The Journal of Supercomputing (2023) 79:3511–3560
https://doi.org/10.1007/s11227-022-04755-2

Ladybug Beetle Optimization algorithm: application for real-world problems

Saadat Safiri[1] · **Amirhossein Nikoofard**[1]

Accepted: 8 August 2022 / Published online: 6 September 2022

Abstract

In this paper, a novel optimization algorithm is proposed, called the Ladybug Beetle Optimization (LBO) algorithm, which is inspired by the behavior of ladybugs in nature when they search for a warm place in winter. The new proposed algorithm consists of three main parts: (1) determine the heat value in the position of each ladybug, (2) update the position of ladybugs, and (3) ignore the annihilated ladybug(s). The main innovations of LBO are related to both updating the position of the population, which is done in two separate ways, and ignoring the worst members, which leads to an increase in the search speed. Also, LBO algorithm is performed to optimize 78 well-known benchmark functions. The proposed algorithm has reached the optimal values of 73.3% of the benchmark functions and is the only algorithm that achieved the best solution of 20.5% of them. These results prove that LBO is substantially the best algorithm among other well-known optimization methods. In addition, two fundamentally different real-world optimization problems include the Economic-Environmental Dispatch Problem (EEDP) as an engineering problem and the Covid-19 pandemic modeling problem as an estimation and forecasting problem. The EEDP results illustrate that the proposed algorithm has obtained the best values in either the cost of production or the emission or even both, and the use of LBO for Covid-19 pandemic modeling problem leads to the least error compared to others.

Keywords Ladybug Beetle Optimization algorithm · Search for a warm position · Annihilated ladybugs · Economic-environmental dispatch problem · Covid-19 · Pandemic prediction

List of symbols

LBO parameters

i	Member of the population that is being updated
j	Member of the population that is used to update the ith member

✉ Amirhossein Nikoofard
a.nikoofard@kntu.ac.ir

1 Faculty of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran

k	Iteration
$k_{\max}$	Maximum iteration that used to terminate the optimization algorithm
t	Index of summation
rand	A uniformly distributed random number between 0 and 1
$N(0)$	Number of the initial population
$N(k)$	Number of the population in the kth iteration
$N_{\min}$	Minimum number of the population during the algorithm process
NFE	Number of function evaluation
$\text{NFE}_{\max}$	The maximum number of function evaluation, which used to terminate the optimization algorithm
$x_i(k), x_j(k)$	Position of the ith and jth members in the search space
D	Dimensions of the decision vector
$f\left(x_i(k)\right)$	The value of the cost function for the ith member in the kth iteration
f_{worst}	The worst value of the cost function up to the current iteration during the algorithm process
f_{opt}	The optimal global value for the cost function
C_i	The ratio of the ith member cost to total members cost
$\overrightarrow{r_1}, \overrightarrow{r_2}$, and $\overrightarrow{r_3}$	Three vectors that are used to update the ith member of the population
P	Generated vector for the population in Roulette-wheel selection method
β	Pressure coefficient in Roulette-wheel selection method

Economic-environmental dispatch problem parameters

i	The power unit in the grid
N_{G}	The number of power unit in the grid
P_i	The value of power generation in the ith unit
$P_i^{\min}, P_i^{\max}$	The lower and upper bound of power generation in the ith unit
$F_{P_i}\left(P_i\right)$	The value of generation cost for the ith unit to generate P_i MW of power
a_i, b_i, c_i, g_i, and h_i	The constant parameters for calculating generation cost in the ith unit
$F_{\text{E}_i}\left(P_i\right)$	The value of emission for the ith unit to generate P_i MW of power
$\alpha_i, \beta_i, \gamma_i, \eta_i$, and δ_i	The constant parameters for calculating emission in the ith unit
F	The total cost function
p	Penalty coefficient
D	Power demand in MW
L_{P}	The value of power losses during the transmission
B	The constant matrix for calculating power losses

Covid-19 pandemic modeling parameters

$S(t)$	Susceptible individuals
$I(t)$	Infected individuals
$D(t)$	Diagnosed individuals
$A(t)$	Ailing individuals
$R(t)$	Recognized individuals
$T(t)$	Threatened individuals
$H(t)$	Recovered individuals
$E(t)$	Death cases
α	Transmission rate from the infected to the susceptible individual
β	Transmission rate from the diagnosed to the susceptible individual
γ	Transmission rate from the recognized to the susceptible individual
δ	Transmission rate from the ailing to the susceptible individual
ε	Detection rate of the individual with no symptoms
θ	Detection rate of the individual with symptoms
ζ	The probability that the infected individual knows that they are infected
η	The probability that the infected individual does not know that they are infected
μ	The probability of developing life-threatening symptoms
ν	The probability of developing life-threatening symptoms for a detected case
τ	Death rate
λ	The recovery rate
κ	The recovery rate
ξ	The recovery rate
ρ	The recovery rate
σ	The recovery rate

1 Introduction

Nowadays, the optimization algorithm field is one of the most important research fields for all researchers, especially for engineers [1]. Research on metaheuristic optimization algorithms is a branch of the optimization field that has also attracted a lot of attention. Many metaheuristic algorithms have been proposed so far. However, since optimization problems have a wide range and their scope is increasing every day, more research in this area is necessary to solve new optimization problems [2].

In general, the metaheuristic algorithms are divided into three classes consist of: (1) the improvement of the existing algorithms, such as Improved Whale Optimization Algorithm (IWOA) [3–5], improvement of ant colony optimization algorithm (ICMPACO) [6], and modified henry gas solubility optimization (HGSWC) [7], (2) the combination of two or more existing optimization algorithms, such as combined with hybrid particle swarm optimization and grey wolf optimization

(ELM-PSOGWO) [8], Differential Evolutionary and Particle Swarm Optimization (DEEPSO) algorithm [9], hybrid Firefly and Self-Regulating Particle Swarm Optimization (FSRPSO) [10], and Hybrid Particle swarm and Ant colony optimization (HAP) [11], and (3) introduction of a new optimization algorithm, such as introduced artificial Jellyfish Search (JA) [12], Political Optimizer (PO) [13], Mayfly Algorithm (MA) [14], and Shuffled Shepherd Optimization Algorithm (SSOA) [15].

Almost all of the metaheuristic algorithms are inspired by natural processes. Hence, they can be divided into the following four categories:

(1) *Swarm algorithms*: The collective and coordinated behavior of creatures in nature inspires one of the most important categories of metaheuristic algorithms. Creatures that live collectively in nature have optimal behavior with others to achieve their purpose, and modeling their behavior leads to important optimization algorithms. Particle Swarm Optimization (PSO) [16], Cuckoo Optimization Algorithm (COA) [17], Whale Optimization Algorithm (WOA) [18], combined Particle Filter and Particle Swarm Optimization (PF-PSO) [19], Tuna Swarm Optimization (TSO) [20], and Rock Hyraxes Swarm Optimization (RHSO) [21] are the most popular optimization algorithms in this category.
(2) *Evolutionary algorithms*: Many biological processes have evolved during the time, and several metaheuristic algorithms are proposed by modeling them. The most important algorithm fallen into this category is the Genetic Algorithm (GA) [22]. Other algorithms such as Differential Evolution (DE) [23] and Snake Optimizer (SO) [24] are also in this category.
(3) *Physical algorithms*: Several practical metaheuristic optimization algorithms are based on physical processes. For example, Water Evaporation Optimization (WEO) [25], Water Cycle Algorithm (WCA) [26], Equilibrium Optimizer (EO) [27], and Flow Regime Algorithm (FRA) [28] are inspired by physical processes.
(4) *Human algorithms*: The behavior of humans is the patterns to present new metaheuristic optimization algorithm. Humans are known as the most intelligent creatures on Earth. Therefore, their behaviors are intelligent against various problems and can inspire optimization algorithms. These algorithms include Teaching–Learning-Based Optimization (TLBO) [29], Heap-Based Optimizer (HBO) [30], Forensic-Based Investigation (FBI) [31], and Soccer League Competition (SLC) algorithm [32].

In addition to the above optimization algorithms, there are many metaheuristic optimization algorithms which have been recently introduced. Several of them are given in Table 1.

In this paper, a novel optimization algorithm inspired by the behaviors of ladybugs is proposed. Ladybugs, like most beetles, experience collective life. Normal ladybugs live for a year from mid-spring or early summer and continues until next year. In early winter, ladybugs are used to find warm positions according to a coordinated behavior and are able to find a warm place by the collective movement. Some of them may get lost and annihilate while searching for the hottest place. Consequently, their population is reduced compared to the initial ones. Moreover,

Table 1 Several metaheuristic optimization algorithms have been proposed recently

Year	Algorithm name	Inspired by
2021	seagull optimization algorithm (SOA) [33]	Seagull behavior
2021	Archimedes optimization algorithm (AOA) [34]	Interesting law of physics Archimedes' Principle
2020	Chimp optimization algorithm (ChOA) [35]	The individual intelligence and sexual motivation of chimps in their group hunting
2020	Search and rescue optimization algorithm (SAR) [36]	The explorations behavior of humans during search and rescue operations
2020	Dynamic group-based optimization algorithm (DGCO) [37]	The cooperative behavior adopted by swarm individuals to achieve their global goals
2020	Equilibrium optimizer (EO) [27]	Control volume mass balance models
2019	Pigeon-inspired optimization (PIO) [38]	Pigeon behavior
2019	Emperor penguins colony (EPC) [39]	Emperor penguins
2019	Harris hawks optimization (HHO) [40]	Cooperative behavior
2019	Heterogeneous pigeon-inspired optimization (HPIO) [41]	Homing behavior of pigeons
2019	The sailfish optimization (TSO) [42]	Hunting sailfish
2018	Farmland fertility (FF) [43]	Farmland fertility

the behavior of each ladybug is influenced by the other ladybugs, especially ladybugs who have discovered places with higher temperatures. The proposed algorithm has some pros and cons that are expressed below:

(1) Using ladybugs' behavior helps the algorithm to utilize new undiscovered potential to update the position of the population in each iteration, which leads to a powerful metaheuristic algorithm.
(2) At first, the optimizer is completely blind to the search space. Thus, it needs a bigger population to discover this space. The search space is known by increasing the number of iterations of the algorithm. Therefore, a smaller population at higher iterations would be sufficient for the algorithm to reach the optimal point. Reducing the population size leads to an increase in the speed of the algorithm, which is considered in LBO.
(3) In other algorithms, for instance in PSO algorithm, the best positions that a member of the population and the whole population have experienced are involved in updating the member's position, whereas other members' positions in LBO have the chance to use in updating a member. This feature of LBO helps to escape local minimums and increases the speed of the algorithm.
(4) On the other hand, reducing the population size would not be appropriate in some cases, especially before discovering all of the search space by the algorithm, which leads to trapping in local minimum or decreasing the speed of the algorithm. However, according to the simulation results, reducing the

population size is useful in most cases, and in the others, the rate of reduction needs to reset.

