lightning jumps towards this point, otherwise, the lightning moves to another direction. This step is presented in Fig. 6, and the mathematical description is shown in Eqs. (5) and (6).

If the electric field of potential point *j* is higher than that of average electric field

Fit, lower

$$X^{i}_{testpoint\_new} = X^{i}_{testpoint} + rand \times (Xave + rand \times (X^{j}_{potential\ point})) \quad (5)$$

If the electric field of potential point *j* is lower than that of average electric field

$$X^{i}_{testpoint\_new} = X^{i}_{testpoint} - rand \times (Xave + rand \times (X^{j}_{potential\ point})) \quad (6)$$

2.2.3. Branch fading

If the electric field of new test point is higher than a critical value (if the fitness function is better than the previous point), the branch sustains; otherwise, it fades. The mathematical formulation of this feature is as Eq. (7)

$$\begin{aligned} X^{i}_{tetpoint} &= X^{i}_{testpoint\_new} && \text{if } F^{i}_{testpoint\_new} < F^{i}_{tetpoint} \\ X^{i}_{testpoint\_new} &= X^{i}_{tetpoint} && \text{otherwise} \end{aligned} \quad (7)$$

This process is performed for all the test points. In other words, in the first phase, all the rest points are considered to move down.

2.2.4. Upward leader movement

As mentioned in the previous steps, all the test points are considered as the downward leader and moved down. In this phase, all the test points are considered as the upward leader and moved up. The upward leader movement is based on the charge of down-

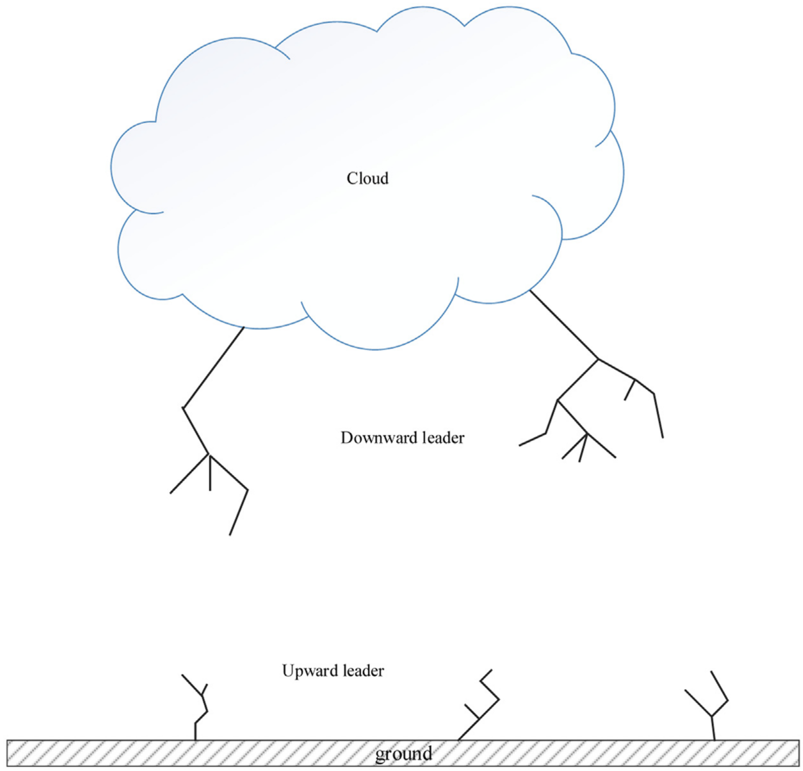

Fig. 5. The upward leader formation and propagation through the downward leader.

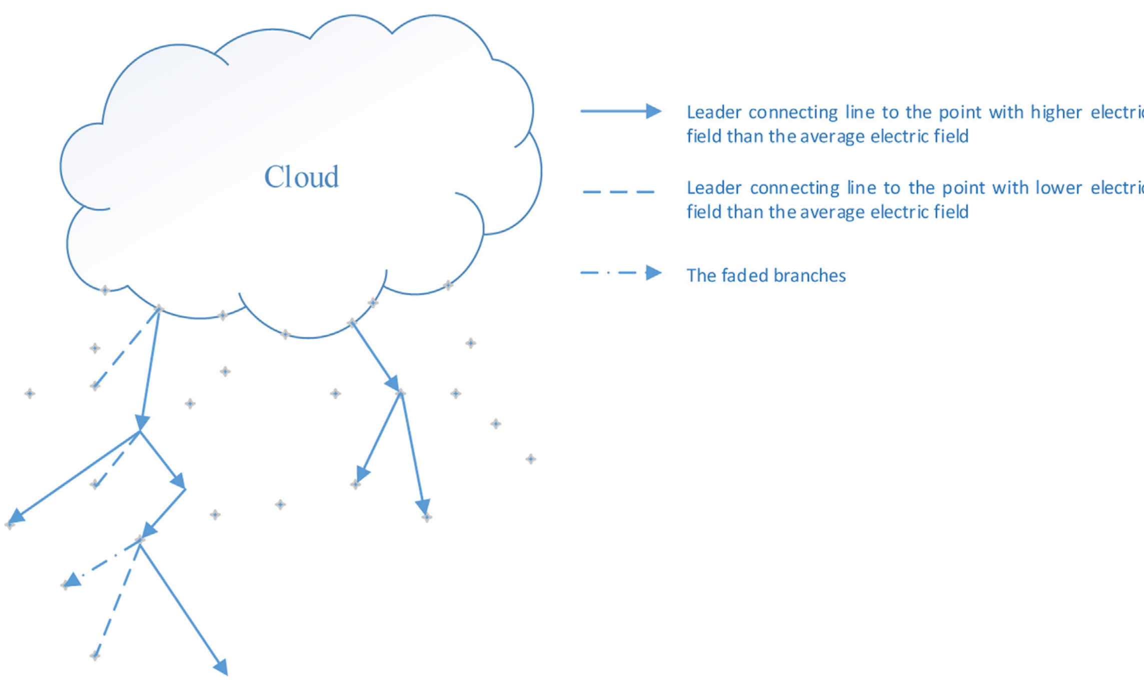

Fig. 6. The next jump determination and lightning trajectory identification.

ward leader which is basically distributed exponentially along the channel. Thus, an exponent factor is defined as follow (Eq. (8)):

$$S = 1\text{-}\left(\frac{t}{t_{max}}\right) \times \exp\left(-\frac{t}{t_{max}}\right) \tag{8}$$

Where t is the number of iteration, and t_{max} is the maximum number of iterations. The next jump of the upward leader where t is the number of iterations, and t_{max} is the maximum number of iterations. The next jump of the upward leader is related to the charge of the channel which is a function of leader tip height, cloud height, and the charge of channel [66]. Thus, the next trajectory of a test point as an upward leader is mathematically formulated as follows (Eq. (9)):

$$X_{testpoint\_new} = X_{testpoint\_new} + rand \times S \times (X_{min}\text{-}X_{max}) \tag{9}$$

where X_{min} and X_{max} are the best and the worst solutions of the population which refer to the leader tip height (best answer) and the height of the cloud (the worst answer).

2.2.5. Final jump

The lightning procedure stops when the upward leader and the downward leader meet each other and the striking point is determined. Here, the optimization algorithm is finished whenever the convergence criterion (for instance finishing the maximum iteration) is satisfied.

2.2.6. Enhancement the performance

In order to enhance performance of the proposed method, in each iteration, the average of the whole population is calculated and the fitness of average solution is obtained. If the fitness of the worst solution is worse than the average solution, it is replaced by the average solution. The pseudo code of the propose LAPO algorithm is as follow:

```
Initialize the first population of test points randomly in the specific range
while the end criterion is not achieved
  Calculate the fitness of test points
  Set the test point with the worst fitness as TestPoint_w
  Obtain TestPoint_ave which is the mean value of all the test points
  if fitness of TestPoint_ave is better than the fitness of TestPoint_w
    TestPoint_w=TestPoint_ave
  end
  for i=1:Npop (each test point)
    select TestPoint_j randomly which is not equal to TestPoint_i
    for e = 1:Nv (number of variables)
      Update the variables of TestPoint_i based on Eqs. (5), and (6), as TestPoint_i,new
      Check the boundaries
    end
    Calculate the fitness of TestPointi,new
    if the fitness of TestPoint_i,new is better than TestPoint_i
      TestPoint_i=TestPoint_i,new
    end
  end
  for i=1:Npop (each test point)
    for e = 1:Nv (number of variables
      Update the variables of TestPoint_i based on Eq. (9), as TestPoint_i,new
      Check the boundaries
    end
    Calculate the fitness of TestPoint_i,new
    if the fitness of TestPoint_i,new is better than TestPoint_i
      TestPoint_i=TestPoint_i,new
    end
  end
  TestPoint_best=the TestPoint with the best fitness
end
  return TestPoint_best
```

Flowchart of the proposed LAPO method is illustrated in Fig. 7.

Table 1
Unimodal benchmark functions.

Function	Dim	Range	fmin
$F_1(x) = \sum_{i=1}^{n} x^2$	30,200	[−100,100]	0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30,200	[−10,10]	0
$F_3(x) = \sum_{i=1}^{n} (\sum_{j=1}^{i} x_j)^2$	30,200	[−100,100]	0
$F_4(x) = \mathrm{Max}\{\lvert x_i \rvert, 1 \le i \le 5\}$	30,200	[−100,100]	0
$F_5(x) = \sum_{i=1}^{n-1} \{100(x_{i+1} - x_i)^2 + (x_i - 1)^2\}$	30,200	[−30,30]	0
$F_6(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	30,200	[−100,100]	0
$F_7(x) = \sum_{i=1}^{n} (ix_i^4 + \mathrm{random}[0,1])$	30,200	[−1.28,1.28]	0

3. Evaluation and discussion

In this section, the proposed method is evaluated by means of 29 different benchmark mathematical functions and the results are compared to those of some other *meta*-heuristic methods. The test functions include four different groups of functions: 1- unimodal, 2- multimodal, 3- fixed-dimension multimodal, and 4- composite functions. These four kinds of test functions are listed in Tables 1–4, respectively. In these tables, *Dim* refers to dimension of the function (i.e., number of decision variables), *Range* denotes the search area (i.e., the limitation of variables), and f_{min} is the optimum value of the test function. It should be mentioned that in order to evaluate the method for solving large scale test functions, unimodal and multimodal test functions with 200 variables are employed. The 2-D version of the test functions are depicted in Figs. 8–11. It should be mentioned that the 2-D version of test functions F20, F22, and F23 are similar to F21. The foregoing test functions are commonly used by researchers for evaluating the optimization methods. In other to test the proposed method in solving more realistic problems, 5 classic mechanical optimization problems and a high constraint electrical optimization problem known as OPF are also employed for evaluating the method. Using these optimization problems, capability of the proposed method in solving the high constraints and discrete problems is also evaluated.

The meta-heuristic methods with which the proposed method is compared include 1- Artificial Bee Colony (ABC) [32], 2- Differen-

Table 2
Multimodal benchmark functions.

Function	Dim	Range	fmin
$F_8(x)=\sum_{i=1}^{n}-x_i\sin(\sqrt{\lvert x_i\rvert})$	30,200	[−500,500]	-418.9829*Dim
$F_9(x)=\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)+10]$	30,200	[−5.12, 5.12]	0
$F_{10}(x)=-20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2})-\exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i))+20+e$	30,200	[−32,32]	0
$F_{11}(x)=\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}})+1$	30,200	[−600,600]	0
$F_{12}(x)=\frac{\pi}{n}\{10\sin(\pi y_1)+\sum_{i=1}^{n-1}(y_i-1)^2[1+10\sin^2(\pi y_{i+1})]+(y_n-1)^2\}+\sum_{i=1}^{n}u(x_i,10,100,4)$ $y_i=1+\frac{x_i+1}{4}$ $u(x_i,a,k,m)=\begin{cases}k(x_i-a)^m & x_i>a\\ 0 & -a<x_i<a\\ k(-x_i-a)^m & x_i<-a\end{cases}$	30,200	[−50,50]	0
$F_{13}(x)=0.1\{10\sin^2(3\pi x_1)+\sum_{i=1}^{n}(x_i-1)^2[1+10\sin^2(3\pi x_i+1)]+$ $(x_n-1)^2[1+\sin^2(2\pi x_n)]\}+\sum_{i=1}^{n}u(x_i,5,100,4)$	30,200	[−50,50]	0

Table 3
Fixed-dimension multimodal benchmark functions.

Function	Dim	Range	fmin
$F_{14}(x)=\left(\frac{1}{500}+\sum_{j=1}^{25}\frac{1}{\sum_{i=1}^{2}(x_i-a_{ij})^6}\right)^{-1}$	2	[−65,65]	1
$F_{14}(x)=\left(\frac{1}{500}+\sum_{j=1}^{25}\frac{1}{\sum_{i=1}^{2}(x_i-a_{ij})^6}\right)^{-1}$	4	[−5,5]	0.00030
$F_{16}(x)=4x_x^2-2.1x_1^4+\frac{1}{3}x_1^6+x_1x_2-4x_2^2+4x_2^4$	2	[−5,5]	−1.0316
$F_{17}(x)=(x_2-\frac{5.1}{4\pi^2}x_1^2+\frac{5}{\pi}x_1-6)^2+10(1-\frac{1}{8\pi})\cos x_1+10$	2	[−5,5]	0.398
$F_{18}(x)=\left[1+(x_1+x_2+1)^2\left(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2\right)\right]\times$ $\left[30+(2x_1-3x_2)^2\times\left(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2\right)\right]$	2	[−2,2]	3
$F_{19}(x)=-\sum_{i=1}^{4}c_i\exp(-\sum_{j=1}^{3}a_{ij}(x_j-p_{ij})^2)$	3	[1,3]	−3.86
$F_{20}(x)=-\sum_{i=1}^{4}c_i\exp(-\sum_{j=1}^{6}a_{ij}(x_j-p_{ij})^2)$	6	[0,1]	−3.32
$F_{21}(x)=-\sum_{i=1}^{5}\left[(x-a_i)(x-a_i)^T+c_i\right]^{-1}$	4	[0,10]	10.1531
$F_{22}(x)=-\sum_{i=1}^{7}\left[(x-a_i)(x-a_i)^T+c_i\right]^{-1}$	4	[0,10]	−10.4028
$F_{23}(x)=-\sum_{i=1}^{10}\left[(x-a_i)(x-a_i)^T+c_i\right]^{-1}$	4	[0,10]	−10.5363

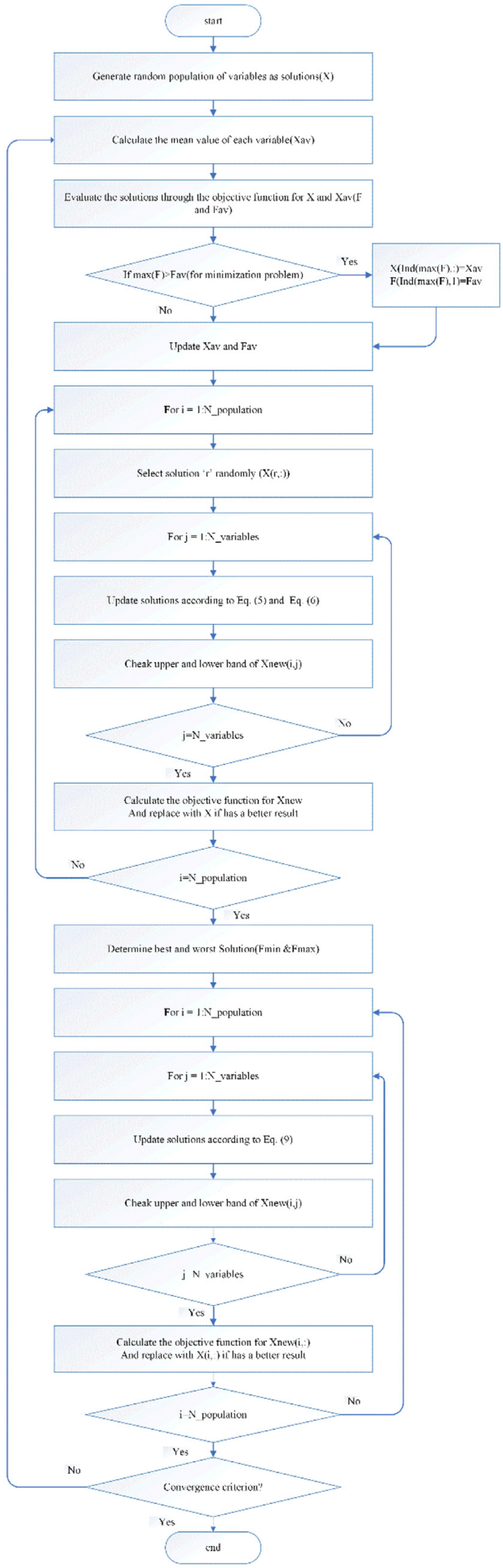

Fig. 7. The flowchart of the proposed LAPO method.

tial Evolution (DE) [67], 3- Shuffled Frog Leaping Algorithm (SFLA) [68], 4- Imperialist Competitive Algorithm (ICA) [46], 5- Particle Swarm Optimization (PSO) [23], 6- Ant-Lion Optimizer (ALO) [33], 7- Gray Wolf Optimizer (GWO) [7], 8- Cuckoo Search Algorithm (CSA) [26], 9- Firefly Optimization Method (FOM) [27], and 10- Lightning Search Algorithm (LSA) [60].