Accordingly, all expressed behaviors of ladybugs are included in the optimizer modeling, and the proposed optimization algorithm is presented. Therefore, in this paper, both the physical behavior of ladybugs in nature and the mathematical modeling extracted from their behavior are presented. Finally, Ladybug Beetle Optimization (LBO) algorithm is introduced. To evaluate the performance of the proposed optimization algorithm, 78 benchmark functions are optimized by LBO, and the results are compared with seven high-performance optimization algorithms.

Most of the metaheuristic algorithms are based on a similar structure. However, most of them are basically different in updating their solutions. In the proposed optimization algorithm, the population is updated based on a unique strategy. This strategy of the updating the population uses the location of two members of the population. First, a random member of the population (with a bigger chance for better members). Second, another member of the population with a cost function close to the first selected member. Besides, the number of the population decreases during the operation of the algorithm, which means the greater number of the population is employed at first when the optimization algorithm does not have any sense of the search space. As the search space becomes more specific and is limited to smaller spaces, the number of the population starts to reduce in order to increase the speed of the algorithm.

Metaheuristic algorithms have a wide range of applications in engineering problems and are a powerful tool to solve these problems by turning them into optimization problems [44–47]. Therefore, given that the ultimate goal of optimization algorithms is to use them in practice, it is essential that a powerful optimization algorithm can show appropriate results in a variety of real-world problems. In this regard, LBO algorithm is employed to solve both Economic-Environmental Dispatch Problem (EEDP) and the Covid-19 pandemic modeling problem. The performance of the proposed algorithm is evaluated by applying the proposed algorithm to the several benchmark functions and two mentioned real-world problems and comparing the results with the other algorithms.

Specifying the power generation amount of power plants in the grid is a main task of power engineers to achieve the minimum of the generated power cost and emission. Consequently, one of the most challenging optimization problems in the engineering field has been defined. Numerous methods have been proposed to solve EEDP so far, making it a popular optimization problem. In this paper, the LBO algorithm is used to solve this problem, and the results are compared with some other methods. In this regard, three different grids, including IEEE three, ten, and forty-bus grids, are studied.

On the other hand, coronavirus pandemic has been rapidly spreading to 221 countries that are involved in this virus. Moreover, 242 million and 4.93 million individuals have been confirmed and died worldwide, respectively [48]. An appropriate model of the covid-19 outbreak would substantially help the authorities to make deliberate decisions. Moreover, accurate prediction of this pandemic depends on accurate modeling. Numerous studies have been conducted in this regard to model the Covid-19 pandemic, which is divided into several categories. First, some proposed models of Covid-19 utilize artificial intelligence and machine learning algorithms. In [49], some algorithms, including Bayesian regression neural network, cubist regression, k-nearest neighbors, quantile random forest, and support vector regression, are employed independently and cooperated with the variational mode decomposition and forecasting is done for one, three, and six days. Furthermore, a deep learning neural network method is proposed in [50] which is an alternative fast screening method by X-ray analyzing. A machine learning algorithm is also used in [51] to classify the cases without diagnosis. Some other methods based on machine learning and artificial intelligence approaches are also recently proposed for some pandemic outbreak, including Covid-19 [52–54]. Second, metaheuristic algorithms play a key role in several research related to Covid-19 modeling. For this purpose, the modeling and prediction of Covid-19 spread is proposed based on Artificial Neural Network-Artificial Bee Colony (ANN-ABC) and Artificial Neural Network- Firefly Algorithm (ANN-FA) [55]. In [56], a modified grey wolf optimizer is presented for predicting the Covid-19 pandemic. Also, [57] has introduced an optimization algorithm named Covid-19 optimizer Algorithm (CVA) and proposed the model of coronavirus pandemic.

In this paper, modeling and forecasting of Covid-19 in two countries, including Iran and Italy, which had different experiences during the pandemic, are conducted by using LBO algorithm and the results are compared with some other algorithms. In this regard, in spite of the myriad of the offered model [58–61], the proposed model in [62] is employed as one of the most recent popular models, and the 16-parameter model will be tuned by the proposed algorithm for available data consists of the daily number of confirmed, dead, and recovered individuals of Covid-19 between 22 January 2020 and 4 August 2021 (561 days). As a matter of fact, each 20-day time series period is divided into two parts, the first 15-day period and the last 5-day. Thus, the local model is tuned by first 15 days and the model is validated by the last 5 days. In this optimization problem, the Mean Square Error (MSE) is the modeling index.

The rest of this paper is sorted out as follows: The lifetime of ladybugs in nature is explained in Sect. 2. In Sect. 3, the mathematical modeling of ladybugs' behavior is formulated. The performance evaluation of the proposed optimization algorithm is examined in Sect. 4 by implementing it on 78 benchmark functions. In Sect. 5, the proposed algorithm is executed for EEDP and the modeling of Covid-19, and the results are compared with several methods. Ultimately, the conclusion is expressed in Sect. 6.

2 Ladybugs in nature

The beetle family includes a broad group of more than 380,000 species, of which ladybugs are one. They vary greatly in body size, shape, color, and body anatomy that reflects their behavioral habits [63]. About 6000 species of ladybugs have been identified so far [64]. Many classifications have been done for ladybugs include six or seven families [65] and 38 tribes [66]. The life of ladybugs incorporates four main parts: eggs, larvae, pupae, and adults, respectively. The duration of each part of their life varies according to their geographical location. Unlike other beetles, they have the same nutrition in the pupae time and adult time. Ladybugs reproduce in summer and are completely sedentary during winter. In fact, by shortening the length of the day and cooling the air, ladybugs search for a suitable place in early winter and go to hibernation there. The growth rate of ladybugs depends largely on environmental temperature. The larvae feed for less than one month, and then, they become pupae. It takes ten days for a pupa to become an adult ladybug. Typically, a ladybug lives for a year, and therefore, it is going through one cold winter in its lifetime [67].

Ladybugs feed more than aphids. They can also feed on mites, small insects, and insect eggs. The adult ladybugs are about 1 to 10 mm, and their body color is a combination of red, yellow, and orange [68]. Several species of ladybugs live in cold winter climates and sometimes migrate to warmer areas. But many ladybugs hide under the bark of plants, cracks, or crevices in winter. Ladybugs spend the winter as the adult ladybugs and lay eggs the following summer. Then the eggs become larvae. Their larvae grow rapidly and reaching maturity in late summer or early fall [69, 70].

As mentioned, ladybugs are social creatures, and they always interact and coordinate with each other. A coordinated swarm of ladybugs looking for a place with more heat gathers, especially ladybugs in the cold weather conditions. In this paper, modeling ladybugs' searching and optimizing their behavior according to their swarm intelligence and coordinated behavior to find a safe and warm place to stay in winter are performed. As a result, LBO algorithm is presented. Therefore, each ladybug moves in the environment according to the position of other ladybugs, especially the ladybugs that have discovered a place with more heat. Several of them deviate from the proper path, while they move through the environment and die of the cold. Therefore, their number is always decreasing while searching for a warm place. In the next section, the mathematical modeling of LBO algorithm is presented.

3 Ladybug Beetle Optimization algorithm

The general process of most metaheuristic algorithms is similar to each other. In this process, the initial population of algorithms is evaluated and sorted based on their evaluation. Then, the population is updated and re-evaluated. After repeating a sufficient number of updating and evaluating processes for the population, the best solution is reported. This process for LBO algorithm is shown in Fig. 1.

In this section, modeling of LBO is presented. LBO is inspired by the coordinated movement of ladybugs in nature to find a location with the most heat. For this purpose, the initial population consists of $N(0)$ ladybugs are considered, and after following the steps described above, the final population includes $N\left(k_{\max}\right)$ ladybugs (in general, $N(0) \geq N\left(k_{\max}\right)$), and the optimal objective function is determined. Next, the modeling of LBO is done in three steps.

3.1 Define the objective function

The objective function is defined for the population in order to specify the heat of each location. It is equal to the heat value of each member's location. Therefore, the higher heat of the position results in the higher objective function value. However, to change the problem's structure to a standard state, the optimization problem is defined as a minimization problem. Therefore, the objective function is assumed to be the inverse amount of heat for each member's location of the population. In this case, the higher heat of a location is equivalent to its smaller objective function.

3.2 Update the population

As mentioned, the initial population consists of $N(0)$ ladybugs, which are placed randomly in the search space based on uniform distribution. The population of ladybugs is evaluated with the defined objective function and sorted. Then, the population moves to the location with the most heat according to a coordinated movement. Due to the social nature of ladybugs, they always move in coordination with the ladybugs' swarm while searching for a suitable location. Ladybugs follow each other through signals emitted by the group members. Therefore, they are more inclined to move toward their front ladybugs. In this modeling, front members are the ladybugs who have been able to find a place with more heat than the others. To balance the exploitation and exploration of the algorithm, a mutation step is considered for some of the individuals of the population, which is randomly employed for some individuals in each iteration. Thus, at each step of updating the position of the population, their position in the search space is updated either according to others' positions or mutation processes, which are described below.