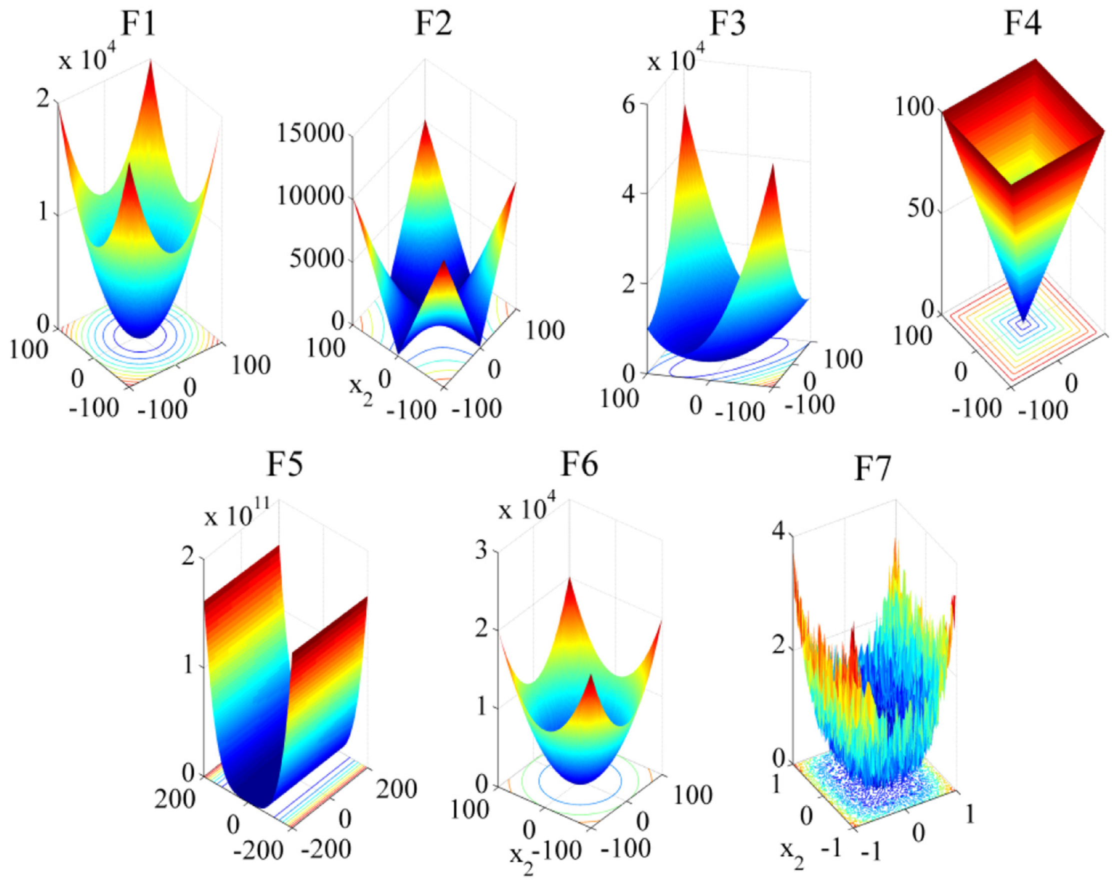

Fig. 8. 2-D version of unimodal test functions.

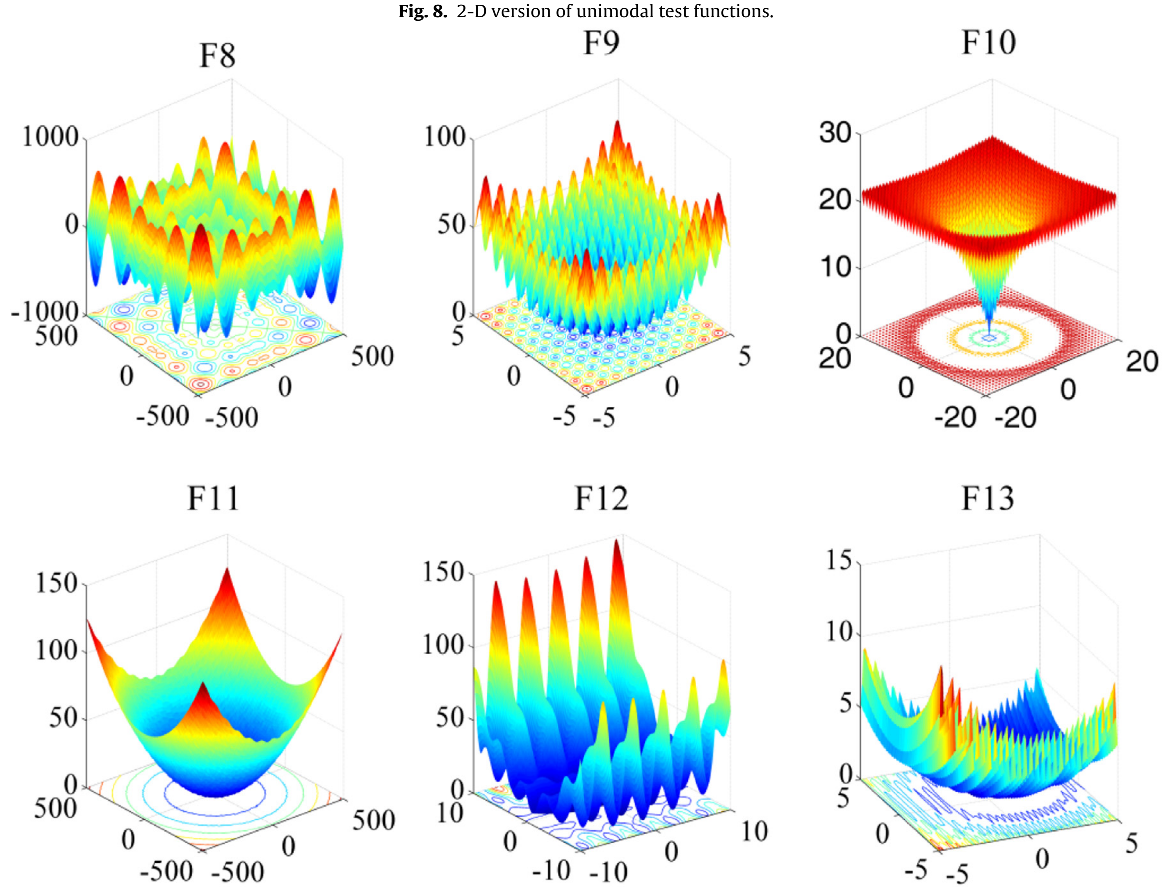

Fig. 9. 2-D version of multimodal test functions.

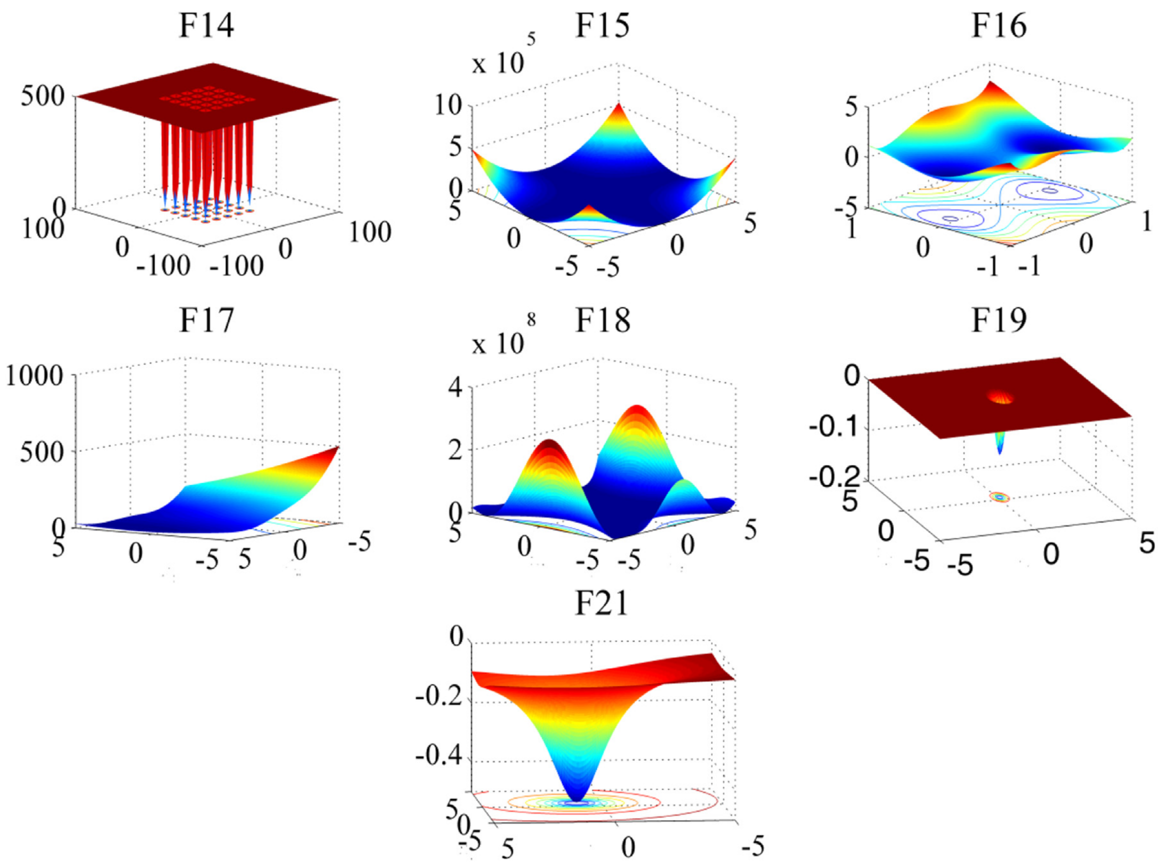

Fig. 10. 2-D version of fixed-dimension multimodal test functions.

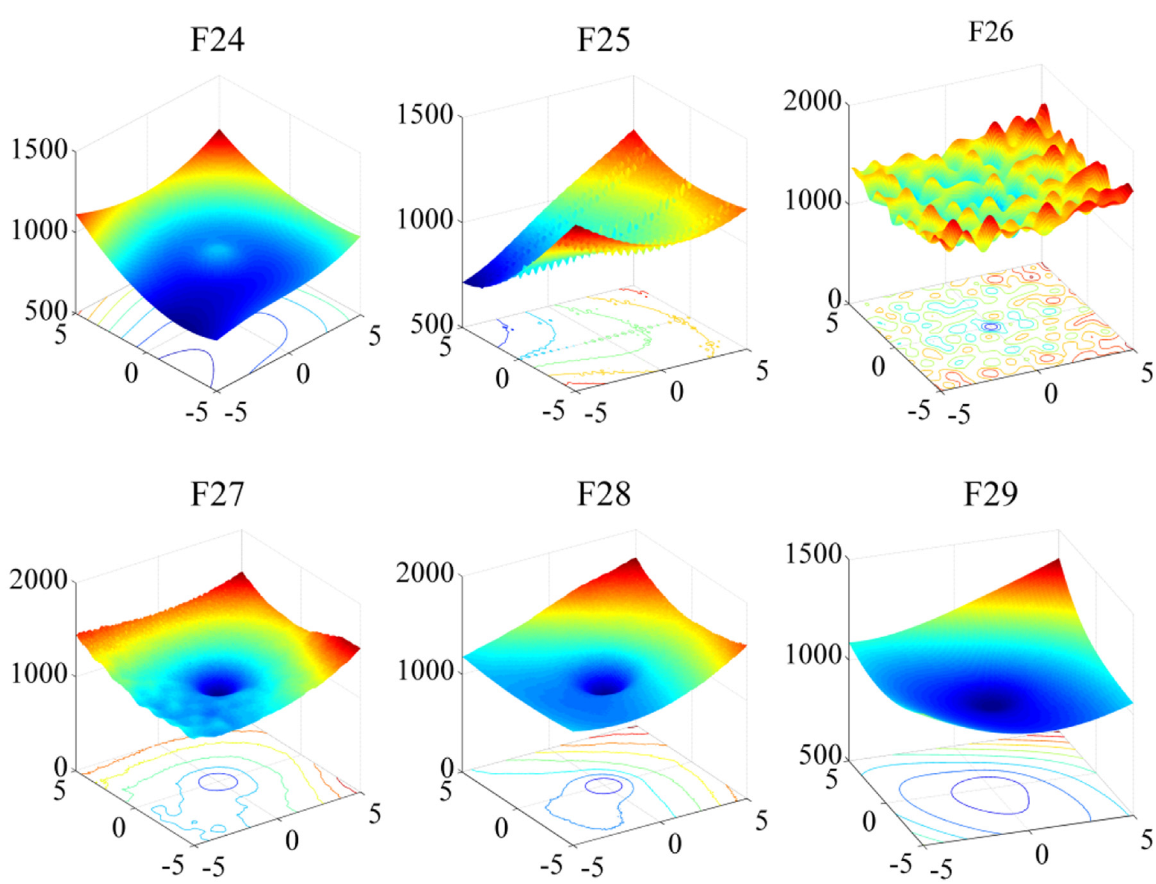

Fig. 11. 2-D version of composite test functions.

Table 4
Composite benchmark functions.

Function	Dim	Range	fmin
F_{24}(CF1) : $f_1, f_2, f_3, \ldots f_{10}$ = Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1,1,1,\ldots,1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [\frac{5}{100}, \frac{5}{100}, \frac{5}{100}, \ldots, \frac{5}{100}]$	10	[−5,5]	0
F_{25}(CF1) : $f_1, f_2, f_3, \ldots f_{10}$ = Griewank's Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1,1,1,\ldots,1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [\frac{5}{100}, \frac{5}{100}, \frac{5}{100}, \ldots, \frac{5}{100}]$	10	[−5,5]	0
F_{26}(CF1) : $f_1, f_2, f_3, \ldots f_{10}$ = Griewank's Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1,1,1,\ldots,1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [1,1,1,\ldots,1]$	10	[−5,5]	0
F_{27}(CF4) : f_1, f_2 = Ackley's Function f_3, f_4 = Rastrigin's Function f_5, f_6 = Weierstras's Function f_7, f_8 = Griewank's Function f_9, f_{10} = Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1,1,1,\ldots,1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [\frac{5}{32}, \frac{5}{32}, 1, 1, \frac{5}{0.5}, \frac{5}{0.5}, \frac{5}{100}, \frac{5}{100}, \frac{5}{100}, \frac{5}{100}]$	10	[−5,5]	0
F_{28}(CF5) : f_1, f_2 = Ackley's Function f_3, f_4 = Rastrigin's Function f_5, f_6 = Weierstras's Functions f_7, f_8 = Griewank's Function f_9, f_{10} = Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [1,1,1,\ldots,1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [\frac{1}{5}, \frac{1}{5}, \frac{5}{0.5}, \frac{5}{0.5}, \frac{5}{100}, \frac{5}{100}, \frac{5}{32}, \frac{5}{32}, \frac{5}{100}, \frac{5}{100}]$	10	[−5,5]	0
F_{29}(CF5) : f_1, f_2 = Rastrigin's Function f_3, f_4 = Weierstras's Function f_5, f_6 = Griewank's Function f_7, f_8 = Ackley's Function f_9, f_{10} = Sphere Function $[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{10}] = [0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1]$ $[\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_{10}] = [0.1 \times \frac{1}{5}, 0.2 \times \frac{1}{5}, 0.3 \times \frac{5}{0.5}, 0.4 \times \frac{5}{0.5}, 0.5 \times \frac{5}{100}, 0.6 \times \frac{5}{100}, 0.7 \times \frac{5}{32}, 0.8 \times \frac{5}{32}, 0.9 \times \frac{5}{100}, 1 \times \frac{5}{100}]$	10	[−5,5]	0

The number of population for each method is selected so that the overall objective function evaluation in the entire iterations would be the same for all the methods. For instance, the objective function is evaluated two times in each iteration in the proposed LAPO method; while, once in each iteration for PSO. Thus, the population of LAPO is half of PSO population. The parameter descriptions of different methods are as follows:

algorithm	Function evaluation in each iteration	Populations size	Other parameter
ABC [32];	2	40	*Alimit*: 15
DE [65]	1	80	F = 0.5, CR = 0.5
SFLA	1	80	–
ICA	1	80	–
PSO	1	80	C1 = C2 = 2, $w = w_{max} - t \times (w_{max} - w_{min})/(t_{max})$, $w_{max} = 0.9$, $w_{min} = 0.4$, t_{max}: maximum iteration number [69]
ALO	2	40	stochastic function limit = 0.5, ratio I based on the current iteration and maximum number of iteration (based on the MATLAB code available in [70]),
GWO	1	80	–
CSA [26]	2	40	discovery rate (Pa): 0.25
Firefly [27],	1	80	α: 0.25, β: 0.2, γ: 1
LSA	2	40	maximum channel time: 10, energy equation based on MATLAB code available in[71],

The maximum cycle number of all the methods is 500 for low dimensional and 2000 for high dimensional functions. Since the meta-heuristic methods are based on random movement of their particles, statistical analysis is needed. Hence, each method is applied on the test functions in 30 different trials. The best overall answer, mean value of the answers, and standard deviation are extracted and compared to each other. It is obvious that a better best answer, a closer mean value to the best answer, and a lower value of standard deviation characterize a better method. Moreover, the methods are compared to each other in terms of computational CPU time consuming. All the algorithms are implemented in MATLAB 2012a, and simulations are performed by a Core i5 PC with 3 GHz processing frequency of CPU and 8GB of RAM. A better method is the faster one. In addition, the methods are compared in terms of convergence. The better is the method it achieves the best overall answer in a lower iteration.

3.1. Exploitation analysis

As mentioned in previous studies such as [7], and [33], unimodal test functions are suitable for evaluating the exploitation characteristic of a method. Since these test functions are simple functions with convex shape, and the meta-heuristic methods are population based methods with lots of particles, finding the best answer of such test functions needs an excellence local search around the optimum point. Results of different methods for solving the unimodal test functions are listed in Table 5. As can be seen from the table, finding the best answers of these test functions are not easy and most of the methods could not obtain the best results in most of their trials. However, as it can be seen, the proposed LAPO method finds acceptable answers for most of the test functions. For

Table 5
results of different method for solving the unimodal test functions (DIM = 30).

F	LAPO				LSA			
	best	Ave	Std	time(s)	best	ave	std	Time(s)
F1	1.3406E-15	2.0664E-13	5.5098E-13	0.8137	2.6192e-11	2.5961e-08	1.3958e-07	13.468
F2	4.2412E-09	2.2547E-08	1.7473E-08	0.8797	0.0015378	0.068542	0.09996	13.819
F3	4.1270E-07	1.1385E-05	3.6624E-05	2.656843	30.719	130.89	168.24	17.374
F4	2.7951E-07	4.3915E-07	1.3825E-07	1.007296	2.8447	4.5491	2.03	13.435
F5	19.5667	22.7427	0.6846	1.061064	21.7546	61.376	34.334	13.94
F6	1.3619E-06	1.1151E-05	1.0200E-05	1.035845	2.00	4.5	2.5495	3.1517
F7	1.3323E-04	7.1418E-04	4.3695E-04	1.2725	0.028583	0.032825	0.0035634	13.9

F	DE				SFLA			
	best	ave	Std	time(s)	best	ave	std	Time(s)
F1	2.6201E-12	1.8812E-04	7.0503E-04	0.5086	4.7749	18.1945	12.1923	1.472477
F2	1.6397E-07	6.1642E-04	0.0021	0.539950	1.1161	3.3016	1.0643	1.559033
F3	137.5353	757.1439	965.7426	1.44286	181.3897	470.3856	154.6271	2.322263
F4	14.1436	25.9679	6.0893	0.586869	3.8694	6.7903	1.6306	1.755044
F5	20.2093	334.0200	444.3179	0.640554	131.6334	558.8711	369.8338	1.536910
F6	5.0067E-5	1.4954E-04	4.4487E-04	0.643048	4.1500	19.0857	13.9022	1.54936
F7	0.0209	0.0364	0.0129	0.763583	0.0171	0.0352	0.0129	3.745837

F	ICA				PSO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	7.2247E-06	1.3205E-04	1.7340E-04	1.48590	23.2860	76.9335	31.9150	0.355952
F2	3.1098E-04	0.0011	6.1653E-04	1.4804	2.3918	7.3666	3.0318	0.3660
F3	614.0714	1.3905E+03	559.3069	2.540456	354.3102	2.6799E+03	2.6157E+03	1.236538
F4	4.1320	9.4623	2.7549	1.563988	4.3508	7.5786	1.7179	0.47463
F5	26.5015	205.4637	299.0098	1.584770	160.6205	1.4821E+03	1.7592E+03	0.461078
F6	3.1110E-05	2.2500E-04	1.6011E-04	1.600717	26.2675	88.1072	68.0237	0.481432
F7	0.0267	0.0226	0.0768	1.724640	0.0038	0.0355	0.0240	0.608523

F	ALO				GWO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	6.0598E-05	3.5790E-04	4.3611E-04	29.4673	9.0481E-12	3.2320E-11	2.5277E-11	0.2945
F2	0.6943	48.7203	46.1718	28.9521	1.0069E-13	3.6499E-13	2.0871E-13	0.3168
F3	963.2682	2.6094E+03	1.0290E+03	30.720294	4.3817E-05	0.0060	0.0097	1.1417
F4	9.0764	15.8283	4.4187	28.354286	4.3976E-06	2.7134E-05	2.7926E-05	0.3805
F5	23.4972	196.5491	191.7828	28.414984	36.0767	37.0454	0.8404	0.4298
F6	6.9376E-05	2.2413E-04	1.1807E-04	28.238345	1.0001	1.4185	0.4020	0.4354
F7	0.0966	0.1734	0.0512	28.42491	0.0016	0.0028	9.1592E-04	0.5522

F	CSA				Firefly			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	3.6952	12.8494	6.8743	0.6817	0.0049	0.0118	0.0045	2.208220
F2	2.3404	4.9688	2.1806	0.596314	0.1556	0.3733	0.1033	2.2510
F3	632.8637	1.5724E+03	446.1137	1.421566	920.4498	2.0394E+03	835.4943	3.065636
F4	7.0618	14.1226	3.7242	0.635692	0.0501	0.0807	0.0164	2.331268
F5	248.7507	1.3990E+03	972.8326	0.686697	26.5929	133.8007	161.0121	2.33665
F6	2.5103	9.2432	5.0420	0.680190	0.0046	0.0130	0.0051	2.308121
F7	0.0504	0.1476	0.0723	0.797525	0.0099	0.0320	0.0172	2.446989

all the test functions except F2, LAPO has the best answers and for F2 has the second best answers among different methods. In addition, standard deviations of the trials are very low which reveals robustness of the proposed method. It can be said that the proposed method has competitive results for solving the unimodal test functions. Furthermore, in terms of CPU time consuming, the proposed LAPO method is not the fastest method, but has an acceptable computational time.

3.2. Exploration analysis

Another feature of a good optimization method is to find the best global answer and does not get stuck at a local optimum. The multimodal and fixed-dimension multimodal test functions are suitable for evaluating the exploration characteristic of a method. Tables 6 and 7 illustrate the results of different methods for the two mentioned groups of test functions. As it can be seen, the proposed LAPO outperforms other methods for all the test function in Table 6. Moreover, for the fixed-dimension multi modal test functions, the proposed method has very competitive results. Thus, it could also be stated that the proposed method has an excellence global search characteristic.

3.3. Global minimum finding

The fourth group of the test functions i.e., the composite ones are challenging functions by which both exploitation and exploration characteristics of a method could be benchmarked simultaneously. To find the best overall answer of these kinds of functions, a method with the ability of excellence global search and local search is needed. The results of different methods for solving the composite test functions are listed in Table 7. It is obvious that the proposed

Table 6
results of different method for solving the multimodal test functions.

F	LAPO				LSA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-1.0613E+4	-1.036E+4	1.9994E+03	0.927065	−8502.8	−8131.6	281.98	13.175
F9	0	1.53344	3.70144	1.095045	46.763	64.274	13.636	13.704
F10	9.5009E-09	5.8694E-08	5.0496E-08	1.130083	1.3404	2.8488	0.78898	13.836
F11	1.7764E-15	1.5914E-13	2.9758E-13	1.217667	2.0206e-14	0.0066515	0.007159	14.139
F12	6.8458E-09	0.0104	0.0311	2.197772	5.287e-06	0.18718	0.3595	15.362
F13	5.5452E-07	0.0098	0.0240	2.048834	3.0594e-3	0.16322	0.30357	15.358
F	DE				SFLA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-1.104E+04	-9.916E+03	690.4544	0.570069	-9.2376E+03	7.5146E+03	709.8488	1.84541
F9	18.9042	37.6106	11.9314	0.677520	25.7164	47.2634	15.0827	1.96860
F10	0.9313	2.6724	1.1995	0.712424	4.1208	6.5725	1.5489	1.81110
F11	2.3758E-04	0.0491	0.0595	0.821489	0.9253	1.1603	0.1557	1.73998
F12	0.0038	3.4061	3.1716	1.134543	2.4217	7.0897	2.3381	2.28280
F13	3.1951E-04	11.7909	10.4224	1.119140	1.7653	30.4043	19.1206	2.61907
F	ICA				PSO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	−1.126E+04	−9.671E+03	824.1584	1.465802	−9.3168E+03	−7.771E+03	1.0051E+03	1.02069
F9	5.9779	14.3925	3.5594	1.546710	44.6941	84.6695	22.4398	1.12109
F10	0.0011	0.0061	0.0057	1.570927	2.8692	4.6089	1.0713	1.15249
F11	2.7347E-05	0.0466	0.0339	1.638158	1.0935	1.6135	0.4053	1.19303
F12	2.4743E-08	1.1269E-05	4.2876E-05	2.10093	0.6399	3.1318	1.7733	1.61608
F13	1.5961E-06	0.0015	0.0037	2.190709	2.3482	14.4184	11.1284	1.60883
F	ALO				GWO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-5.5636E+03	-5.4872E+03	59.3746	28.954543	-8.5581E+03	-6.964E+03	1.665E+03	0.4936
F9	40.7934	75.9153	20.8421	29.172624	5.6843E-05	3.1699	6.6764	0.4701
F10	1.1567	2.5447	1.0413	28.910295	5.7643E-08	1.5083E-07	6.5140E-08	0.4932
F11	0.0061	0.0290	0.0131	29.139031	0.0012	0.0060	0.0109	0.5270
F12	7.4959	12.2652	5.9904	30.796473	0.0438	0.0753	0.0255	0.9347
F13	0.0344	15.5303	22.1428	30.290157	0.9754	1.2997	0.1932	0.9449
F	CSA				Firefly			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-8.909E+03	-8.5464E+03	172.9980	3.620904	-7.6344E+03	-6.216E+03	697.3859	2.28836
F9	93.0875	116.9196	10.9529	3.449429	16.1872	28.3631	5.8445	2.35327
F10	10.5414	13.1248	1.2356	3.637010	0.0294	0.0523	0.0157	2.39521
F11	1.3221	1.5594	0.1394	3.770781	0.0037	0.0057	0.0013	2.42128
F12	3.9863	5.5999	0.9444	4.234347	8.0382E-05	2.2886E-03	1.2389E-03	3.12889
F13	13.5315	19.3205	2.6419	4.307192	0.0013	0.0023	7.1625E-04	3.15469

method outperforms all the other methods in some test functions such as F24 and F28. In F25, although the best result of firefly is better than the proposed method, the average of 30 different trials of the proposed method is better than the other methods. For some other test functions, the proposed method has competitive results compared to the other methods. From these results, it can be concluded that not only the proposed method is able to find the overall best answer of a function among lots of local optimum points, but also it can reach nearly the exact best answer by excellence local search.

3.4. Large scale optimization problems

Another feature of a good optimization method is the ability of leading lots of variables towards the best answer. In other words, the optimization methods must be testified on solving the large scale optimization problems. In this section, the proposed LAPO method is evaluated for solving the large scale optimization problems. To do this, the unimodal, and multimodal test functions with 200 variables are employed. The optimization algorithms are performed in two cases. In the first case, the population size is 500 and the methods are executed for 2000 iterations. In the second case, the functions are solved by the methods with population size of 40 and 500 iterations. In the second case, the capability of the proposed method is testified for solving the large scale problems with low population size and lower number of iterations. Meanwhile, 10 different trials are performed for each method.

The optimization results for large scale problems for case 1 are shown in Tables 9 and 10. As it can be seen, the proposed LAPO method outperforms the other methods for F4, F5, F6, and F8, and has competitive results for other test functions. For instance, F1, F2, and F3 give far better results than most of the methods and it is ranked as the second best method after GWO. Moreover, the small standard deviation values show robustness of the proposed method. For the F9, F10, and F11 the proposed method outperforms the other ones and for the other test functions, it has very competitive results.

Results of solving the unimodal and multimodal test functions with fewer population and iteration are listed in Tables 11 and 12. For the unimodal test functions, the proposed LAPO method out-

Table 7
results of different method for solving the fixed-dimensional multimodal test functions.