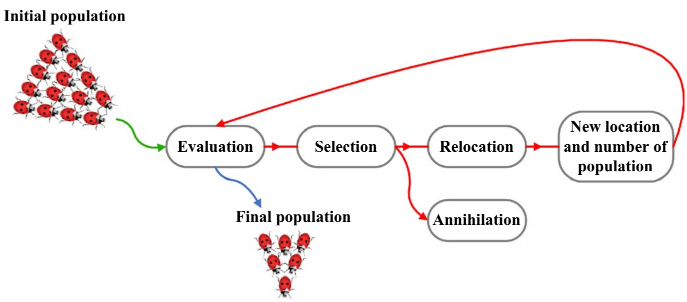


Fig. 1 LBO algorithm schematic

3.2.1 Update according to the position of other ladybugs

In each step, all the ladybugs' positions are updated and evaluated. The new and old positions of ladybugs are integrated, and the best group of them are chosen according to their objective function values. The new population is employed to update and evaluate in the next iteration. To update each member of the population in each iteration, another member of the population is selected using the method that will be described. For example, consider the aim is to update the ith ladybug of the population, and the jth ladybug of the population has been chosen to update this member. Consider the position of the ith ladybug in the kth iteration is shown by $x(k)$. This ladybug moves in the result of three vectors to update in the $(k+1)$th iteration. It moves a little toward the jth ladybug and also moves a little in the direction of the jth ladybug toward the $(j-1)$th ladybug. Finally, to create a balance between exploitation and exploration, and also, to escape the local minimums for the movement of ith ladybugs, the third direction is also considered, which is a coefficient of its current location. To maintain a random search, all the mentioned movements need to be multiplied by a random value. Additionally, the third movement, which is employed to avoid the local minimum, should be multiplied by the proportion of individual heat value to the total heat values of the population. As a result, a member of the population that traps in local minimum has the chance to escape because of the bigger coefficient of the third movement.

In the mathematical expression, the new position of the ith ladybug is obtained by (1).

$$\begin{aligned} x_i(k+1) = x_i(k) + \mathrm{rand} \times \left(x_j(k) - x_i(k)\right) \\ + \mathrm{rand} \times \left(x_j(k) - x_{j-1}(k)\right) + \mathrm{rand} \times \left|C_i\right|^{-\frac{k}{N(k)}} \times x_i(k), \end{aligned} \tag{1}$$

where C_i is equal to the ratio of the ith ladybird cost to the total cost of all the ladybirds in the kth iteration of the proposed optimization algorithm. The value of this parameter is calculated by

$$C_i = \frac{f\left(x_i(k)\right)}{\sum_{t=1}^{N^k} f\left(x_t(k)\right)} \tag{2}$$

Roulette-wheel selection is employed to choose the jth ladybug that was used to update the ith ladybug position in (1) [71]. In a simple expression, in order to select the jth ladybug from the $N(k)$ ladybugs, the distance between 0 and 1 is divided into the $N(k)$ unequal parts. Each part belongs to one of the ladybugs, and the length of each part is inversely related to the objective function of the corresponding ladybug. Thus, for a ladybug with the more optimal objective function, the length of the corresponding part is longer. Then, a random number between 0 and 1 is chosen. In the current step, the random number is determined in which part of the division is located. The corresponding ladybug of the selected part is chosen as the jth ladybug. It is clear that the ladybug with the warmer location (better objective function) has a higher chance to choose. The P vector corresponding to the population with $N(k)$ ladybugs is **defined** below. Roulette-wheel selection expects a an input vector, such as P.

$$P = \left[P_1, P_2, \dots, P_{N^k}\right], P_i = e^{-\beta \cdot \frac{f(x_i(k))}{f_{\text{worst}}}}, \tag{3}$$

where β is the pressure coefficient in the Roulette-wheel selection method and f_{worst} is the worst value of the objective function up to the current iteration during the algorithm process. The larger β, the better ladybugs of the population have a better chance of being selected on Roulette-wheel selection. Updating process of the ith ladybug is shown in Fig. 2 if the jth ladybug is selected in Roulette-wheel selection. As is clear, the new position of the ith ladybug is determined by the result of three vectors r_1, r_2, and r_3, and these make three terms of (1), respectively.

3.2.2 Update according to mutation process

Considering the mutation in the update process of the population is critical to exploring undiscovered parts of the search space and escaping from the local minima. Besides, a mutation stage in the search process leads to an increase in the algorithm's speed. Hence, the updating method of each ladybug's position, including according to other ladybugs and mutation, is randomly determined. In this regard, consider the ith ladybug should be mutated. The number of decision variables of the ith ladybug that must be mutated is calculated according to (4).

$$n_{\text{m}} = \text{round}\left(n * \mu_{\text{m}}\right), \tag{4}$$

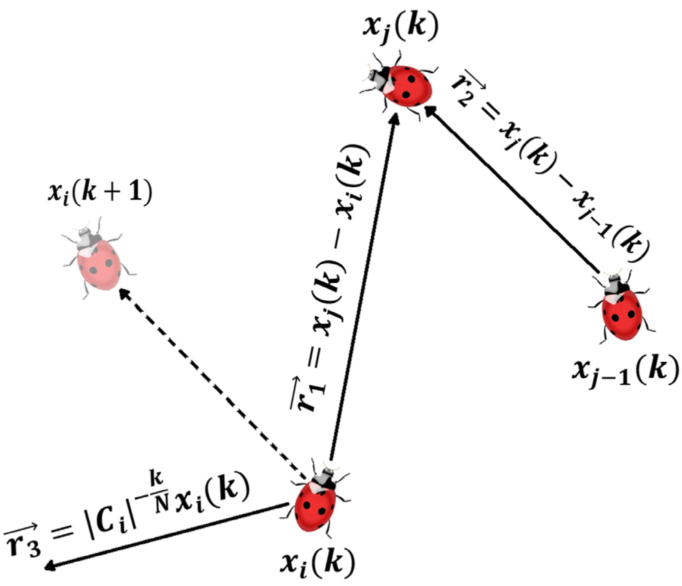


Fig. 2 Update position of the ith ladybug

where μ_{m} is the mutation rate and n is the length of the decision variable. Therefore, the n_m variables of available n variables of the ith ladybug are randomly selected. Then, the random variables in the feasible region are replaced to selected position of the ith ladybug.

According to the description provided, the pseudo code for updating the position of ladybugs in each iteration is shown in Algorithm 1.

3.3 Update the number of population size

During the search for a warm place, it is normal for the ladybugs to get lost and disappear. The ladybug(s) may move away from the others and annihilate due to the cold. The mathematical modeling of the ladybugs' annihilation during the search is considered in LBO. Therefore, the number of ladybugs in the various steps is calculated as

$$N(k+1) = \mathrm{round}\left(N(k) - \mathrm{rand} \times N(k)\left(\frac{\mathrm{NFE}}{\mathrm{NFE}_{\max}}\right)\right), \tag{5}$$

where NFE is the number of function evaluations and $\mathrm{NFE}_{\max}$ is the maximum of NFE. Thus, (5) is used if the number of function evaluations is the termination condition of LBO. Nevertheless, if the number of iterations is the condition for terminating the algorithm, the new number of ladybugs in each iteration is obtained by

$$N(k+1) = \mathrm{round}\left(N(k) - \mathrm{rand} \times N(K)\left(\frac{k}{k_{\max}}\right)\right), \tag{6}$$

where k is the iteration and $k_{\max}$ is the maximum iteration.

Algorithm 1. update position of the ladybugs

Inputs: the current position and all of the ladybugs objective function values,

Outputs: the next step position of all ladybugs,

Initialize: the number of ladybugs in current step k is equal to $N(k)$.

calculate the worst cost function, which is the biggest cost function of all iterations,
calculate the P vector, according to (3),
for $\boldsymbol{i}$ from 1 to number of the ladybugs **do**
  calculate the sum of the ladybugs' objective functions,
  create a random number (Rnd), between 0 and 1
  if $Rnd > R_M$, **do** (update the position of the ith ladybug according to other ladybugs' positions)
    $j = 0$;
    while (j<2) do
      determine the jth ladybug according to Roulette-wheel selection method,
    end while,
    calculate the ratio of ith ladybug's cost to sum of the objective functions, according to (2),
    calculate the next step position of ith ladybug $x_i(k+1)$, according to (1),
    calculate the cost function value of new population,
  else do (update the position of the ith ladybug with mutation)
    calculate the ith ladybugs variable numbers (n_m) that should be mutated, according to (4),
    create $n_m$random integer numbers between 1 and n (n is the length of the variable decision),
    create $n_m$random values in feasible region,
    replace the created $n_m$random values in the $n_m$random integer numbers in the ith ladybug,
  end if
  if $x_i(k+1) < x_{min}$ **do**
    $x_i(k+1) = x_{min}$,
  end if
  if $x_i(k+1) > x_{max}$ **do**
    $x_i(k+1) = x_{max}$,
  end if,
end for.
collect initial and new populations,
sort the population according to their cost function values,
ignore the worse and keep the best population as the final population of the current iteration.

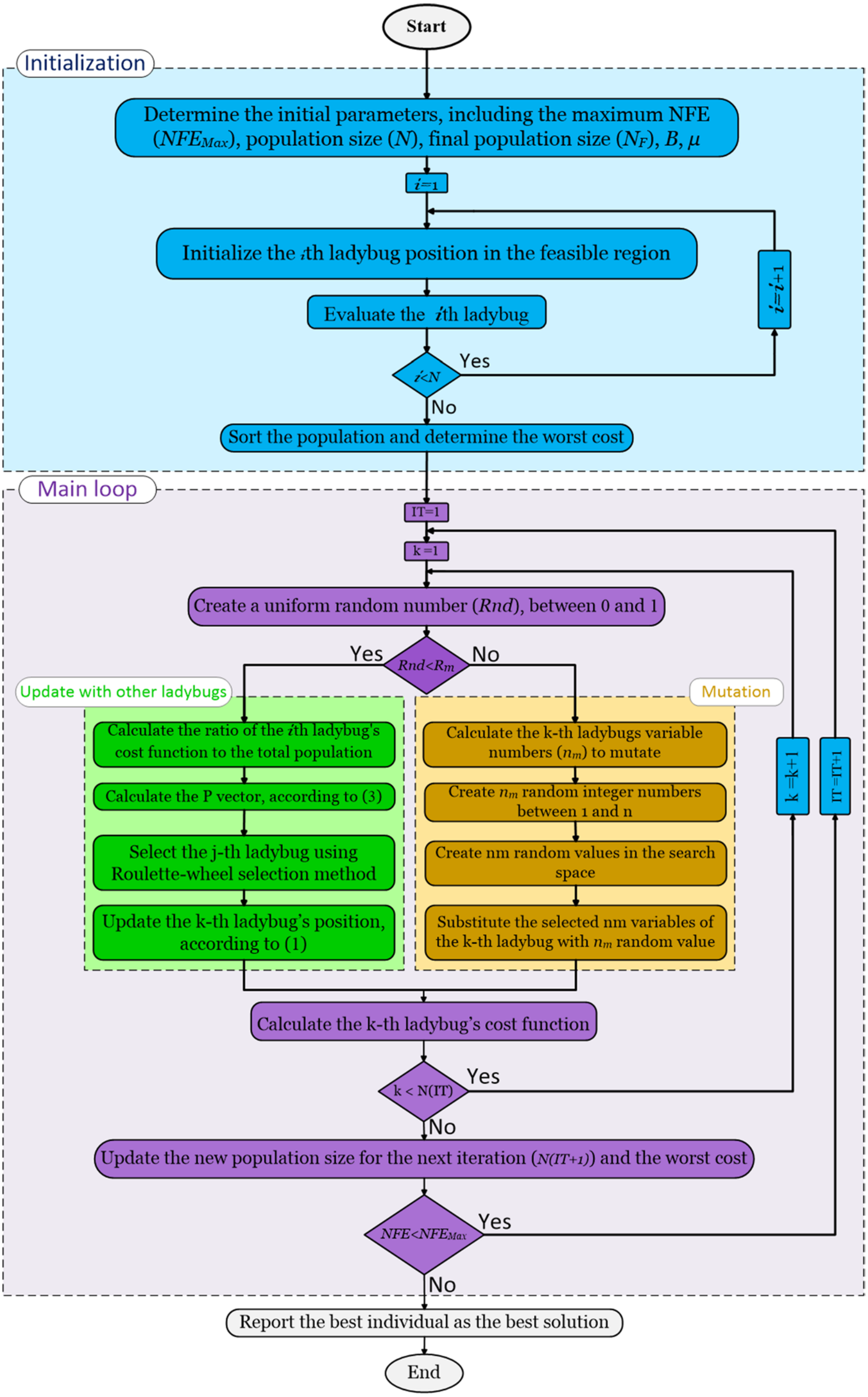


Fig. 3 Flowchart of the proposed LBO algorithm

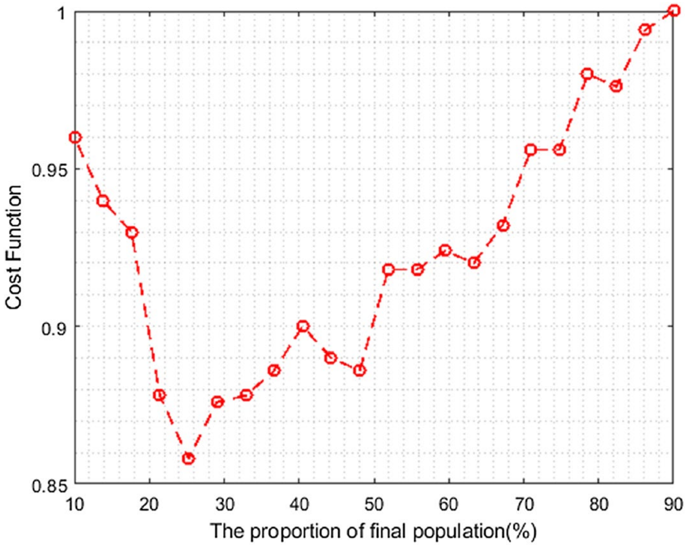


Fig. 4 Impact of the proportion of the final population value in the reduction process of the number of populations in LBO algorithm

Table 2 Unimodal fixed-dimension benchmark functions

No.	Name	D	Range	f_{opt}	No.	Name	D	Range	f_{opt}
f_1	Beale	2	[− 4.5, 4.5]	0	f_6	Wayburn Seader 3	2	[− 500, 500]	19.10588
f_2	Booth	2	[− 10, 10]	0	f_7	Leon	2	[− 1.2, 1.2]	0
f_3	Brent	2	[− 10, 10]	0	f_8	Cube	2	[− 10, 10]	0
f_4	Matyas	2	[− 10, 10]	0	f_9	Zettle	2	[− 5, 10]	− 0.00379
f_5	Schaffer N.4	2	[− 100, 100]	0.292579					

Table 3 Unimodal variable-dimension benchmark functions

No.	Name	D	Range	f_{opt}	No.	Name	D	Range	f_{opt}
f_{10}	Sphere	30	[− 100, 100]	0	f_{18}	Rosenbrock	30	[− 30, 30]	0
f_{11}	Power sum	30	[− 1, 1]	0	f_{19}	Brown	30	[− 1, 4]	0
f_{12}	Schwefel's 2.20	30	[− 100, 100]	0	f_{20}	Dixon and price	30	[− 10, 10]	0
f_{13}	Schwefel's 2.21	30	[− 100, 100]	0	f_{21}	Power singular	30	[− 4, 5]	0
f_{14}	Step	30	[− 100, 100]	0	f_{22}	Xin-she Yang	30	[− 20, 20]	0
f_{15}	Stepint	30	[− 5.12, 5.12]	− 155	f_{23}	Perm 0, D, Beta	5	[− 5, 5]	0
f_{16}	Schwefel's 2.22	30	[− 100, 100]	0	f_{24}	Sum squares	30	[− 10, 10]	0
f_{17}	Schwefel's 2.23	30	[− 10, 10]	0					

According to (5) or (6), the number of ladybugs is decreasing by performing the LBO and increasing the number of iterations. However, the decreasing trend should not lead to the extinction of all ladybugs. Therefore, by considering the minimum number

Table 4 Multimodal fixed-dimension benchmark functions

No.	Name	D	Range	f_{opt}	No.	Name	D	Range	f_{opt}
f_{25}	Egg crate	2	[− 5, 5]	0	f_{39}	Cross function	2	[− 10, 10]	0
f_{26}	Ackley N.3	2	[− 32, 32]	− 195.629	f_{40}	Cross leg table	2	[− 10, 10]	− 1
f_{27}	Adjiman	2	[− 1, 2]	− 2.02181	f_{41}	Crowned cross	2	[− 10, 10]	0.0001
f_{28}	Bird	2	[− 2π, 2π]	− 106.765	f_{42}	Easom	2	[− 100, 100]	− 1
f_{29}	Camel 6 Hump	2	[− 5, 5]	− 1.0316	f_{43}	Giunta	2	[− 1, 1]	0.060447
f_{30}	Branin RCOS	2	[− 5, 5]	0.397887	f_{44}	Helical Valley	3	[− 10, 10]	0
f_{31}	Goldstien Price	2	[− 2, 2]	3	f_{45}	Himmelblau	2	[− 5, 5]	0
f_{32}	Hartman 3	3	[0, 1]	− 3.86278	f_{46}	Holder	2	[− 10, 10]	− 19.2085
f_{33}	Hartman 6	6	[0, 1]	− 3.32236	f_{47}	Pen Holder	2	[− 11, 11]	− 0.96354
f_{34}	Cross-in-tray	2	[− 10, 10]	− 2.06261	f_{48}	Test Tube Holder	2	[− 10, 10]	− 10.8723
f_{35}	Bartels Conn	2	[− 500, 500]	1	f_{49}	Shubert	2	[− 10, 10]	− 186.731
f_{36}	Bukin 6	2	[− (15, 5), − (5, 3)]	180.3276	f_{50}	Shekel	4	[0, 10]	− 10.5364
f_{37}	Carrom Table	2	[− 10, 10]	− 24.1568	f_{51}	Three-Hump Camel	2	[− 5, 5]	0
f_{38}	Chichinadze	2	[− 30, 30]	− 43.3159					

Table 5 Multimodal variable-dimension benchmark functions

No.	Name	D	Range	f_{opt}	No.	Name	D	Range	f_{opt}
f_{52}	Schwefel's 2.26	30	[− 500, 500]	− 418.983	f_{61}	Styblinski-Tang	30	[− 5, 5]	− 1174.98
f_{53}	Rastrigin	30	[− 5.12, 5.12]	0	f_{62}	Griewank	30	[− 100, 100]	0
f_{54}	Periodic	30	[− 10, 10]	0.9	f_{63}	Xin-She Yang N. 4	30	[− 10, 10]	− 1
f_{55}	Qing	30	[− 500, 500]	0	f_{64}	Xin-She Yang N. 2	30	[− 2π, 2π]	0
f_{56}	Alpine N. 1	30	[− 10, 10]	0	f_{65}	Gen. Penalized	30	[− 50, 50]	0
f_{57}	Xin-She Yang	30	[− 5, 5]	0	f_{66}	Penalized	30	[− 50, 50]	0
f_{58}	Ackley	30	[− 32, 32]	0	f_{67}	Michalewics	30	[0, π]	− 29.6309
f_{59}	Trignometric 2	30	[− 500, 500]	0	f_{68}	Quartic Noise	30	[− 1.28, 1.28]	0
f_{60}	Salomon	30	[− 100, 100]	0					

Table 6 CEC-C06 2019 Benchmarks "The 100-Digit Challenge"

No.	Name	D	Range	f_{opt}
f_{69}	Storn's Chebyshev polynomial fitting program	9	[− 8192, 8192]	1
f_{70}	Inverse Hilbert matrix problem	16	[− 16382, 16382]	1
f_{71}	Lennard–Jones minimum energy cluster	18	[− 4, 4]	1
f_{72}	Rastrigin's function	10	[− 10, 1000]	1
f_{73}	Griewank's function	10	[− 10, 1000]	1
f_{74}	Weierstrass function	10	[− 10, 1000]	1
f_{75}	Modified Schwefel's function	10	[− 10, 1000]	1
f_{76}	Expanded Schaffer's F6 function	10	[− 10, 1000]	1
f_{77}	HappyCat function	10	[− 100, 100]	1
f_{78}	Ackley function	10	[− 100, 100]	1

Table 7 Parameter settings of algorithms used for comparative [30]

Name	Parameters
LBO	$N(0) = 60, \beta = 10, N_{\min} = 0.25N(0)$
EO [27]	$N = 60, \beta_{\min} = 0.2, \beta_{\max} = 0.8, P_{CR} = 0.2$
Chimp [35]	$N = 60, \alpha = 20, G_0 = 100$
PSO [16]	$N = 60, c_1 = 2, c_2 = 2, w = [0.9 \rightarrow 0.2]$
SCA [74]	$N = 60, a = 2, \left[r_1, r_2, r_3, r_4\right]$ from corresponding eq.
TLBO [29]	$N = 60,$
SLO [75]	$N = 60, P = 0.8$
ICA [76]	$N = 60, Pa = 0.25$

of ladybugs, it is not allowed to reduce the population to less than the assumed number. In this regard, the average of normalized optimal values of the 78 benchmark functions (introduce in Sect. 4) by LBO algorithm based on various proportion of the final population number is illustrated in Fig. 3. Obviously, reducing the population size to 20–30% initial size leads to minimum cost function. Hence, the minimum number of ladybugs in LBO is tuned on 25% its initial number. Thus, both (5) or (6) is rewritten as follows:

$$\begin{cases} N(k+1) = \max\left\{0.25N(0), \mathrm{round}\left(N(k) - \mathrm{rand} \times N(k)\left(\frac{\mathrm{NFE}}{\mathrm{NFE}_{\max}}\right)\right)\right\} \\ N(k+1) = \max\left\{0.25N(0), \mathrm{round}\left(N(k) - \mathrm{rand} \times N(k)\left(\frac{k}{k_{\max}}\right)\right)\right\} \end{cases}, \tag{7}$$

The flowchart of the proposed LBO algorithm is shown in Fig. 4.