F	LAPO				LSA			
	best	ave	Std	time(s)	best	ave	std	Time(s)
F14	0.9980	0.9980	5.7495E-08	0.767174	0.998	1.1968	0.41912	3.7214
F15	3.0749E-04	5.5811E-04	2.2495E-04	0.167664	0.000307	0.00053523	0.00043114	3.1152
F16	−1.0316	−1.0316	1.4460E-07	0.134042	−1.031628	−1.031628	1.95824e-16	0.6379
F17	0.3979	0.3983	4.8405E-04	0.123149	0.3979	0.3979	0	0.1835
F18	3.0000	3.0000	7.5626E-16	0.132780	3.001	3.001	1.03620e-15	1.788
F19	−3.8628	−3.8628	8.5422E-16	0.198560	−0.305	−0.3004	7.40148e-17	1.270
F20	−3.3220	−3.2729	0.0571	0.201384	−3.321	−3.27443	0.061396	1.383
F21	−10.1532	−9.6960	0.8042	0.245951	−10.053	−8.385	2.9225	1.632
F22	−10.4029	−10.1728	0.6905	0.276474	−10.402	−6.0452	3.101	2.000
F23	−10.5364	−10.2295	0.6352	0.316722	−10.536	−7.7078	3.730	2.457

F	DE				SFLA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F14	0.9980	1.4568	1.6536	0.4269	0.9980	2.4509	1.3675	0.5811
F15	3.0749E-04	0.0060	0.0087	0.1007	3.0764E-04	7.4043E-04	4.9286E-04	0.2595
F16	−1.0316	−1.0316	5.5880E-16	0.0838	−1.0316	−1.0316	2.1065E-16	0.3186
F17	0.3979	0.3979	0	0.0787	0.3979	0.3979	0	0.3166
F18	3.0000	3.0000	1.7123E-15	0.0830	3.0000	3.0000	3.7891E-15	0.3135
F19	−3.8628	−3.8628	2.5252E-15	0.1172	−3.8628	−3.8628	4.8648E-16	0.2631
F20	−3.3220	−3.2625	0.0594	0.1190	−3.3220	−3.2830	0.0584	0.2748
F21	−10.1532	−5.1345	2.9569	0.1501	−10.1532	−6.6493	3.5714	0.2787
F22	−10.4029	−6.0078	3.4431	0.1587	−10.4029	−8.3021	3.2189	0.2885
F23	−10.5364	−7.2796	3.7594	0.1758	−10.5364	−5.7586	3.9045	0.3049

F	ICA				PSO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F14	0.9980	0.9980	4.3414E-14	0.635135	0.9980	1.0311	0.1784	0.379351
F15	4.0430E-04	0.0021	0.0028	0.311820	3.0750E-04	0.0056	0.0084	0.075062
F16	−1.0316	−1.0316	5.0634E-16	0.288696	−1.0316	−1.0316	2.7133E-11	0.069478
F17	0.3981	0.3983	3.8523E-03	0.287647	0.3979	0.3979	6.1259E-12	0.055647
F18	3.0000	3.0000	2.1353E-11	0.285817	3.0000	5.7000	14.5399	0.059684
F19	−3.8628	−3.8628	3.1377E-12	0.329280	−3.8628	−3.8625	0.0014	0.091313
F20	−3.3220	−3.3220	1.2090E-08	0.318563	−3.3220	−3.2755	0.0681	0.093961
F21	−10.1532	−7.4016	3.4354	0.350146	−10.1532	−6.9746	3.4764	0.119316
F22	−10.4029	−9.6392	2.2911	0.367494	−10.4029	−7.8672	3.3898	0.133108
F23	−10.5364	−8.4297	3.2277	0.386303	−10.5364	−7.0843	3.8065	0.152079

F	ALO				GWO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F14	0.9980	5.6102	3.5498	0.689122	0.998	4.3865	3.7196	0.745630
F15	0.0011	0.0041	0.0065	0.602195	3.0846E-04	0.0033	0.0069	0.155647
F16	−1.0316	−1.0316	3.4120E-13	0.398864	−1.0316	−1.0316	2.2117E-04	0.122084
F17	0.3979	0.3979	4.7070E-13	0.293945	0.3979	0.3979	3.6777E-05	0.123547
F18	3.0000	3.0000	5.5923E-12	0.397778	3.0000	3.0005	4.8378E-04	0.120457
F19	−3.8628	−3.8526	0.0302	0.524160	−3.8628	−3.8610	0.0025	0.182076
F20	−3.3220	−3.2622	0.0784	0.809817	−3.3219	−3.2421	0.0853	0.219921
F21	−5.1008	−3.8729	1.2007	0.642313	−10.1502	−8.6959	2.6781	0.235932
F22	−10.4029	−7.2446	3.2225	0.650708	−10.4003	−9.6799	2.1149	0.258903
F23	−10.5364	−5.9964	3.7391	0.666060	−10.5326	−10.0440	1.7398	0.297190

F	CSA				Firefly			
	best	ave	std	time(s)	best	ave	std	Time(s)
F14	0.9980	1.0530	0.1915	0.407883	0.9980	5.0897	2.8480	0.713667
F15	5.4387E-04	0.0013	4.9224E-04	0.106910	7.7204E-04	0.0054	0.0045	0.419765
F16	−1.0316	−1.0316	1.2697E-07	0.095428	−1.0316	−1.0315	5.5489E-04	0.412668
F17	0.3979	0.3979	7.2995E-06	0.090751	0.3979	0.4016	0.0199	0.403244
F18	3.0000	3.0001	1.8279E-04	0.092633	3.0000	3.0000	2.1246E-07	0.401207
F19	−3.8628	−3.8627	1.0269E-04	0.124587	−3.8628	−3.8578	0.0081	0.433102
F20	−3.3179	−3.2888	0.0258	0.129720	−3.3217	−3.2364	0.0953	0.447982
F21	−10.1029	−9.0731	0.9759	0.149500	−10.1532	−5.6266	3.5789	0.456901
F22	−10.3858	−9.4701	0.7201	0.167090	−10.4029	−9.3773	2.6150	0.476818
F23	−10.4778	−9.1012	0.8841	0.181062	−10.5364	−10.5361	0.0015	0.487832

performs the other methods for F1-F7 except F6 which the method has a competitive result. The proposed method has also far better results for F9-F13 and competitive result for F8. Based on these results, the method has excellence results for large scale optimization method even with a small population size and low number of iterations.

Table 8
results of different method for solving the composite test functions.

F	LAPO				LSA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F24	0	7.7214E-23	4.2143E-23	264.797980	12.14	120.000	103.279	141.483
F25	2.7340	29.87214	32.23484	244.215891	4.446	148.26	137.7999	285.57
F26	116.0883	175.86457	90.15781	247.254191	147.312	257.40	115.45	268.814
F27	265.2541	312.1458	67.9681	277.169869	286.999	483.162	136.21	309.031
F28	0	45.8928	56.4218	279.241119	3.4655	155.64	154.408	312.72
F29	500.000	544.8186	119.6597	254.226818	500.411	782.051	193.86	323.642

F	DE				SFLA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F24	4.7080E-24	52.2576	69.0770	131.1335	2.3809E-06	70.0002	78.1025	102.3544
F25	7.4468	94.8094	91.3860	114.5313	52.1968	198.5301	120.2203	101.3876
F26	163.9931	223.8490	81.1158	111.2575	273.8043	400.3250	116.3799	95.6668
F27	284.2209	326.3868	43.1065	143.6823	408.1973	443.0856	35.3076	137.0135
F28	4.3712	92.2381	83.2146	140.4139	12.9486	82.5981	85.9617	182.5330
F29	500.0000	741.9855	196.6948	136.8673	501.9177	697.4181	188.0246	114.8238

F	ICA				PSO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F24	1.5873E-18	70.0000	100.4988	160.0497	0.3053	186.3420	138.9956	132.5921
F25	22.9368	174.3427	78.8243	133.2649	28.2429	203.1709	159.6724	114.8665
F26	160.9427	363.9297	122.3701	122.9039	271.7938	396.7644	86.3684	111.5002
F27	471.4093	599.9630	69.9404	143.7568	337.6973	474.7387	148.4452	139.3360
F28	12.4696	59.2129	28.9372	152.4287	40.5131	241.2266	123.3227	136.3697
F29	506.3825	784.8621	181.8203	149.5143	515.2240	833.4793	150.7666	136.9043

F	ALO				GWO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F24	1.2055E-09	1.6480E-09	6.4372E-10	148.6606	80.8012	224.3565	128.8061	34.4861
F25	140.0910	162.7753	32.4937	135.2122	172.4628	302.5475	108.3346	125.3465
F26	229.6219	439.6753	236.0181	130.2807	150.6736	303.6108	182.9697	118.2997
F27	334.1691	485.3973	136.6477	139.3494	287.2757	388.5935	138.7823	137.3359
F28	3.4804	4.8742	1.5446	138.7856	11.1285	73.1323	65.5494	138.5388
F29	500.3315	769.7003	233.2831	139.4281	902.3750	905.2132	2.3131	135.0984
F29	411.1958	525.5262	97.3582	142.2612	500.4599	782.2253	184.0946	143.8334

F	CSA				Firefly			
	best	ave	std	time(s)	best	ave	std	Time(s)
F24	0.0034	2.0362	3.3849	132.3139	0.1841	40.0000	48.9898	138.7930
F25	5.0744	42.4796	50.4552	124.8886	1.7922	78.4052	78.7343	123.2573
F26	214.0799	253.2691	38.7039	115.3250	109.3281	146.0378	29.8617	117.3834
F27	287.6394	335.8235	24.9870	130.8379	268.6814	284.2349	14.8796	139.4033
F28	5.8431	9.7506	2.2546	139.9418	1.1924	52.6011	49.0204	141.2526
F29	411.1958	525.5262	97.3582	142.2612	500.4599	782.2253	184.0946	143.8334

3.5. Convergence behavior

Another feature of an optimization method is the convergence behavior. In the population based optimization methods, a number of solutions are defined in the predefined range in the first step. Then, these solutions are updated by some mathematical equations in order to find the best answer of the problem. Movement of the solutions along the search space towards the best answer is important. In other words, the good updating procedure of a method results in finding the best answer in lower iterations. In this section, this criterion of the proposed method is evaluated. First of all, the search history and the trajectory of the first variable of the first solution are depicted in Fig. 12 for some test functions. In this figure, the first column depicts the 2-D version of the function, the second column is the search history, the third one is the parameter S which is illustrated in Eq. (8), the forth column is the trajectory of the first variable of the first solution, fifth column is the average of the fitness function, and the last column is the convergence behavior. As it can be seen in the second column, the search space is evaluated by different solutions very well and most of the solutions lead to the best answer. Moreover, column four reveals that the variables lead to the best value in early iterations. To show the good convergence behavior of the proposed method, the convergence behavior of the proposed method is compared with some other methods for some of the test functions in Fig. 13. As it can be seen, the proposed method hits a better answer compared to other methods. Moreover, it reaches the best answer in lower iterations with regard to other methods.

To sum up, by the four tables (i.e., Tables 5–8), the proposed method is evaluated from different points of view. By the unimodal test functions the exploitation feature of the proposed method is benchmarked. Using the multimodal test functions, the global search characteristic i.e., exploration of the method is also evaluated. Finally, the exploitation and exploration of the proposed method are testified simultaneously using the composite test functions. Moreover, robustness of the proposed method is evaluated by obtaining the standard deviation of 30 different trials. Small standard deviation values of these 30 trials reveal high robustness of the proposed method. The computational time of the proposed method is also taken into consideration and the relatively low time

Table 9
results of different methods for solving the 200-dimensional version of unimodal test functions, with the population size of 200 and the 2000 iterations.

F	LAPO				ABC			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	3.7643E-29	8.1296E-29	3.2204E-29	40.1957	8.9499E+03	1.4148E+04	1.9222E+03	27.5773
F2	1.4339E-15	1.8098E-15	2.7520E-16	41.3853	73.0196	81.6573	5.8675	28.9298
F3	0.4160	2.6872	3.5371	315.1508	4.7612E+05	6.5662E+05	7.3296E+04	163.0073
F4	2.0925E-10	7.2703E-10	3.3558E-10	45.4820	90.5305	92.4391	0.7630	28.6786
F5	192.6322	193.0862	0.3316	47.1548	7.2876E+06	1.1209E+07	1.9375E+06	30.9131
F6	0.0283	0.0412	0.0072	44.7716	1.0346E+04	1.3273E+04	2.1238E+03	30.4218
F7	2.8619E-05	1.8284E-04	1.3097E-04	71.8251	23.0871	32.1411	5.5446	43.1471

F	DE				SFLA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	0.0094	3.4267	2.9338	18.4324	164.9749	275.9424	62.8101	14.1161
F2	0.5247	5.9106	6.6286	19.3465	17.2986	21.7154	1.9108	15.8777
F3	3.8709E+05	4.6516E+05	4.7067E+04	154.4452	5.5593E+03	7.1881E+03	1.0812E+03	82.5222
F4	46.7385	52.1769	2.8675	20.8550	10.0687	11.7461	0.7967	32.3306
F5	3.6550E+03	2.7380E+04	3.0000E+04	21.3643	3.3207E+03	4.2391E+03	664.7247	20.5108
F6	0.0137	1.7010	3.0727	20.9296	184.4612	270.8978	74.0124	16.4177
F7	0.2162	0.6486	0.5584	34.2304	0.2077	0.2268	0.0140	62.8454

F	ICA				PSO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	3.6177E-05	4.9125E-05	1.107E-05	29.8373	22.2761	42.9502	10.0862	9.0901
F2	0.0018	0.0024	4.8728E-04	30.3026	3.1852	8.2847	2.8917	10.0612
F3	2.5280E+04	3.4281E+04	4.7509E+03	187.6643	1.9931E+03	6.5972E+03	4.6773E+03	147.541
F4	11.5323	14.6959	1.9066	29.9090	2.2303	2.8657	0.6449	11.4063
F5	546.6862	685.3896	85.6767	32.8787	340.4599	540.7680	108.3468	11.8282
F6	2.7313E-05	4.4321E-05	1.2661E-05	30.2679	47.2150	80.7581	16.8446	12.0781
F7	0.1920	0.2377	0.0330	45.2809	4.4161E-04	0.0114	0.0098	25.0383

F	ALO				GWO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	3.4918E-03	5.3074E-03	1.8514E-03	7943.24	4.5190E-38	2.2323E-37	1.2855E-37	22.4423
F2	652.3152	889.2314	312.2145	7542.31	1.8924E-20	5.2932E-20	3.0364E-20	23.2889
F3	2.857E+03	3.2144E+03	410.3142	24231.23	2.4381E-08	2.0677E-05	4.1389E-05	156.0946
F4	50.1289	52.4897	3.32458	8023.197	9.2622E-08	0.0014	0.0037	23.5751
F5	183.5943	203.1251	42.2674	8415.12	194.8496	196.6516	0.7670	26.1214
F6	2.124E-03	0.01437	0.0097	7625.23	17.5594	19.9656	1.2912	25.2861
F7	1.3985	3.2457	1.9875	8326.15	1.2338E-04	3.2376E-04	1.0256E-04	38.8676

F	CSA				Firefly			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	149.1530	196.5869	31.0417	18.4890	0.1528	196.5869	0.0068	381.8076
F2	1.0000E+10	1.0000E+10	0	18.4920	2.5763	1.0000E+10	0.1915	385.0120
F3	1.6174E+05	1.9185E+05	2.0367E+04	156.4794	6.5049E+04	1.9185E+05	5.8542E+03	527.6232
F4	12.7108	14.0077	0.9181	20.0465	52.1716	14.0077	4.3816	372.9640
F5	5.3684E+03	6.7598E+03	993.6765	21.4486	203.4103	6.7598E+03	115.6933	383.8150
F6	161.0285	192.7601	21.0184	21.0825	0.1427	192.7601	0.0079	378.1490
F7	0.5030	0.6048	0.0849	34.2627	0.0859	0.6048	0.0261	393.2078

computation of the proposed method is concluded. In addition, the ability of the proposed LAPO method on solving the large scale optimization algorithm is tested using 200-dimensional version of the unimodal and multimodal test functions. The proposed method has an excellence performance for large scale problems even with low number of population size and iterations. Furthermore, the convergence behavior of the proposed method is investigated by plotting the search history and trajectory of the first variable of the first solution during the iterations. Moreover, the convergence behavior of the proposed method is compared to that of some other methods. The good final result and fast convergence to the best answer are concluded from this investigation.