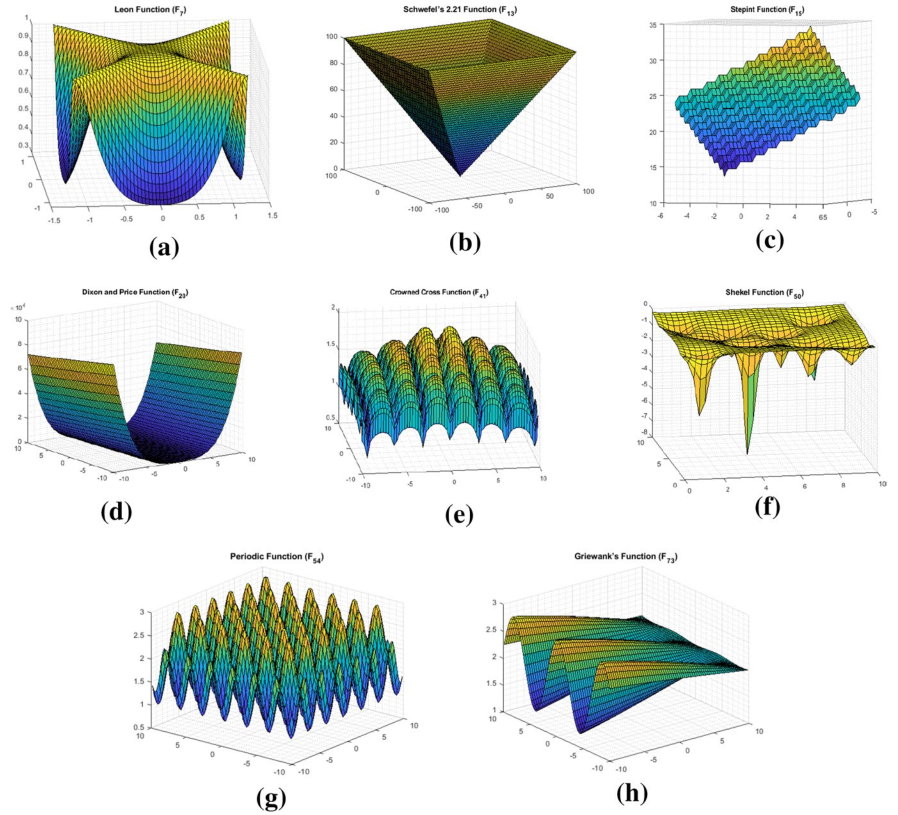


Fig. 5 Perspective views ($D = 2$) of 8 benchmark functions from all 5 benchmark function categories

4 LBO performance evaluation

A new optimization algorithm must be competitive with other efficient algorithms and has special abilities in solving some optimization problems. Besides, the exploration and exploitation capacities in the proposed algorithm must be balanced so that being able to solve variety of optimization problems. According to this point, the proposed algorithm will be evaluated with 78 well-known benchmark functions and will compare with some powerful algorithms in this section in order to specify its abilities.

4.1 Experimental setup

In this subsection, the performance of the proposed algorithm is evaluated on 78 different benchmark functions, and the results are compared with some state-of-the-art meta-heuristics optimization algorithms. These functions are classified into five different groups according to their nature and will be used to evaluate the proposed optimization algorithm. These five groups include unimodal fixed-dimension, unimodal variable-dimension, multimodal fixed-dimension,

Table 8 Comparison of the various algorithms results with LBO in unimodal fixed-dimension benchmark functions

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_1	Med	**0**	2.24E−08	0.0030	**0**	7.23E−05	**0**	1.60E−08	9.01E−05
	Mean	**0**	1.78E−08	0.255	**0**	7.88E−05	**0**	1.98E−08	0.303
	Std.	0	1.10E−08	0.4387	0	5.60E−05	0	1.44E−08	0.5257
f_2	Med	**0**	1.66E−08	0.0048	**0**	0.000148	**0**	1.36E−07	2.54E−07
	Mean	**0**	1.76E−08	0.0086	**0**	0.000313	**0**	1.77E−07	4.08E−07
	Std.	0	1.76E−08	0.0088	0	3.31E−04	0	1.61E−07	2.99E−07
f_3	Med	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**
	Mean	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**	**1.38E−87**
	Std.	4.5E−103	2.73E−103	2.73E−103	4.6E−103	4.6E−103	2.73E−103	4.60E−103	2.73E−103
f_4	Med	**0**	**0**	4.65E−05	1.30E−95	2.1E−121	1.09E−111	3.55E−09	**0**
	Mean	**0**	**0**	6.96E−05	7.36E−93	2.1E−121	9.59E−112	5.03E−09	**0**
	Std.	0	0	7.65E−05	3.08E−92	9.8E−125	9.02E−112	5.21E−09	0
f_5	Med	**0.292579**	**0.292579**	**0.2941095**	**0.292579**	**0.292579**	**0.292579**	**0.292579**	**0.292579**
	Mean	**0.292579**	**0.292579**	0.294677	**0.292579**	**0.292579**	**0.292579**	**0.292579**	**0.292579**
	Std.	8.30E−05	1.28E−10	0.002074	7.93E−17	1.76E−07	5.55E−17	6.19E−07	9.75E−08
f_6	Med	**19.10588**	**19.10588**	20.25673	**19.10588**	19.11139	**19.10588**	19.1067	19.1062
	Mean	**19.10588**	**19.10588**	22.47853	**19.10588**	19.11828	**19.10588**	19.1071	19.1062
	Std.	9.76E−11	9.36E−07	4.253951	9.17E−15	0.014647	2.51E−15	0.00145	0.000297
f_7	Med	**3.33E−29**	2.31E−07	0.030955	1.50E−22	6.50E−05	2.00E−17	1.25E−08	0.00902
	Mean	3.82E−13	2.88E−07	0.031153	**1.45E−20**	0.000133	6.90E−17	2.48E−08	0.00739
	Std.	1.06E−13	2.60E−07	0.030009	3.86E−20	0.000196	1.02E−16	3.40E−08	0.0048
f_8	Med	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
	Mean	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
	Std.	0	0	0	0	0	0	0	0
f_9	Med	**− 0.00379**	**− 0.00379**	**− 0.00379**	**− 0.00379**	**− 0.00379**	**− 0.00379**	**− 0.00379**	− 0.003777
	Mean	**− 0.00379**	**− 0.00379**	− 0.00373	**− 0.00379**	**− 0.00379**	**− 0.00379**	**− 0.00379**	− 0.003752
	Std.	2.19E−12	2.86E−14	8.84E−05	1.33E−18	2.89E−10	5.31E−19	2.90E−08	5.15E−05

Best-obtained results are shown in bold

Table 9 Comparison of the various algorithms results with LBO in unimodal variable-dimension benchmark functions

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{10}	Med	**2.44E−215**	1.43E−112	142.9852	2.10E−12	0.00029	8.61E−72	0.19031	3.24e−118
	Mean	**1.05E−203**	1.85E−110	145.2001	4.80E−11	0.00737	2.80E−71	0.21839	1.30E−181
	Std.	0	3.18E−110	14.3285	2.10E−10	0.02329	3.80E−71	0.06965	0
f_{11}	Med	**0**	**0**	0.588724	4.90E−25	2.20E−15	6.71E−164	8.60E−08	**0**
	Mean	**0**	**0**	0.658029	6.60E−22	2.00E−10	7.60E−164	8.80E−08	**0**
	Std.	0	0	0.172813	2.80E−21	8.00E−10	0	5.90E−08	0
f_{12}	Med	**1.89E−130**	1.59E−60	1205.5753	3.60E−06	1.50E−05	2.77E−35	2.62075	1.18E−106
	Mean	**3.64E−127**	3.24E−60	1200.9199	6.80E−06	5.90E−05	3.83E−35	2.8031	1.73E−100
	Std.	3.96E−113	2.95E−60	40.4981	8.90E−06	0.00013	2.27E−35	0.79795	0
f_{13}	Med	**2.74E−82**	3.17E−41	84.65790	0.41017	11.7186	3.05E−29	0.62211	2.15E−80
	Mean	**4.94E−76**	1.03E−40	86.3526	0.4066	13.742	3.66E−29	0.65522	2.27E−55
	Std.	1.75E−67	1.41E−40	3.46370	0.1181	8.471	1.48E−29	0.20993	3.93E−55
f_{14}	Med	7.14E−10	0.2706	6.119E+04	**2.10E−12**	4.11205	1.59E−10	0.19494	2.352
	Mean	1.24E−09	0.2008	6.240E+04	**1.50E−10**	4.20256	1.53E−09	0.19032	2.332
	Std.	1.89E−09	0.1293	3.828E+03	5.30E−10	0.4928	2.50E−09	0.02881	0.0347
f_{15}	Med	**− 155**	**− 155**	**− 155**	− 133	− 107	**− 155**	− 147	− 46
	Mean	**− 155**	**− 155**	**− 155**	− 132.16	− 106.16	**− 155**	− 147.8	− 43.333
	Std.	0	0	0	10.455	4.615	0	2.39	5.5075
f_{16}	Med	**1.17E−129**	8.22E−60	2.12E+40	9.70E−06	2.50E−06	1.15E−34	1.10E−19	3.94E−114
	Mean	**1.68E−127**	1.08E−59	1.10E+41	0.0002	7.50E−05	1.54E−34	1.10E−21	8.50E−102
	Std.	1.09E−114	5.41E−60	1.64E+41	0.00067	0.0003	6.76E−35	3.60E−21	0
f_{17}	Med	**0**	**0**	6.44E+06	2.1E−19	0.01581	1.4e−315	2.00E−15	**0**
	Mean	**0**	**0**	5.60E+06	2.80E−16	1711.85	**0**	1.60E−14	**0**
	Std.	0	0	2.642E+06	1.20E−15	8248.58	0	4.90E−14	0
f_{18}	Med	**24.8363**	25.1917	2.047E+07	25.9212	31.1826	26.57554	31.831	28.135
	Mean	**24.93023**	25.1244	2.033E+07	3643.9	73.3545	26.16673	265.89	28.130
	Std.	0.33733	0.37298	5.533E+06	17,994.4	103.248	0.959855	531.457	0.2128