Now, the proposed method is evaluated by some real engineering optimization problem. To do this, five classical mechanical engineering problems and a high constraint electrical optimization problem are employed. These evaluations are performed as follows.

3.6. Solving engineering problems by LAPO

In this part, the proposed LAPO is used for solving 6 different engineering optimization problems. The first five problems are mechanical optimization problem including tension/compression spring, welded beam, pressure vessel designs, Gear train design, and Cantilever beam design. The sixth problem is a high constraint electrical optimization problem known as Optimal Power Flow. These problems are associated with some equality and inequality constraints which must be satisfied during solving the problem. In this study, the constraints are handled using penalty factors. In other word, whenever the constraints are violated for a solution, a big fitness function is designated to the solution. Thus, a solution would be worst among the population and would be discarded in further iterations.

Table 10
results of different methods for solving the 200-dimensional version of multimodal test functions, with the population size of 200 and the 2000 iterations.

F	LAPO				ABC			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-5.617E+04	-5.5361E+04	527.2355	52.582	-5.87116E+04	-5.799E+04	516.7504	34.319
F9	0	0	0	53.500	5.5206E+02	581.1772	15.0524	33.449
F10	8.8817E-16	8.8818E-16	0	54.894	12.057293	12.5316	0.2296	35.306
F11	0	0	0	57.355	1.10033E+02	122.2712	9.9774	35.508
F12	8.64884E-05	1.1944E-04	2.2692E-05	118.133	7.99036E+05	2.7655E+06	9.0463E+05	65.586
F13	0.071735	0.1748	0.0712	117.209	2.84168E+06	1.9198E+07	1.0240E+07	65.177

F	DE				SFLA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-6.106E+04	-5.8193E+04	2.162E+03	25.2330	-4.8685E+04	-3.884E+04	4.8031E+03	39.5269
F9	450.7332	570.4642	86.3149	24.4353	405.3934	474.7053	49.5121	27.8925
F10	5.5220	8.0100	1.5294	24.9623	5.4943	6.9422	0.7936	33.8247
F11	0.0844	0.7206	0.4426	25.8570	3.0548	3.5738	0.5035	20.7398
F12	8.9614	10.6343	1.6729	56.7757	4.0529	5.4245	0.6846	64.4823
F13	187.8197	229.3965	27.4957	56.5316	180.8564	226.0896	26.3839	66.7729

F	ICA				PSO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-5.432E+04	-5.0741E+04	2.390E+03	33.7845	-4.1848E+04	-3.885E+04	3.4655E+03	29.3711
F9	292.5286	422.3477	59.2422	33.0526	221.1672	424.0131	99.6931	28.2308
F10	0.0123	0.0620	0.0834	33.9011	0.5767	1.4667	0.3718	29.3400
F11	1.0037E-05	7.5631E-04	0.0022	35.6621	1.0995	1.3244	0.1599	30.7325
F12	9.2675E-05	0.1099	0.3190	69.7339	0.3038	0.4822	0.1204	61.7301
F13	2.3728E-05	5.0033E-05	2.6147E-05	70.2588	20.7460	28.2002	3.7339	61.9184

F	ALO				GWO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-3.8756E+04	-3.754E+04	231.24	8.976E+03	-3.4619E+04	-3.046E+04	1.8936E+03	25.1740
F9	310.2421	572.89	110.158	8.4474E+03	0	0	0	27.2447
F10	7.53781	8.19321	1.3152	8.759E+03	2.9310E-14	3.6060E-14	3.7091E-15	27.9993
F11	0.02389	0.0312	0.1134	8.8255E+03	0	1.1102E-17	3.3307E-17	29.7272
F12	4.32151	6.5876	3.5487	1.8551E+04	0.2177	0.2606	0.0264	61.0197
F13	0.8965	1.3985	1.3215	1.9231E+04	11.7505	12.8678	0.5411	60.2916

F	CSA				Firefly			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-4.148E+04	-3.9512E+04	921.6486	24.4254	-4.0562E+04	-3.492E+04	3.1571E+03	380.843
F9	1.0443E+03	1.0927E+03	30.5688	23.7244	166.6876	194.7440	15.6631	378.833
F10	9.0663	12.3232	2.4897	25.1727	0.0905	0.1003	0.0061	385.306
F11	2.1657	2.6432	0.2161	25.7426	0.0260	0.0300	0.0023	386.951
F12	3.0427	3.2927	0.2055	56.703	2.1867E-04	0.0036	0.0100	415.116
F13	99.6652	126.5403	17.0453	56.4433	0.0151	0.0181	0.0026	412.845

3.6.1. *Tension/compression spring design*

This problem is to minimize the weight of a tension/compression spring and the problem is depicted in Fig. 14 [72–74]. In this problem, three decision variables must be obtained optimally and a number of constraints such as shear stress, surge frequency, and minimum deflection must be satisfied during solving procedure. The formulation of the problem is as follows:

Function
$$f(x) = (x_3+2)x_2x_1^2 \quad (10)$$

Inequality constraints
$$g_1(x) = 1-\frac{x_2^3x_3}{71785x_1^4} \le 0$$
$$g_2(x) = \frac{4x_2^2-x_1x_2}{12566(x_2x_1^3-x_1^4)}+\frac{1}{5108x_1^4} \le 0 \quad (11)$$
$$g_3(x) = 1-\frac{140.45x_1}{x_2^2x_3} \le 0$$
$$g_4(x) = \frac{x_1+x_2}{1.5}-1 \le 0$$

Variable range
$$0.05 \le x_1 \le 2.00$$
$$0.25 \le x_2 \le 1.30 \quad (12)$$
$$2.00 \le x_3 \le 15.0$$

where x_1, x_2, and x_3 are the decision variables as wire diameter (d), mean coil diameter (D), and the number of active coils (N). This problem is solved by different methods including meta-heuristic methods and mathematical methods. The results are listed in Table 13. As it can be seen, the best results are obtained by the proposed LAPO algorithm.

3.6.2. *Welded beam design*

This problem is to minimize the fabrication cost of a welded beam as illustrated in Fig. 15 [76]. The decision variables are thickness of weld (h), length of attached part of bar (l), the height of the bar (t), and thickness of the bar (b). The constraints are also shear stress (s), Bending stress in the beam (h), buckling load on the bar (Pc), end deflection of the beam (d), and Side constraints. The mathematical formulation of this problem is as follows:

Function
$$f(x) = 1.10471x_1^2x_2+0.04811x_3x_4(14+x_2) \quad (13)$$

Inequality constraints
$$g_1(x) = \tau(x)-\tau_{max} \le 0$$
$$g_2(x) = \sigma(x)-\sigma_{max} \le 0$$
$$g_3(x) = \delta(x)-\delta_{max} \le 0$$
$$g_4(x) = x_1-x_4 \le 0 \quad (14)$$
$$g_5(x) = P-P_c(x) \le 0$$
$$g_6(x) = 0.125-x_1 \le 0$$
$$g_7(x) = 1.10471x_1+0.04811x_3x_4(14+x_2)-5 \le 0$$

Variable range
$$0.1 \le x_1 \le 2.00$$
$$0.1 \le x_2 \le 10 \quad (15)$$
$$0.1 \le x_3 \le 10$$
$$0.1 \le x_4 \le 2$$

where

$$\begin{aligned}
&\tau(x)=\sqrt{(\tau')^2+2\tau'\tau''\frac{x_2}{2R}+(\tau'')^2}\\
&\tau'=\frac{P}{\sqrt{2}x_1x_2},\ \tau''=\frac{MR}{J},M=P(L+\frac{x_2}{2})\\
&R=\sqrt{\frac{x_2^2}{4}+\left(\frac{x_1+x_3}{x}\right)^2}\\
&J=2\{\sqrt{2}x_1x_2[\frac{x_2^2}{4}+\left(\frac{x_1+x_3}{x}\right)^2]\}\\
&\sigma(x)=\frac{6PL}{x_4x_3^2},\ \delta(x)=\frac{6PL^3}{Ex_4x_3^2}\\
&P_c(x)=\frac{4.013\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}(1-\frac{x_3}{2L}\sqrt{\frac{E}{4G}})\\
&P=6000lb, L=14in., \delta_{max}=0.25in., E=30\times10^6psi,\\
&G=12\times10^6psi, \tau_{max}=13600psi, \sigma_{max}=30000psi
\end{aligned}\tag{16}$$

This problem is also solved by different methods and the results are shown in Table 14. As it can be seen, the result of the proposed method is the best one; it outperforms most of the methods.

3.6.3. *Pressure vessel design*

This problem which is presented in Fig. 16, is to minimize the total cost including material, forming, and welding of a cylindrical vessel. Both ends of the vessel are capped, and the head has a hemispherical shape. Variables of this problem are thickness of the shell (Ts), thickness of the head (Th), inner radius (R), and length of the cylindrical section without considering the head (L). This problem is formulated as follows:

Function $$f(x)=0.6224x_1x_3x_4+1.7781x_3^2x_2+3.1661x_1^2x_4+19.84x_1^2x_3 \tag{17}$$

Inequality constraints
$$\begin{aligned}
&g_1=-x_1+0.0193x_3\le0\\
&g_2=-x_3+0.0095x_3\le0\\
&g_3=-\pi x_3^2x_4-\frac{4}{3}\pi x_3^3+1296000\le0\\
&g_4=x_4-240\le0
\end{aligned}\tag{18}$$

Table 11
results of different methods for solving the 200-dimensional version of multimodal test functions, with the population size of 40 and the 500 iterations.

F	LAPO				ABC			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	1.0288E-12	1.8090E-09	6.1216E-09	1.871493	2.4360E+05	2.7219E+05	1.2199E+04	0.856961
F2	2.0087E-07	3.5023E-06	4.3034E-06	1.901820	1.4795E+14	2.9094E+22	1.0439E+23	6.386335
F3	0.9660	6.7051	5.7391	15.430342	1.1942E+06	1.2362E+06	4.2002E+04	7.589553
F4	9.8895E-06	2.4845E-05	1.4955E-05	2.108931	93.1279	95.5376	1.2093	0.852726
F5	196.3227	197.3273	0.5454	2.148707	8.1614E+08	9.4098E+08	7.2997E+07	0.977426
F6	13.7919	16.5038	1.3669	2.098711	2.3291E+05	2.7077E+05	1.4489E+04	0.931721
F7	3.5309E-04	0.0010	4.5410E-04	3.332803	2.3945E+03	3.0210E+03	310.5942	1.525904

F	DE				SLFA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	3.0715E+04	4.4865E+04	8.7919E+03	0.984315	1.8473E+03	2.8660E+03	550.3588	2.794788
F2	171.2055	232.9325	36.3980	1.010780	49.9520	57.6806	3.8804	3.256660
F3	5.4828E+05	6.8268E+05	8.2453E+04	7.839305	2.4304E+04	3.2983E+04	6.0560E+03	21.028389
F4	52.8730	59.1089	3.6240	1.049797	12.0156	15.4449	1.2661	5.204148
F5	2.6482E+07	5.8002E+07	1.8757E+07	1.142108	9.0164E+04	1.4844E+05	4.3592E+04	3.199952
F6	3.0321E+04	4.8823E+04	1.1315E+04	1.094466	2.1183E+03	2.8351E+03	533.9642	3.164061
F7	64.1714	151.7041	55.7205	1.646033	0.4357	0.6017	0.0838	23.22358

F	ICA				PSO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	1.5421E+03	2.4932E+03	427.4432	1.562873	2.3606E+03	4.6042E+03	1.6342E+03	0.542811
F2	51.3144	73.8622	9.1490	1.562673	50.9413	88.4166	24.2348	0.554422
F3	1.0483E+05	2.4815E+04	2.3915E+04	8.660863	4.9300E+04	1.7849E+05	9.6867E+04	6.996546
F4	36.5269	43.1785	3.5339	1.614045	12.3977	20.2241	3.2475	0.606078
F5	1.3918E+05	2.7371E+05	8.9298E+04	1.659135	1.1221E+05	4.1913E+05	3.2195E+05	0.638427
F6	1.8802E+03	2.4839E+03	598.6150	1.612699	1.3346E+03	5.8691E+03	2.5229E+03	0.645103
F7	4.2427	0.7137	5.6604	2.310599	0.4911	1.9675	1.5525	1.178825

F	ALO				GWO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	1.9446E02	2.8545E02	7.8596E01	187.21457	5.0773E-09	1.8412E-08	1.1280E-08	1.0913
F2	273.293	325.324	91.257	190.15789	7.0949E-06	9.0642E-06	1.4596E-06	1.1181
F3	2.6191E+04	3.1931E+04	1.0249E+04	182.12503	6.3143E+03	1.5097E+04	8.3781E+03	7.8298
F4	20.2158	175.21458	142.5678	184.174021	14.4764	21.5119	5.7528	1.2095
F5	3.9246E+06	7.8716E+06	6.0127E+06	194.995556	195.8662	197.7457	0.7064	1.3368
F6	1.8288E+04	2.9352E+04	6.2965E+03	185.243149	26.2291	27.9390	1.3802	1.2796
F7	14.7902	25.6137	9.0479	197.237721	0.0060	0.0105	0.0029	1.9543

F	CSA				Firefly			
	best	ave	std	time(s)	best	ave	std	Time(s)
F1	1.4190E+04	2.2656E+04	4.2071E+03	1.034282	0.9584	2.3924	1.6209	3.808463
F2	10E+10	10E+10	0	2.27514	42.5075	69.0379	13.9463	3.836813
F3	3.9105E+05	5.2689E+05	4.4417E+04	45.196943	1.6087E+05	2.3346E+05	3.5424E+04	11.036595
F4	3.6254	47.6729	33.3905	5.604903	70.0208	74.2297	2.4547	3.604922
F5	5.4107E+06	8.5370E+06	1.8119E+06	6.129917	1.4679E+03	3.6955E+03	2.5851E+03	4.065498
F6	1.6471E+04	2.3203E+04	3.6669E+03	5.977193	0.9638	1.9872	0.8689	4.118451
F7	20.0720	32.8526	8.7539	1.722193	1.1936	2.2279	0.5156	4.537479

Variable range	$0 \le x_1 \le 99$ $0 \le x_2 \le 99$ $10 \le x_3 \le 200$ $10 \le x_4 \le 200$	(19)

The results of solving this problem using different methods are illustrated in Table 15. As it can be seen, the proposed method has much better results than the other methods.

Table 12
results of different methods for solving the 200-dimensional version of multimodal test functions, with the population size of 40 and the 500 iterations.

F	LAPO				ABC			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-1.5279E + 04	-1.324E + 04	815.8747	2.23657	-3.4035E + 04	-3.294E + 04	734.7127	0.9800
F9	4.5475E-13	9.2776E-10	3.6949E-10	2.242719	1.8525E + 03	1.9515E + 03	45.5173	1.0466
F10	7.6647E-08	4.2081E-06	2.8691E-06	2.267153	19.5318	19.6937	0.0774	1.1107
F11	2.8019E-12	5.6723E-10	2.8363E-10	2.431825	2.1671E + 03	2.4488E + 03	111.3545	1.1884
F12	0.0995	0.0149	0.1178	5.632202	1.6032E + 09	2.0864E + 09	2.1788E + 08	2.7197
F13	12.7156	1.9412	15.7881	5.562711	3.3108E + 09	4.0474E + 09	3.1469E + 08	2.566407
F	DE				SLFA			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-5.4017E + 04	-4.977E + 04	2.0663E + 03	1.1282	-3.0403E + 04	-2.649E + 04	2.3060E + 03	5.7415
F9	731.6366	964.7553	114.3569	1.2444	880.8990	1.0968E + 03	98.0936	4.1798
F10	14.5240	16.1965	0.7866	1.2590	6.7096	7.8612	0.7120	4.5330
F11	235.9151	440.5763	122.8757	1.3054	19.9710	30.0968	5.4912	4.0709
F12	2.6572E + 06	2.6120E + 7	1.6150E + 07	2.8559	7.6205	11.2009	1.8581	9.4083
F13	3.9146E + 07	1.3105E + 08	7.2073E + 07	2.7561	236.0627	335.8872	108.7458	9.890213
F	ICA				PSO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-4.7091E + 04	-4.055E + 04	3.9585E + 03	1.60234	-2.9405E + 04	-2.4118E + 04	3.0779E + 03	1.307349
F9	367.8463	527.6486	78.9449	1.704648	652.9414	1.2740E + 03	154.8052	1.441813
F10	7.3976	9.1514	0.7618	1.767502	5.0312	8.4714	3.2569	1.454020
F11	17.7185	25.3039	3.1858	1.858962	22.3847	50.5061	19.7840	1.540186
F12	12.5943	18.2755	3.5619	3.554902	4.4008	10.9512	3.7138	3.138827
F13	849.3433	8.4779E + 3	9.2190E + 03	3.469389	171.5073	4.7033E + 03	6.5763E + 03	3.061198
F	ALO				GWO			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-3.6118E + 04	-3.6118E + 4	0	189.5486	-3.4022E + 04	-2.5747E + 4	8.6739E + 03	1.2000
F9	910.1357	1.0018E + 3	171.3214	189.5022	1.0633	17.9953	11.9642	1.2994
F10	12.9351	14.0165	1.1293	189.7388	6.0842E-06	8.4374E-06	1.5815E-06	1.3449
F11	173.2457	232.9929	88.5350	189.8364	1.3757E-09	4.0790E-09	3.2534E-09	1.4355
F12	9.7521E + 04	1.541E + 05	1.4595E + 05	191.5433	0.4078	0.4781	0.0417	3.0040
F13	5.5724E + 05	5.3616E + 6	3.4425E + 06	199.0284	15.7488	16.3107	0.4155	2.9299
F	CSA				Firefly			
	best	ave	std	time(s)	best	ave	std	Time(s)
F8	-3.0948E + 04	-2.9295E + 4	854.0313	1.191315	-4.4063E + 04	-3.6144E + 4	3.7575E + 03	4.155578
F9	1.2874E + 03	1.3985E + 3	49.5729	1.261941	482.2428	566.8940	49.7806	4.085035
F10	11.8777	13.9563	1.4820	1.323054	1.0800	1.7067	0.2831	4.144103
F11	157.5781	198.8925	22.9063	1.405092	0.2400	0.3097	0.0597	4.206908
F12	4.9955E + 04	4.7774E + 5	3.9327E + 05	2.914501	6.4607	7.8803	0.8738	5.606936
F13	2.5374E + 06	8.8288E + 6	3.4526E + 06	2.832080	189.4229	252.7130	36.7520	5.661512

Table 13
optimal results of Tension/compression spring design.

method	Optimal values for variables			Fmin
	d	D	N	
LAPO	0.0519038638	0.361890902	11.28885	**0.01265722**
GWO [7]	0.05169	0.356737	13.525410	0.012666
GSA [22]	0.050276	0.323680	11.244543	0.0127022
PSO [75]	0.051728	0.357644	10.6531340	0.0126747
GA [76]	0.051480	0.351661	11.632201	0.0127048
ES [73]	0.051989	0.363965	10.890522	0.0126810
HS [77]	0.051154	0.349871	12.076432	0.0126706
DE [78]	0.051609	0.354714	11.410831	0.0126702
Mathematical optimization [72]	0.053396	0.399180	9.1854000	0.0127303
Constraint correction [74]	0.050000	0.315900	14.250000	0.0128334

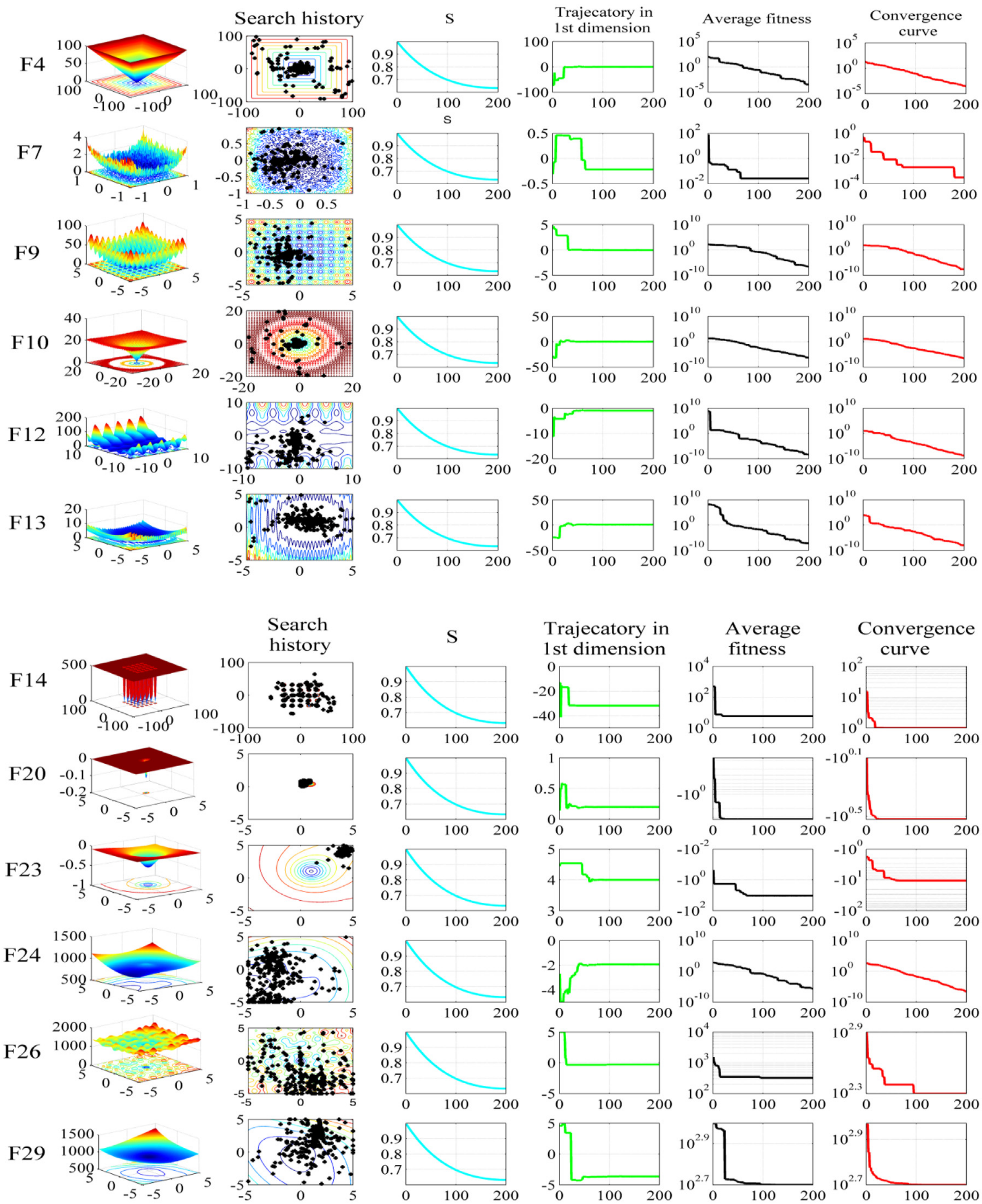

Fig. 12. Search history and trajectory of the first variable of the first solution.

Table 14
optimal results of Welded beam design.

method	Optimal values for variables				fmin
	h	l	t	b	
LAPO	0.205528028615	3.394773587424	9.076635213428	0.2055309791245	**1.71960676580**
GWO [7]	0.205676	3.478377	9.03681	0.205778	1.72624
GSA [7]	0.182129	3.856979	10.00000	0.202376	1.879952
GA [79,80]	0.2489	6.1730	8.1789	0.2533	2.4331
HS [81]	0.2442	6.2231	8.2915	0.2443	2.3807
RANDOM [82]	0.4575	4.7313	5.0853	0.6600	2.5307
SIMPLEX [82]	0.2792	5.6256	7.7512	0.2796	2.5307
APPROX [82]	0.2444	6.2189	8.2915	0.2444	2.3815

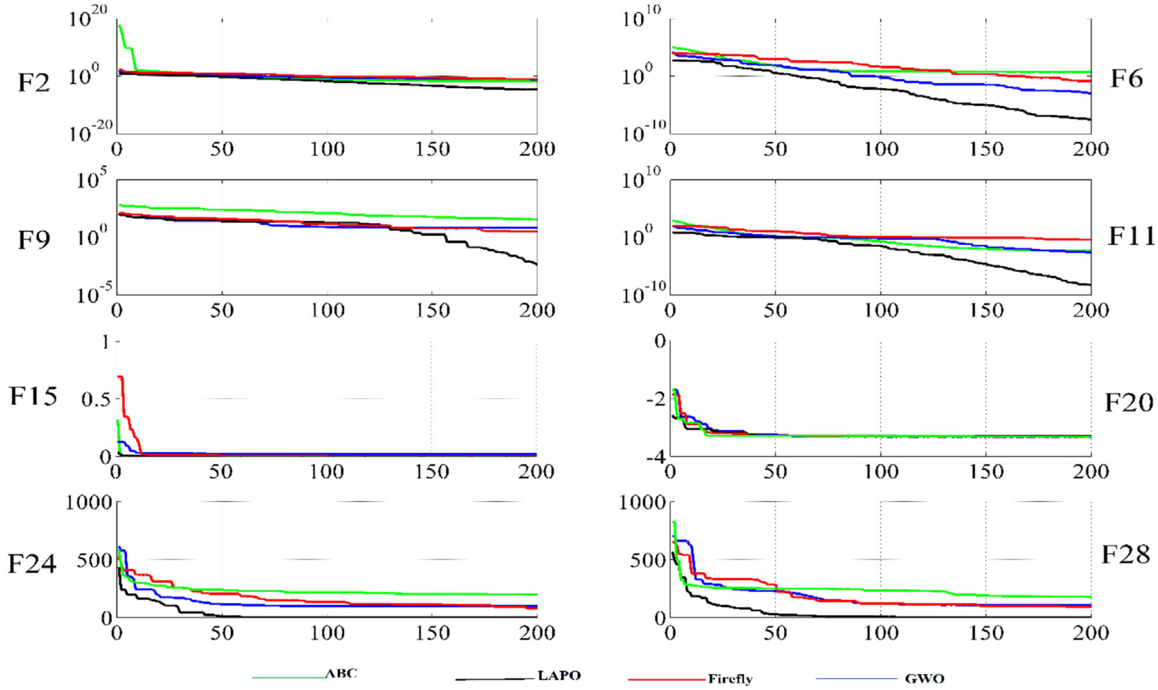

Fig. 13. Comparison of the convergence behavior of different methods.

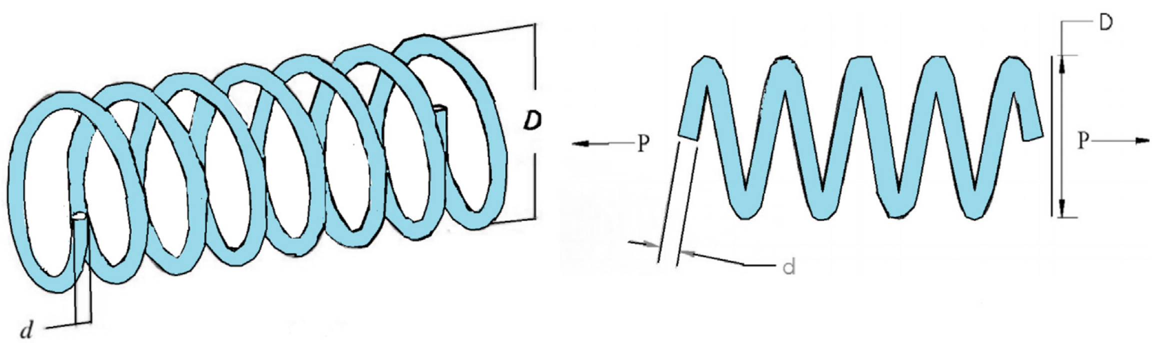

Fig. 14. The problem of Tension/compression spring design.

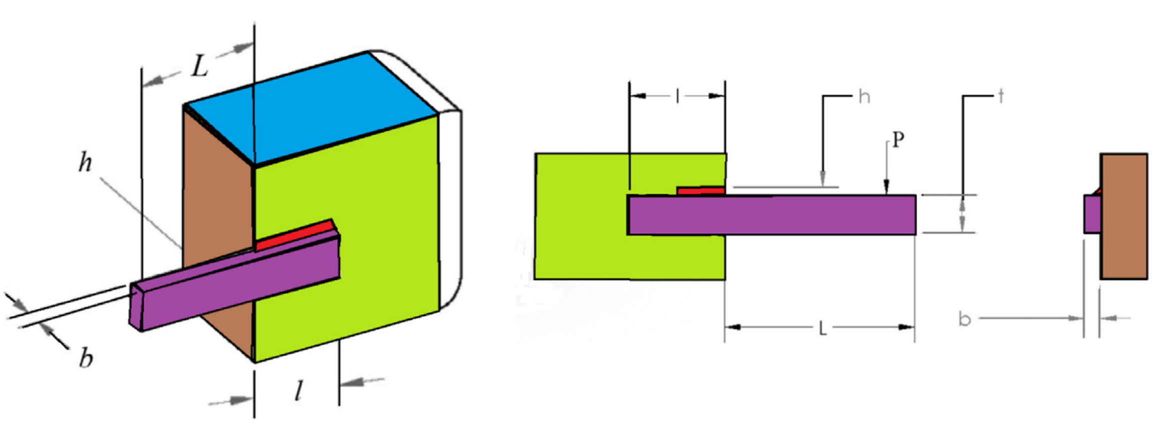

Fig. 15. Illustration of welded beam design problem.

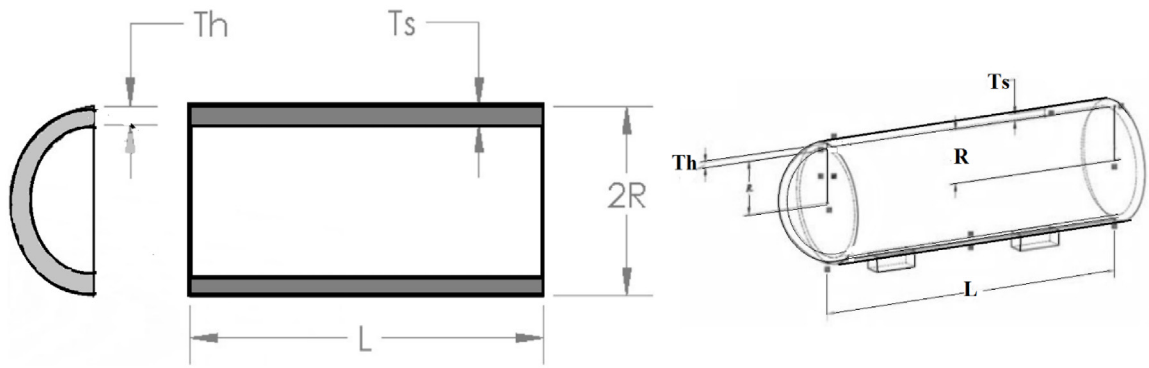

Fig. 16. The problem of Pressure vessel design.

Table 15
Optimal results of Pressure vessel design.