Table 9 (continued)

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{19}	Med	**1.91E−230**	3.01E−117	34.3157	31	2.00E−08	2.08E−74	0.00053	1.038
	Mean	**1.36E−221**	2.08E−116	35.1876	39	7.80E−07	2.92E−74	0.00055	1.862
	Std.	0	3.12E−116	3.4617	24.1039	2.80E−06	1.85E−74	0.00018	1.819
f_{20}	Med	**0.6667**	**0.6667**	1.396E+04	0.67049	0.72082	**0.6667**	0.850	**0.6667**
	Mean	**0.6667**	**0.6667**	1.420E+04	9.1583	2.6131	**0.6667**	2.075	**0.6667**
	Std.	2.29E−05	1.90E−08	1.6172E+03	4.03	4.85844	4.68E−15	2.713	1.37E−09
f_{21}	Med	**2.17E−27**	8.45E−16	1.1243E+04	390.268	0.0122	1.62E−05	0.320	5.79E−18
	Mean	**3.68E−16**	4.43E−15	1.1425E+04	690.979	2.520	2.18E−05	0.320	2.46E−12
	Std.	7.34E−12	7.67E−55	3.77E+02	812.839	12.164	1.40E−05	0.152	4.26E−12
f_{22}	Med	**4.34E−232**	**4.34E−232**	**4.34E−232**	**4.00E−232**	**6.00E−199**	**4.34E−232**	**4.34E−232**	**4.34E−232**
	Mean	**4.34E−232**	**4.34E−232**	**4.34E−232**	**4E−232**	**5E−189**	**4.34E−232**	**4.34E−232**	**4.34E−232**
	Std.	0	0	0	0	0	0	0	0
f_{23}	Med	**0.0577**	1.3418	8.165E+04	0.0712	19.947	0.3783	0.22093	284.2
	Mean	**0.1843**	3.556	1.5670E+04	**0.1843**	21.540	0.9257	0.39812	614.6
	Std.	0.1504	5.211	1.553E+04	0.2790	13.235	1.154	0.48897	585.8
f_{24}	Med	**3.90E−227**	2.80E−112	1.02E+04	8.10E−10	2.60E−05	2.01E−72	0.1506	1.17E−211
	Mean	**1.99E−221**	1.24E−110	9.99E+03	68	0.00023	1.97E−72	0.1836	5.44E−214
	Std.	2.75E−201	2.12E−110	6.29E+02	98.826	0.00052	6.15E−73	0.1016	9.70E−200

Best-obtained results are shown in bold

Table 10 Comparison of the various algorithms results with LBO in multimodal fixed-dimension benchmark functions

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{25}	Med	**0**	**0**	0.000243	1.00E−129	4.40E−158	1.40E−159	2.26E−07	**0**
	Mean	**0**	**0**	0.000383	7.80E−128	3.50E−150	2.50E−159	2.94E−07	**0**
	Std.	0	0	0.000372	1.70E−127	1.70E−149	2.75E−159	2.60E−07	0
f_{26}	Med	**− 195.629**	**− 195.629**	− 195.6238	**− 195.629**	**− 195.629**	**− 195.629**	**195.629**	**− 195.629**
	Mean	**− 195.629**	**− 195.629**	− 195.621	**− 195.629**	**− 195.629**	**− 195.629**	**195.629**	**− 195.629**
	Std.	3.80E−11	1.75E−09	0.00455	5.80E−14	6.38E−05	0	2.90E−06	2.73E−06
f_{27}	Med	**− 2.021**	**− 2.021**	**− 2.021**	**− 2.021**	**− 2.021**	**− 2.021**	**2.021**	**− 2.021**
	Mean	− 2.018	**− 2.021**	**− 2.021**	**− 2.021**	**− 2.021**	**− 2.021**	**2.021**	**− 2.021**
	Std.	1.55E−11	7.02E−16	2.58E−08	1.36E−15	7.60E−10	0	3.49E−11	3.23E−07
f_{28}	Med	**− 106.765**	− 106.764	− 106.654	**− 106.765**	− 106.753	**− 106.765**	**− 106.765**	− 106.764
	Mean	− 102.874	− 106.764	− 106.599	**− 106.765**	− 106.745	**− 106.765**	**− 106.765**	− 106.764
	Std.	12.1483	1.58E−06	0.193	8.70E−15	0.019459	0	2.35E−06	3.06E−06
f_{29}	Med	**− 1.031**	**− 1.031**	− 1.030	**− 1.031**	**− 1.031**	**− 1.031**	**1.031**	**− 1.031**
	Mean	**− 1.031**	**− 1.031**	− 1.029	**− 1.031**	**− 1.031**	**− 1.031**	**1.031**	**− 1.031**
	Std.	1.85E−08	4.17E−09	0.00247	6.72E−16	1.29E−05	0	5.75E−08	2.21E−08
f_{30}	Med	**0.3978**	**0.3978**	0.40321	**0.3978**	0.3981	**0.3978**	**0.3978**	**0.3978**
	Mean	**0.3978**	**0.3978**	0.4018	**0.3978**	0.3982	**0.3978**	**0.3978**	**0.3978**
	Std.	3.80E−08	4.92E−08	0.0027	0	0.000339	0	2.58E−08	7.63E−09
f_{31}	Med	**3**	**3**	3.0316	**3**	3.000004	**3**	**3**	**3**
	Mean	**3**	**3**	3.0273	**3**	3.00001	**3**	3.000001	12.000001
	Std.	5.26E−07	2.60E−10	0.0192	9.46E−16	1.33E−05	3.14E−16	6.55E−07	15.588
f_{32}	Med	**− 3.862**	**− 3.862**	− 3.8525	**− 3.862**	− 3.854	**− 3.862**	**− 3.862**	− 3.8527
	Mean	**− 3.862**	**− 3.862**	− 3.8522	**− 3.862**	− 3.855	**− 3.862**	**− 3.862**	− 3.8543
	Std.	7.07E−06	4.55E−07	0.00201	0.001576	0.002786	0	3.02E−07	0.0033
f_{33}	Med	− 3.203	− 3.321	− 2.5618	− 3.199	− 3.075	**− 3.322**	− 3.321	− 3.158
	Mean	− 3.250	− 3.282	− 2.397	− 3.220	− 3.031	**− 3.322**	− 3.264	− 3.120
	Std.	0.0679	0.0686	0.428	0.1130	0.1455	1.29E−15	0.0607	0.0885

Table 10 (continued)

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{34}	Med	**− 2.062**	**− 2.0626**	**− 2.062**	**− 2.062**	**− 2.062**	**− 2.062**	**2.0626**	**− 2.0626**
	Mean	**− 2.062**	**− 2.0626**	**− 2.062**	**− 2.062**	**− 2.062**	**− 2.062**	**2.0626**	**− 2.0626**
	Std.	2.08E−11	8.58E−10	0.00046	9.06E−16	5.31E−06	0	6.14E−09	5.50E−09
f_{35}	Med	**1**	**1**	1.0243	**1**	**1**	**1**	1.0027	**1**
	Mean	**1**	**1**	1.033	**1**	**1**	**1**	1.00291	**1**
	Std.	0	0	0.0162	0	0	0	0.0016	0
f_{36}	Med	**180.327**	**180.327**	**180.327**	**180.327**	**180.327**	**180.327**	**180.3276**	**180.327**
	Mean	**180.327**	**180.327**	**180.327**	**180.327**	**180.327**	**180.327**	**180.3276**	**180.327**
	Std.	0	0	0	0	0	0	0	0
f_{37}	Med	**− 24.156**	**− 24.156**	− 23.865	**− 24.156**	− 24.143	**− 24.156**	**− 24.1568**	**− 24.156**
	Mean	**− 24.156**	**− 24.156**	− 23.890	**− 24.156**	− 24.138	**− 24.156**	**− 24.1568**	− 22.174
	Std.	3.73E−07	3.71E−07	0.0833	8.97E−15	0.0156	0	1.60E−06	3.4339
f_{38}	Med	**− 42.944**	**− 42.944**	− 42.496	**− 42.944**	− 42.943	**− 42.944**	**− 42.944**	− 39.155
	Mean	**− 42.944**	**− 42.944**	− 42.496	− 42.872	− 42.942	**− 42.944**	**− 42.944**	− 34.322
	Std.	0	3.21E−07	0.00018	0.1673	0.0017	0	4.06E−06	11.8050
f_{39}	Med	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**
	Mean	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**	**4.85E−05**
	Std.	6.36E−16	2.46E−14	4.57E−09	6.92E−21	1.10E−10	0	1.36E−13	1.09E−13
f_{40}	Med	− 0.0028	− 0.0004	− 0.0002	− 0.0847	− 0.00037	− 0.0271	0.00035	**− 1**
	Mean	− 0.0032	− 0.0003	− 0.0002	− 0.1541	− 0.3798	− 0.0315	0.00038	**− 1**
	Std.	0.00015	5.98E−05	3.41E−05	0.2551	0.4820	0.0261	0.000141	0
f_{41}	Med	0.0030	0.2301	0.5222	0.00118	0.4130	0.00118	0.227	**0.0001**
	Mean	0.0054	0.1655	0.4967	0.001012	0.2823	0.0097	0.227	**0.0001**
	Std.	0.0045	0.144	0.0662	0.0004	0.2407	0.0148	0.0372	0