Method	Optimal values for variables				Fmin
	Ts	Th	R	L	
LAPO	0.786283665	0.39160673845	40.758736322	194.296457539	**5916.1935786**
GWO [7]	0.812500	0.434500	42.089181	176.758731	8538.8359
GSA [7]	1.125000	0.625000	55.9886598	84.4542025	6061.0777
PSO [75]	0.812500	0.437500	42.091266	176.746500	6288.7445
GA [73]	0.812500	0.434500	40.323900	200.000000	6059.7456
ES [83]	0.812500	0.437500	42.098087	176.640518	6410.3811
DE [78]	0.812500	0.437500	42.098411	176.637690	6059.7340
ACO [84]	0.812500	0.437500	42.103624	176.572656	6059.0888
Lagrangian Multiplier [85]	1.125000	0.625000	58.291000	43.6900000	7198.0428
Branch-bound [86]	1.125000	0.625000	47.700000	117.701000	8129.1036

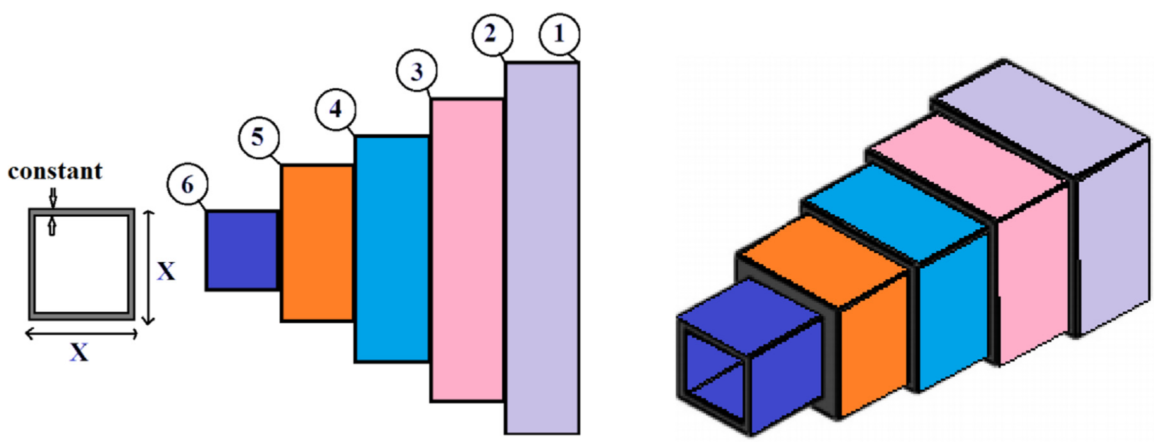

Fig. 17. Cantilever beam design problem.

3.6.4. Cantilever beam design problem

This problem is the to minimize weight of the beam in cantilever beam and it is explained in Fig. 17 [87]. This problem consists of 5 variables and also one vertical displacement constraint. The problem is formulated as follows:

Function	$f(x) = 0.6224(x_1+x_2+x_3+x_4+x_5)$	(20)
Inequality constraints	$g(x) = \frac{61}{x_1^3}+\frac{37}{x_2^3}+\frac{19}{x_3^3}+\frac{7}{x_4^3}+\frac{1}{x_5^3} \le 1$	(21)
Variable range	$0.01 \le x_1,x_2,x_3,x_4,x_5 \le 100$	(22)

The results of solving this problem with different methods are illustrated in Table 16. As it can be seen, the results of different methods are close to each other, however, the best results belong to the proposed LAPO method.

3.6.5. Gear train design problem

The problem is to find the optimal number of tooth for four gears of a train in order to minimize the gear ratio [56,86]. The problem is a discrete problem in which the variables are rounded to be discrete. The graphics of the problem is depicted in Fig. 18, and the mathematical formulation of this problem is as follows:

Function	$f(x) = \left(\frac{1}{6.931} - \frac{x_3 x_2}{x_1 x_4}\right)^2$	(23)
Variable range	$12 \le x_1,x_2,x_3,x_4 \le 60$	(24)
where	x_1, x_2, x_3, x_4 are discrete	(25)

By this problem, ability of the proposed method in solving the discrete problems is tested. The optimal results of this problem are obtained by different methods listed in Table 17. As it can be seen, the proposed LAPO is among the methods which achieve the best results.

3.6.6. Optimal power flow

Optimal Power Flow (OPF) is a high constraints electrical optimization problem in which the amount of generation of power plants (large generators) must be determined optimally [91,92].

Table 16
Optimal results of Cantilever beam design.

method	Optimal values for variables					fmin
	X1	X2	X3	X4	X5	
LAPO	6.01243634	5.314870556	4.4959135494	3.4993942765	2.151154796	**1.336521415**
ALO [33]	6.01812	5.31142	4.48836	3.49751	2.158329	1.33995
MMA [87]	6.0100	5.3000	4.4900	3.4900	2.1500	1.3400
GCA_I [87]	6.0100	5.30400	4.4900	3.4980	2.1500	1.3400
GCA_II [87]	6.0100	5.3000	4.4900	3.4900	2.1500	1.3400
SOS [88]	6.01878	5.30344	4.49587	3.49896	2.15564	1.33996
CS [26]	6.0089	5.3049	4.5023	3.5077	2.1504	1.33999

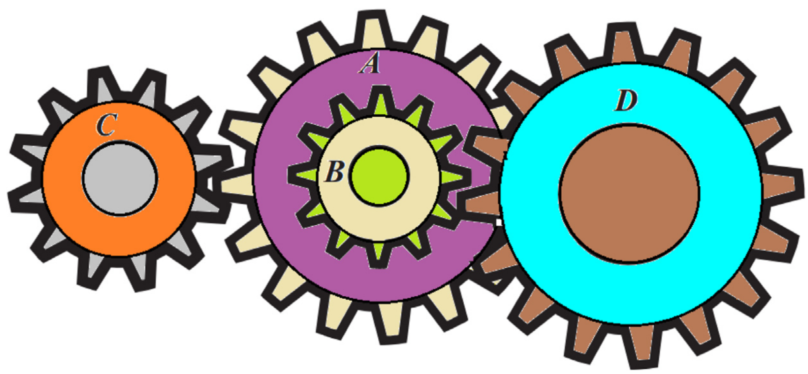

Fig. 18. The Gear train design problem.

Table 17
Optimal results of Gear train design.

method	Optimal values for variables				f_{min}
	n_A	n_B	n_C	n_D	
LAPO	49	16	19	43	**2.700857E-12**
ALO [33]	49	19	16	43	2.7009e012
CS [26]	43	16	19	49	2.7009e012
MBA [89]	43	16	19	49	2.7009e012
ISA [56]	N/A	N/A	N/A	N/A	2.7009e012
ABC [90]	19	16	44	49	2.78e11
GA [80]	33	14	17	50	1.362E-09
ALM [85]	33	15	13	41	2.1469E-08

The objective function of this optimization problem can be cost minimization, power loss minimization, emission reduction, etc. However, cost minimization is the most common objective function of this problem. Thus, the problem can be formulated as follow:

$$\begin{aligned} &\min \quad F=\sum_{i=1}^{N_g} C_i(x) \\ &\text{S.T} \quad g(x)=0 \\ &\qquad\;\; h(x)\le 0 \end{aligned} \tag{26}$$

Where x is the optimization variables (power generation), C_i is the cost of i^{th} power plant, g is the set of equality constraints and h is the set of inequality constraints.

In OPF there is only one equality constraint which refers to power balance in which the power generation must be equal to the sum of load demand and power losses as follow:

$$\sum_{i=1}^{N_g} P_i^G = P_{load}+P_{loss} \tag{27}$$

Where P^G is the power generation, N_g is the number of power plants (generators), P_{load} is load demand and P_{loss} is power loss. The load demand is specified, but the power losses must be calculated based on Eq. (28):

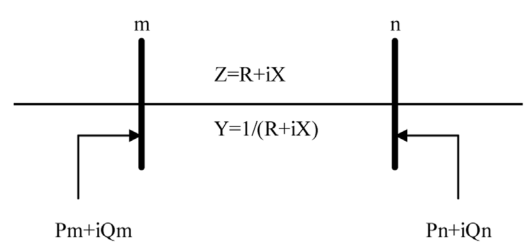

Fig. 19. a line between two buses in a power system.

$$P_{loss}=\sum_{j=1}^{Nl} R_j I_j^2 \tag{28}$$

Where R_j is the resistance of the line j, I_j is the electricity current of line j and N_l is the number of lines in a power system. This current can be calculated as follow:

$$I_j=\frac{V_m - V_n}{Z_j} \tag{29}$$

where the line j is placed between the buses m and n (as shown in Fig. 19), V refers to the voltage of a bus (a complex number), and Z_j is the impedance of line j (a complex number). In order to calculate the voltage of buses, the nonlinear equations (Eqs. (30), and (31)) denoting as power flow (load flow) equation must be calculated.

$$P_i=\sum_{j=1}^{N_b} |V_i||V_j||Y_{ij}|\cos(\delta_i-\delta_j-\theta_{ij}) \tag{30}$$

$$Q_i=\sum_{j=1}^{N_b} |V_i||V_j||Y_{ij}|\sin(\delta_i-\delta_j-\theta_{ij}) \tag{31}$$

Where P_i, and Q_i are active and reactive power injected to the bus i, $|V|$ and δ are the magnitude and the angle of voltage, $|Y_{ij}|$and θ_{ij}

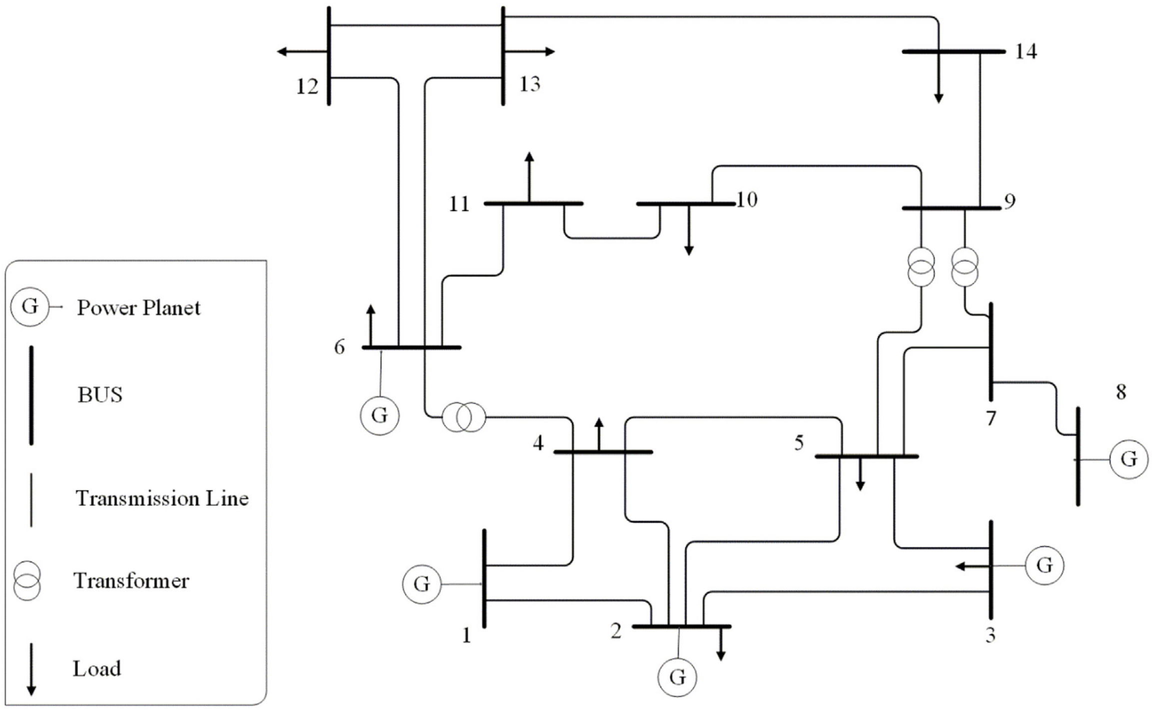

Fig. 20. Single line diagram of IEEE-14 bus test system.

Table 18
Optimal results of Gear train design.

method	Optimum variables								Optimum cost	time
	P_{G1}	P_{G2}	P_{G3}	P_{G4}	V_1	V_2	V_3	V_4		
LAPO	0.368	0.290	0.000	0.080	1.0385	1.0156	1.0599	1.0594	8079.9264	85.882
ABC	0.347	0.219	0.048	0.123	1.0379	1.0157	1.043	1.0067	8086.4	56.81
DE	0.367	0.287	0	0.088	1.0403	1.0167	1.0196	0.9500	8083.3881	75.01
PSO	0.367	0.283	0	0.09	1.0395	1.06	1.0256	0.95	8085.43	79.25
CSA	0.034	0	0.9	0.404	1.0401	1.0025	1.0097	0.99887	8772.4419	78.322
Firefly	0.350	0.338	0.090	0.108	1.0403	1.0428	1.0194	0.99184	8103.4	72.32
GA	0.368	0.298	5.1E-4	0.078	1.0384	1.0156	1.0599	1.05	8080.019	98.21
Pattern search	0.350	0.467	0.0	0.0	1.0417	1.0164	0.9968	1.031	8096.0007	14.11

are the magnitude and angle of admittance of line placed between buses *i* and *j*, respectively. These equations are mainly solved by a numerical method known as Newton-Raphosn.

The inequality constraints include the following:

Voltage magnitude

$$V_{min} \leq V \leq V_{max} \quad (32)$$

Voltage angle

$$\delta_{min} \leq \delta \leq \delta_{max} \quad (33)$$

Line Thermal limit

$$I \leq Limit \quad (34)$$

Generation limitaion

$$P^G_{min} \leq P^G \leq P^G_{max} \quad (35)$$

Where V_{max} and V_{min} are the maximum and minimum allowable voltage magnitude in each bus, δ_{max} and δ_{min} are the maximum and minimum allowable voltage angle in each bus, *Limit* is the thermal limit of a line, P^G,P^G_{max}, andP^G_{min} are the active power generation, maximum allowable and minimum allowable power generation of a power plant. For each solution generated by the optimization algorithm in the predefined range (i.e., the fourth inequality constraint), first of all, the load flow equation is run and the power loss is calculated. Then after, the equality constraint, and also inequality constraints are checked. If the constraints are satisfied, the solution is accepted and updated by the optimization algorithm, otherwise, it is discarded.

OPF is performed for a benchmark test function known as IEEE 14-bus test system which the single line diagram is depicted in Fig. 20. The optimal results of different methods including proposed LAPO are listed in Table 18. As it can be seen, the proposed method has the best answer among different methods. In other words, the proposed LAPO reaches the high quality results in solving the high constraint problem. These results show the superiority of the proposed method in solving the high constraints engineering optimization problem.

4. Conclusion

In this paper, a novel nature-inspired meta-heuristic optimization algorithm known as Lightning Attachment Procedure Optimization (LAPO) is proposed. The proposed population-based approach is inspired by the physical phenomena of lightning attachment procedure. The procedure includes air breakdown, downward leader movement, upward leader inception and propagation, and final jump. The proposed algorithm is free from any parameter tuning which is performed in two main phases as downward leader movement and upward leader propagation. The method is benchmarked by means of 29 test functions which are classified in 4 groups. Using the first group (i.e., unimodal test function), and the second and third groups (i.e., multimodal and fixed dimensional multimodal) the characteristics of exploitation and exploration of the method are tested, respectively. The results reveal that the proposed LAPO method has excellence performance not only in local search but also in global search. The results of LAPO for the fourth group of test functions (i.e., composite) demonstrate that LAPO well balances exploitation and exploration. Moreover, the proposed method is evaluated for large scale optimization problems (200-dimensional version of unimodal and multi modal test functions) in two cases including 1-large number of iterations and large population size 2-small number of iteration and small population size. The proposed LAPO has superior performance in both cases. Results of the proposed method are compared to 9 high quality meta-heuristic optimization algorithms from different points of view in most of the test functions. The proposed method outperforms other methods in some test functions and has very competitive results for the rest of the test functions. Small standard deviations of the results of the 30 different trials prove robustness of the proposed method. Although the proposed method is not the fastest one among the methods to which the results are compared, it has an almost low CPU time consumption. The convergence behavior of the proposed LAPO method is also evaluated. From the figures and the comparisons it can be comprehended that the proposed method reaches the best overall answer in the early iterations.

In addition to 29 test functions, five classic mechanical and a high constraints electrical engineering problems are also employed to evaluate the proposed method for solving the challenging, high constraint, and discrete optimization problems with unknown search space. The results of the proposed method are compared to those of literature. From the results it can be concluded that the proposed method has high quality results even in large constraint discrete optimization problems.

To sum up, based on this comprehensive comparative study, the merits of the purposed method could be listed as follows:

- The purposed method is free from any parameter tuning.
- The procedure of downward leader movement and upward leader propagation in each step (two times solutions updating in each iteration) enhance the characteristic of exploration.
- Two phase solution updating in each iteration increases the balance between exploration and exploitation
- The procedure of upward leader propagation in each step causes the solutions to move away from worse answers and get close to the better answers.
- The parameter S which simulates the charge distribution of lightning channel accelerates converging to the local optimum point.
- Replacing the worth solution of each iteration with average solution leads to faster convergence.
- The branch fading procedure helps the method to discard the worse answer.
- The method is able to solve challenging, high constraint, discrete optimization problem with unknown search space

References

[1] C. Blum, J. Puchinger, G.R. Raidl, A. Roli, Hybrid metaheuristics in combinatorial optimization: a survey, Appl. Soft Comput. 11 (2011) 4135–4151.

[2] I. BoussaïD, J. Lepagnot, P. Siarry, A survey on optimization metaheuristics, Inf. Sci. (N.Y.) 237 (2013) 82–117.

[3] A. Gogna, A. Tayal, Metaheuristics: review and application, J. Exp. Theor. Artif. Intell. 25 (2013) 503–526.

[4] K. Miettinen, P. Preface By-Neittaanmaki, Evolutionary Algorithms in Engineering and Computer Science: Recent Advances in Genetic Algorithms, Evolution Strategies, Evolutionary Programming, GE, John Wiley & Sons, Inc., 1999.

[5] P. Venkataraman, Applied Optimization with MATLAB Programming, John Wiley & Sons, 2009.

[6] R. Statnikov, J.B. Matusov, Multicriteria Optimization and Engineering, Springer Science & Business Media, 2012.

[7] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61.

[8] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simmulated annealing, Science 220 (80) (1983) 671–680.

[9] L. Davis, Handbook of Genetic Algorithms, 1991.

[10] J. Knowles, D. Corne, The pareto archived evolution strategy: a new baseline algorithm for pareto multiobjective optimisation, in: Evol Comput. 1999. CEC 99. Proc. 1999 Congr., IEEE, 1999.

[11] J.R. Koza, Genetic Programming: on the Programming of Computers by Means of Natural Selection, MIT press, 1992.

[12] D. Simon, Biogeography-based optimization, IEEE Trans. Evol. Comput. 12 (2008) 702–713.

[13] A. Kaveh, S. Talatahari, A novel heuristic optimization method: charged system search, Acta Mech. 213 (2010) 267–289.

[14] R.A. Formato, Central force optimization: a new metaheuristic with applications in applied electromagnetics, Prog. Electromagn. Res. 77 (2007) 425–491.

[15] B. Alatas, ACROA: artificial chemical reaction optimization algorithm for global optimization, Expert Syst. Appl. 38 (2011) 13170–13180.

[16] A. Hatamlou, Black hole: a new heuristic optimization approach for data clustering, Inf. Sci. (N.Y.) 222 (2013) 175–184.

[17] A. Kaveh, M. Khayatazad, A new meta-heuristic method: ray optimization, Comput. Struct. 112 (2012) 283–294.

[18] H. Du, X. Wu, J. Zhuang, Small-world optimization algorithm for function optimization, in: Int Conf. Nat. Comput., Springer, 2006, pp. 264–273.

[19] H. Shah-Hosseini, Principal components analysis by the galaxy-based search algorithm: a novel metaheuristic for continuous optimisation, Int. J. Comput. Sci. Eng. 6 (2011) 132–140.

[20] A. Kaveh, Water evaporation optimization algorithm, in: Adv Metaheuristic Algorithms Optim. Des. Struct., Springer, 2017, pp. 489–509.

[21] S. Mirjalili, S.M. Mirjalili, A. Hatamlou, Multi-verse optimizer: a nature-inspired algorithm for global optimization, Neural Comput. Appl. 27 (2016) 495–513.

[22] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, GSA: a gravitational search algorithm, Inf. Sci. (Ny). 179 (2009) 2232–2248.

[23] J. Kennedy, Particle swarm optimization, in: Encycl Mach. Learn., Springer, 2011, pp. 760–766.

[24] S. Naka, T. Genji, T. Yura, Y. Fukuyama, Hybrid particle swarm optimization based distribution state estimation using constriction factor approach, Proc Int. Conf. SCIS ISIS (2002) 1083–1088.

[25] C. Yang, X. Tu, J. Chen, Algorithm of marriage in honey bees optimization based on the wolf pack search, in: Intell. Pervasive Comput. 2007. IPC. 2007 Int. Conf., IEEE, 2007, pp. 462–467.

[26] A.H. Gandomi, X.-S. Yang, A.H. Alavi, Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems, Eng. Comput. 29 (2013) 17–35.

[27] X.-S. Yang, Firefly algorithms for multimodal optimization, in: Int Symp. Stoch. Algorithms, Springer, 2009, pp. 169–178.

[28] A. Askarzadeh, Bird mating optimizer: an optimization algorithm inspired by bird mating strategies, Commun. Nonlinear Sci. Numer. Simul. 19 (2014) 1213–1228.

[29] A. Mucherino, O. Seref, Monkey search: a novel metaheuristic search for global optimization, in: Data Mining Syst. Anal. Optim. Biomed., AIP Publishing, 2007, pp. 162–173.

[30] S. Salcedo-Sanz, J. Del Ser, I. Landa-Torres, S. Gil-López, J.A. Portilla-Figueras, The coral reefs optimization algorithm: a novel metaheuristic for efficiently solving optimization problems, Sci. World J. 2014 (2014).

[31] S. Salcedo-Sanz, A. Pastor-Sánchez, D. Gallo-Marazuela, A. Portilla-Figueras, A novel coral reefs optimization algorithm for multi-objective problems, in: Int Conf. Intell. Data Eng. Autom. Learn., Springer, 2013, pp. 326–333.

[32] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Glob. Optim. 39 (2007) 459–471.

[33] S. Mirjalili, The ant lion optimizer, Adv. Eng. Softw. 83 (2015) 80–98.

[34] S. Mirjalili, Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm, Knowledge-Based Syst. 89 (2015) 228–249.

[35] S. Mirjalili, A. Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67.

[36] S. Mirjalili, Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems, Neural Comput. Appl. 27 (2016) 1053–1073.

[37] A. Kaveh, N. Farhoudi, A new optimization method: dolphin echolocation, Adv. Eng. Softw. 59 (2013) 53–70.

[38] A.H. Gandomi, A.H. Alavi, Krill herd: a new bio-inspired optimization algorithm, Commun. Nonlinear Sci. Numer. Simul. 17 (2012) 4831–4845.

[39] R.V. Rao, V.J. Savsani, D.P. Vakharia, Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems, Comput. Des. 43 (2011) 303–315.

[40] Z.W. Geem, J.H. Kim, G.V. Loganathan, A new heuristic optimization algorithm: harmony search, Simulation 76 (2001) 60–68.

[41] F. Glover, Tabu search-part I, ORSA J. Comput. 1 (1989) 190–206.
[42] F. Glover, M. Laguna, Tabu Search∗, Springer, 2013.
[43] F. Glover, Tabu search—part II, ORSA J. Comput. 2 (1990) 4–32.
[44] F. Glover, Tabu search: a tutorial, Interfaces (Providence) 20 (1990) 74–94.
[45] S. He, Q.H. Wu, J.R. Saunders, Group search optimizer: an optimization algorithm inspired by animal searching behavior, IEEE Trans. Evol. Comput. 13 (2009) 973–990.
[46] S. Talatahari, B.F. Azar, R. Sheikholeslami, A.H. Gandomi, Imperialist competitive algorithm combined with chaos for global optimization, Commun. Nonlinear Sci. Numer. Simul. 17 (2012) 1312–1319.
[47] A.H. Kashan, League Championship Algorithm (LCA): An algorithm for global optimization inspired by sport championships, Appl. Soft Comput. 16 (2014) 171–200.
[48] A.H. Kashan, An efficient algorithm for constrained global optimization and application to mechanical engineering design: league championship algorithm (LCA), Comput. Des. 43 (2011) 1769–1792.
[49] Y. Tan, Hybrid Fireworks Algorithms, In: Firew Algorithm, Springer, 2015, pp. 151–161.
[50] Y. Tan, Discrete Firework Algorithm for Combinatorial Optimization Problem, In: Firew Algorithm, Springer, 2015, pp. 209–226.
[51] Y. Tan, Y. Zhu, Fireworks algorithm for optimization, in: Int Conf. Swarm Intell., Springer, 2010, pp. 355–364.
[52] A. Kaveh, V.R. Mahdavi, Colliding bodies optimization method for optimum design of truss structures with continuous variables, Adv. Eng. Softw. 70 (2014) 1–12.
[53] A. Kaveh, V.R. Mahdavi, Colliding bodies optimization method for optimum discrete design of truss structures, Comput. Struct. 139 (2014) 43–53.
[54] A. Kaveh, V.R. Mahdavi, Colliding bodies optimization: a novel meta-heuristic method, Comput. Struct. 139 (2014) 18–27.
[55] A. Kaveh, Tug of war optimization, in: Adv Metaheuristic Algorithms Optim. Des. Struct., Springer, 2017, pp. 451–487.
[56] A.H. Gandomi, Interior search algorithm (ISA): a novel approach for global optimization, ISA Trans. 53 (2014) 1168–1183.
[57] E. Alba, B. Dorronsoro, The exploration/exploitation tradeoff in dynamic cellular genetic algorithms, IEEE Trans. Evol. Comput. 9 (2005) 126–142.
[58] A.E. Eiben, C.A. Schippers, On evolutionary exploration and exploitation, Fundam. Informaticae 35 (1998) 35–50.
[59] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evol. Comput. 1 (1997) 67–82.
[60] H. Shareef, A.A. Ibrahim, A.H. Mutlag, Lightning search algorithm, Appl. Soft Comput. 36 (2015) 315–333.
[61] A.R. Hileman, Insulation Coordination for Power Systems, CRC Press, 1999.
[62] A. Rahiminejad, B. Vahidi, LPM-Based Shielding performance Analysis of High-Voltage Substations against Direct Lightning Strokes, IEEE Trans. Power Deliv. (n.d.).
[63] M.R.B. Tavakoli, B. Vahidi, Transmission-lines shielding failure-rate calculation by means of 3-D leader progression models, IEEE Trans. Power Deliv. 26 (2011) 507–516.
[64] M. Yahyaabadi, B. Vahidi, M.R.B. Tavakoli, Estimation of shielding failure number of different configurations of double-circuit transmission lines using leader progression analysis model, Electr. Eng. (Archiv Fur Elektrotechnik) 92 (2010) 79–85.
[65] W. Shi, Q. Li, L. Zhang, A stepped leader model for lightning including charge distribution in branched channels, J. Appl. Phys. 116 (2014) 103303.
[66] M. Becerra, V. Cooray, A simplified physical model to determine the lightning upward connecting leader inception, IEEE Trans. Power Deliv. 21 (2006) 897–908.
[67] K. Price, R.M. Storn, J.A. Lampinen, Differential Evolution: a Practical Approach to Global Optimization, Springer Science & Business Media, 2006.
[68] M.M. Eusuff, K.E. Lansey, Optimization of water distribution network design using the shuffled frog leaping algorithm, J. Water Resour. Plan. Manage. 129 (2003) 210–225.
[69] X. Hu, R. Eberhart, Solving constrained nonlinear optimization problems with particle swarm optimization, in: Proc Sixth World Multiconference Syst. Cybern. Informatics, Citeseer, 2002, pp. 203–206.
[70] S. Mirjalili, ALO MATLAB Code, 2015, https://www.mathworks.com/matlabcentral/fileexchange/49920-ant-lion-optimizer?alo- (Accessed May 21, 2017).
[71] H. Shareef, LSA MATLAB Code, 2015, https://www.mathworks.com/matlabcentral/fileexchange/54181-lightning-search-algorithm-lsa- (Accessed May 21, 2017).
[72] A.D. Belegundu, J.S. Arora, A study of mathematical programming methods for structural optimization. Part I: Theory, Int. J. Numer. Methods Eng. 21 (1985) 1583–1599.
[73] C.A.C. Coello, E.M. Montes, Constraint-handling in genetic algorithms through the use of dominance-based tournament selection, Adv. Eng. Inf. 16 (2002) 193–203.
[74] J. Arora, Introduction to Optimum Design, Academic Press, 2004.
[75] Q. He, L. Wang, An effective co-evolutionary particle swarm optimization for constrained engineering design problems, Eng. Appl. Artif. Intell. 20 (2007) 89–99.
[76] C.A.C. Coello, Use of a self-adaptive penalty approach for engineering optimization problems, Comput. Ind. 41 (2000) 113–127.
[77] M. Mahdavi, M. Fesanghary, E. Damangir, An improved harmony search algorithm for solving optimization problems, Appl. Math. Comput. 188 (2007) 1567–1579.
[78] F. Huang, L. Wang, Q. He, An effective co-evolutionary differential evolution for constrained optimization, Appl. Math. Comput. 186 (2007) 340–356.
[79] K. Deb, Optimal design of a welded beam via genetic algorithms, AIAA J. 29 (1991) 2013–2015.
[80] K. Deb, M. Goyal, A combined genetic adaptive search (GeneAS) for engineering design, Comput. Sci. Inf. 26 (1996) 30–45.
[81] K.S. Lee, Z.W. Geem, A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice, Comput. Methods Appl. Mech. Eng. 194 (2005) 3902–3933.
[82] K.M. Ragsdell, D.T. Phillips, Optimal design of a class of welded structures using geometric programming, J. Eng. Ind. 98 (1976) 1021–1025.
[83] 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 (2008) 443–473.
[84] A. Kaveh, S. Talatahari, An improved ant colony optimization for constrained engineering design problems, Eng. Comput. 27 (2010) 155–182.
[85] B.K. Kannan, S.N. Kramer, An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design, J. Mech. Des. 116 (1994) 405–411.
[86] E. Sandgren, Nonlinear integer and discrete programming in mechanical design optimization, J. Mech. Des. 112 (1990) 223–229.
[87] H. Chickermane, H.C. Gea, Structural optimization using a new local approximation method, Int. J. Numer. Methods Eng. 39 (1996) 829–846.
[88] M.-Y. Cheng, D. Prayogo, Symbiotic organisms search: a new metaheuristic optimization algorithm, Comput. Struct. 139 (2014) 98–112.
[89] A. Sadollah, A. Bahreininejad, H. Eskandar, M. Hamdi, Mine blast algorithm: a new population based algorithm for solving constrained engineering optimization problems, Appl. Soft Comput. 13 (2013) 2592–2612.
[90] T.K. Sharma, M. Pant, V.P. Singh, Improved Local Search in Artificial Bee Colony Using Golden Section Search, ArXiv Prepr. ArXiv1210.6128, 2012.
[91] H.W. Dommel, W.F. Tinney, Optimal power flow solutions, IEEE trans, Power Appar. Syst. (1968) 1866–1876.
[92] M.A. Abido, Optimal power flow using particle swarm optimization, Int. J. Electr. Power Energy Syst. 24 (2002) 563–571.