Table 10 (continued)

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{42}	Med	**− 1**	− 0.9999	− 0.9483	**− 1**	− 0.999	**− 1**	− 0.99999	− 8.11E−05
	Mean	**− 1**	− 0.9999	− 0.957	**− 1**	− 0.999	**− 1**	− 0.99999	− 0.3333
	Std.	0	3.04E−08	0.0180	0	0.0003	0	8.90E−06	0.5773
f_{43}	Med	**0.06447**	**0.0644**	**0.0644**	**0.06447**	**0.06447**	**0.06447**	**0.06447**	**0.0644**
	Mean	**0.06447**	**0.0644**	0.0645	**0.06447**	**0.06447**	**0.06447**	**0.06447**	**0.0644**
	Std.	1.23E−09	1.16E−09	0.00014	4.84E−17	6.06E−06	0	7.09E−10	4.48E−10
f_{44}	Med	7.15E−18	1.57E−06	17.5734	**1.53E−38**	0.000147	8.96E−33	1.75E−05	2.64E−09
	Mean	0.010295	3.36E−06	21.1670	**4.36E−34**	0.001891	7.77E−33	5.41E−05	3.05E−09
	Std.	0.010378	4.46E−06	6.669	2.02E−33	0.005523	3.99E−33	7.55E−05	2.98E−09
f_{45}	Med	**0**	8.23E−07	0.1458	**0**	0.003713	**0**	1.44E−07	3.94E−07
	Mean	2.84E−31	1.18E−06	0.1598	2.84E−31	0.004536	**0**	2.39E−07	3.93E−07
	Std.	3.44E−31	1.33E−06	0.0351	3.86E−31	0.004803	0	2.74E−07	3.35E−07
f_{46}	Med	**− 19.208**	**− 19.208**	− 19.1907	**− 19.208**	− 19.2011	**− 19.208**	**− 19.208**	**− 19.208**
	Mean	**− 19.208**	**− 19.208**	− 19.187	− 19.161	− 19.1994	**− 19.208**	**− 19.208**	**− 19.208**
	Std.	8.83E−08	6.31E−07	0.0189	0.237557	0.00849	0	3.32E−07	3.43E−07
f_{47}	Med	**− 0.96353**	**− 0.96353**	− 0.96327	**− 0.96353**	− 0.96352	**− 0.96353**	**0.96353**	**− 0.96353**
	Mean	**− 0.96353**	**− 0.96353**	− 0.9633	**− 0.96353**	− 0.96352	**− 0.96353**	**0.96353**	**− 0.96353**
	Std.	1.59E−10	5.43E−10	0.0001	0	1.85E−05	0	6.19E−10	1.86E−10
f_{48}	Med	**− 10.872**	**− 10.872**	− 10.823	**− 10.872**	**− 10.872**	**− 10.872**	− 10.8723	**− 10.872**
	Mean	**− 10.872**	**− 10.872**	− 10.803	**− 10.872**	**− 10.872**	**− 10.872**	− 10.8652	**− 10.872**
	Std.	0.029456	8.82E−11	0.0782	3.63E−15	4.13E−07	0	0.009704	1.51E−09
f_{49}	Med	**− 186.731**	− 186.730	− 178.457	**− 186.731**	− 186.618	**− 186.731**	**186.731**	− 186.730
	Mean	− 184.205	− 186.730	− 175.941	**− 186.731**	− 186.549	**− 186.731**	**186.731**	− 165.679
	Std.	1.6308	4.00E−05	11.682	4.26E−14	0.167024	2.01E−14	7.45E−05	36.461

Table 10 (continued)

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{50}	Med	**− 10.5364**	− 10.530	− 3.6707	**− 10.5364**	− 4.91002	**− 10.5364**	**− 10.5364**	− 3.535
	Mean	− 9.75558	− 10.531	− 3.219	− 10.1038	− 5.05406	**− 10.5364**	− 9.89179	− 3.151
	Std.	2.7595	0.0024	2.089	1.497388	1.893045	0	1.783	0.8976
f_{51}	Med	**0**	**0**	4.14E−06	9.20E−129	9.80E−154	4.61E−159	7.66E−09	**0**
	Mean	**0**	**0**	3.65E−06	1.40E−126	2.20E−146	2.62E−157	1.22E−08	**0**
	Std.	0	0	3.07E−06	4.5E−126	9.7E−146	4.49E−157	1.12E−10	0

Best-obtained results are shown in bold

Table 11 Comparison of the various algorithms results with LBO in multimodal variable− dimension benchmark functions

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{52}	Med	132.222	131.081	339.882	199.384	280.834	**123.071**	152.85	225.112
	Mean	133.608	141.342	332.208	197.091	280.294	**122.686**	153.686	227.059
	Std.	28.323	19.8491	15.352	34.563	9.9261	5.352	19.59	10.376
f_{53}	Med	45.169	**0**	413.13	80.591	3.602	10.992	92.585	**0**
	Mean	44.414	**0**	414.643	79.828	10.188	8.315	96.532	**0**
	Std.	14.1753	0	9.22698	21.803	14.530	7.351	27.121	0
f_{54}	Med	**1.001**	2.1499	8.399	2.775	4.206	6.558	1.0021	1.008
	Mean	**1.008**	2.1284	8.24012	2.775	4.531	6.399	1.0091	**1.008**
	Std.	0.018293	0.3167	0.3358	0.955203	1.744219	0.447065	0.04	0
f_{55}	Med	0.1373	247.694	1.96E+11	**6.29E−11**	5111.529	1.43E−06	346.61	1634.80
	Mean	205.088	229.92	2.00E+11	**1.33E−09**	44,938.17	1.28E−06	348.86	1446.812
	Std.	452.3653	34.934	2.60E+08	5.36E−09	165,063.9	4.80E−07	107.58	365.55
f_{56}	Med	**7.09E−113**	1.93E−62	68.079	5.97E−06	0.005094	1.72E−36	3.688	0.2547
	Mean	0.1210	**7.26E−61**	66.006	0.3553	0.019103	1.07E−07	3.758	0.2547
	Std.	0.3052	1.23E−60	3.642	1.229	0.047921	1.85E−07	1.351	0.0041
f_{57}	Med	**1.29E−111**	5.77E−68	1.664E+13	50,742.82	1.28E−05	1.29E−17	0.0331	0.0012
	Mean	**3.65E−72**	1.35E−67	2.680E+13	1,928,062	0.000228	1.33E−13	2.602	0.0411
	Std.	1.83E−71	3.02E−73	3.252E+13	6,519,140	0.00045	2.30E−13	10.108	0.030
f_{58}	Med	2.739	2.66E−15	19.961	1.40E−06	18.603	2.66E−15	0.650	**8.88E−16**
	Mean	2.671	2.66E−15	19.961	3.95E−06	13.925	3.85E−15	0.735	**8.88E−16**
	Std.	1.2074	0	0.00030	6.03E−06	7.874642	2.05E−15	0.619771	0
f_{59}	Med	40.753	17.4914	1.540E+04	**5.933**	103.861	9.521	182.020	123.849
	Mean	40.161	23.8879	1.550E+04	**8.117**	107.440	11.630	182.154	126.480
	Std.	29.812	11.5015	5.309E+03	4.972	20.735	7.862	27.56	9.937
f_{60}	Med	**0.09987**	**0.09987**	24.91	0.2998	0.1999	0.09988	0.4998	**0.09987**
	Mean	0.2319	**0.09987**	25.052	0.3478	0.2561	0.09988	0.5478	**0.09987**
	Std.	0.382	9.85E−10	0.4318	0.0653	0.0962	3.37E−09	0.0962	3.08E−10

Table 11 (continued)

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{61}	Med	− 1019.4	− 1072.812	− 524.640	**− 1076.03**	− 612.026	− 1019.48	− 1005.34	− 524.1970
	Mean	− 1016.7	− 1064.21	− 513.89	**− 1085.08**	− 613.355	− 1024.19	− 1011.56	− 522.4848
	Std.	30.5915	61.3522	62.87	31.57904	31.91868	49.64653	38.73212	14.40772235
f_{62}	Med	0.0149	**0**	17.797	0.0123	0.00159	**0**	0.0346	**0**
	Mean	0.0220	**0**	18.208	0.0111	0.1153	**0**	0.0343	**0**
	Std.	0.02868	0	0.8618	0.0064	0.2115	0	0.0096	0
f_{63}	Med	− 0.13201	2.82E−13	2.62E−07	**9.39E−25**	1.59E−10	1.22E−19	6.45E−16	4.19E−08
	Mean	− 0.17082	3.36E−13	2.69E−07	1.88E−14	1.69E−10	**2.70E−16**	6.52E−16	5.28E−08
	Std.	0.16543	1.46E−13	2.71E−08	6.50E−14	8.18E−11	4.68E−16	1.68E−16	4.60E−08
f_{64}	Med	**1.16E−11**	2.08E−11	3.14E−11	3.06E−11	3.47E−10	3.07E−11	1.79E−11	3.15E−05
	Mean	**1.01E−11**	2.12E−11	3.14E−11	3.06E−11	3.82E−10	3.04E−11	2.33E−11	4.76E−05
	Std.	4.33E−12	8.62E−13	1.13E−13	1.34E−12	1.95E−10	6.49E−13	1.51E−11	4.21E−05
f_{65}	Med	0.0109	0.118	8.04E+07	**4.92E−12**	2.3782	0.0109	0.0343	2.8017
	Mean	0.0465	0.1262	7.94E+07	**0.004395**	3.0277	0.0255	0.039	2.7459
	Std.	0.0906	0.1151	2.201E+07	0.0054	1.743	0.0252	0.0193	0.1732
f_{66}	Med	1.49E−08	0.00824	5.530E+08	**1.99E−14**	0.6550	5.08E−12	1.234	0.3530
	Mean	1.99E−08	0.0116	5.68E+08	0.0041	1.219	**1.36E−11**	1.296	0.3353
	Std.	1.95E−08	0.0079	8.565E+08	0.0207	1.671	1.70E−11	1.224	0.03274
f_{67}	Med	**− 20.6989**	− 9.493	− 7.039	− 17.865	− 7.935	− 20.664	− 15.1814	− 7.67923
	Mean	− 19.0636	− 10.4140	− 7.343	− 17.925	− 7.937	**− 21.874**	− 14.8792	− 7.80731
	Std.	4.039	1.629	1.374	1.839	0.838	5.374	1.6625	0.3819
f_{68}	Med	**0.000677**	0.00014	89.3487	2.720734	0.016155	0.000957	0.0108	**0.000677**
	Mean	**0.000804**	0.0001	99.586	4.873331	0.02002	0.000863	0.012	**0.000804**
	Std.	0.000436	7.73E−05	17.77	6.195275	0.012613	0.000242	0.0049	4.23E−04

Best-obtained results are shown in bold

Table 12 Comparison of the various algorithms results with LBO in CEC-C06 2019 Benchmark functions

No.	Index	LBO	EO	Chimp	PSO	SCA	TLBO	SLO	ICA
f_{69}	Med	5.13E+09	41,925.6	2.01E+12	8.99E+07	**2.95E+05**	8.92E+07	4.48E+07	1.53E+06
	Mean	4.88E+09	41,242.3	2.14E+12	6.85E+07	**3.41E+05**	1.08E+08	9.93E+07	1.82E+06
	Std.	4.44E+07	2598.4	3.80E+11	4.81E+07	1.64E+04	5.45E+06	1.10E+07	8.48E+06
f_{70}	Med	**17.342**	17.343	7400.52	**17.342**	19.847	**17.342**	**17.342**	19.22
	Mean	**17.342**	17.344	7672.37	**17.342**	19.284	**17.342**	**17.342**	19.300
	Std.	9.75E−06	0.00067	2710.906	0	0.9760	0	4.04E−06	0.2161
f_{71}	Med	**12.702**	**12.702**	**12.702**	**12.702**	**12.702**	**12.702**	**12.7024**	**12.702**
	Mean	**12.702**	**12.702**	**12.702**	**12.702**	**12.702**	**12.702**	**12.7024**	12.7025
	Std.	4.83E−07	8.66E−07	4.07E−06	0	8.758E−05	3.08E−15	0	0.00025
f_{72}	Med	**9.985**	63.375	26,620.82	22.883	21.916	24.841	13.929	5083.447
	Mean	**10.388**	72.268	26,678.62	20.562	34.271	21.832	15.919	7358.37
	Std.	7.593	15.730	4019.15	8.692	21.937	7.048	4.3369	4779.17
f_{73}	Med	1.118	1.4335	7.477	1.103	1.220	1.074	**1.039**	2.8190
	Mean	1.086	1.467	6.923	1.086	1.197	1.073	**1.0377**	2.8445
	Std.	0.066	0.0856	1.085	0.0298	0.0537	0.0098	0.0124	1.0410
f_{74}	Med	**3.517**	9.980	13.394	4.956	4.649	10.624	10.021	7.961
	Mean	**3.492**	9.583	12.990	5.030	4.626	10.479	9.86529	8.0036
	Std.	0.826	1.135	1.095	1.483	0.2289	0.3375	0.3261	0.7401
f_{75}	Med	**21.09**	412.07	1515.175	22.945	391.580	426.404	117.731	117.156
	Mean	**22.871**	436.58	1469.099	28.907	277.359	434.960	133.609	153.181
	Std.	4.089	194.94	337.789	97.045	206.530	32.860	106.064	112.92
f_{76}	Med	**3.554**	4.633	6.859	4.880	4.132	5.134	4.821	5.665
	Mean	**3.879**	4.234	6.811	4.795	4.684	5.081	4.067	5.658
	Std.	0.938	0.6976	0.138	0.625	1.187	0.9843	0.527	0.293
f_{77}	Med	2.344	2.665	6212.14	2.345	2.861	**2.342**	2.557	8.289
	Mean	2.345	2.928	6620.28	2.344	2.948	**2.343**	2.521	34.577
	Std.	0.0035	0.461	1528.594	0.0012	0.1650	0.0020	0.0758	47.362
f_{78}	Med	20.012	20.339	20.55	**19.999**	20.093	20.223	20.095	20.114
	Mean	20.018	20.367	20.516	**19.999**	20.083	17.563	20.157	20.113
	Std.	0.0176	0.053	0.0874	6.42E−05	0.0261	4.700	0.104	0.0175

Best-obtained results are shown in bold

multimodal variable-dimension [30], and CEC-C06 2019 [72, 73] benchmark functions, respectively. Dimension, upper and lower bounds, and the global optimal value is given in Tables 2, 3, 4, 5, and 6, respectively. In these tables, No., Name, D, Range, and f_{opt} are the number, name, dimension, upper and lower bounds, and the global optimum value of each function, respectively.

LBO is implemented for each of the functions in Tables 2, 3, 4, 5, and 6, and its results are compared with several most popular optimization algorithms, including Equilibrium Optimizer (EO) [27], Chimp [35], PSO, Sine Cosine

Algorithm (SCA) [74], TLBO, Seven-spot Ladybird Optimization (SLO) [75], and Imperialist Competitive Algorithm (ICA) [76]. The parameters set for the seven algorithms used to compare and the proposed algorithm are given in Table 7. Besides, several functions' 2D landscapes from all five-category function are presented in Fig. 5, which demonstrates the variety of the studied functions' types.

4.2 Evaluation of LBO algorithm

In this subsection, the results of the 78 benchmark functions are employed to evaluate the LBO algorithm. For this purpose, LBO and seven popular algorithms are used to optimize the mentioned functions and the results are compared. All the algorithms used for comparison and LBO are executed 30 times for all benchmark functions, and the median, average, and standard deviation of the best solutions are represented. Also, in all algorithms, the termination condition is considered the number of function evaluations, which are set at 50,000. The results obtained from the implementation of the algorithms on the benchmark functions are given in Tables 8, 9, 10, 11, and 12.

According to the results of Table 8, LBO is quite powerful in optimizing these functions and only has not gained the first position in mean value of the best algorithm in function f_7 compared to the other algorithms. However, in this function, the results of LBO are comparable, and only the results of the PSO algorithm are more appropriate. These benchmark functions are considered simple functions for the optimizers. Among the other seven high-performance algorithms, the results of the PSO algorithm are considerable, which is the second algorithm after LBO. Overall, the proposed algorithm demonstrates the best performance among the others, making it the best algorithm in the first comparison.

By examining the results obtained for the optimization of the second benchmark functions group, it is clear that the proposed optimization algorithm has a high ability to optimize this group of functions, and in most of them, there is a dramatic improvement in the results. In the f_{10},f_{12},f_{13},f_{16},f_{18},f_{19},f_{21},f_{23}, and f_{24} functions, the proposed LBO algorithm has a much more appropriate optimal solution than other algorithms and has shown its superiority. The proposed algorithm is the only one that has reached the best solution among others in these 9 benchmark functions. In addition to the results related to the optimal points, the speed of the proposed algorithm is better than that of the others in most cases of this group of benchmark functions. Therefore, LBO is a suitable algorithm for unimodal variable-dimension benchmark functions. Also, the proposed algorithm in the f_{11},f_{15},f_{17}, f_{20}, and f_{22} functions has reached the same optimum value as several other algorithms, in which LBO is still an appropriate algorithm

Table 13 General comparison of 78 benchmark functions' results for the all algorithms

	LBO	ICA	Chimp	PSO	SCA	TLBO	SLO	EO
Number of benchmark functions with the best result	**58 (73.3%)**	33 (42.3%)	12 (15.3%)	37 (47.4%)	17 (21.8%)	38 (48.7%)	24 (30.7%)	33 (42.3%)
Number of functions that alone reached to the optimal value	**16 (20.5%)**	0 (0%)	0 (0%)	2 (2.6%)	1 (1.3%)	1 (1.3%)	1 (1.3%)	2 (2.6%)

Best-obtained results are shown in bold

for this group of functions. The results obtained in several algorithms only in the f_{14} function have a more optimal value than LBO, but the results of the proposed algorithm are still comparable in this function.

The results obtained in Tables 8 and 9 show that LBO has a high ability in exploitation. The great advantage of LBO in exploiting is for two main reasons: First, according to the process of updating the ladybugs' position, each ladybug is updated by the other ladybug that was selected by Roulette-wheel selection. Therefore, more importance is given to better ladybugs, and the probability of selecting better ladybugs to update each one is higher. Therefore, the exploitative behavior of LBO is desirable. Second, the second term in updating the position of each ladybug that specified in (1) is the direction of the selected ladybug in the Roulette-wheel selection toward the next ladybug, which caused the ladybug to be more inclined toward the better ones.

The results of the LBO implementation on multimodal fixed-dimension benchmark functions with the result of the other algorithms in Table 10 demonstrate that the proposed algorithm has achieved the optimal value obtained by the other algorithms in most functions. The LBO only does not specify the optimal value in the f_{33}, f_{40}, f_{41}, and f_{44} functions of 27 tested benchmark functions of this benchmark function group, which means the proposed algorithm reached the best solution in more than 85% of benchmark functions. These results make LBO the best algorithm compared to the others. Also, by observing the results obtained in Table 11, related to the multimodal variable-dimension benchmark function group, it is clear that the optimal values of several benchmark functions of this group have been obtained using the proposed LBO. Of course, this is also true in the other algorithms. These results indicate that the optimal values of f_{56}, f_{57}, f_{60}, and f_{68} benchmark functions have been obtained only by LBO, and the results obtained from this algorithm are close and comparable to the achieved optimal results in most other functions. The multimodal functions that the optimization results are present in Tables 10 and 11 have many local minimums, and the optimization algorithms must escape them. As mentioned, LBO has been able to distance well from local optimal minimums of multimodal fixed-dimension benchmark functions and stuck in local minimums in a few functions of the last group, however. Therefore, the global optimal minimums have not achieved.

Table 12 demonstrates the results of the optimization algorithms for CEC-C06 2019 Benchmark functions, a challenging group of benchmark functions for optimizers. Looking at the results in this table, the optimal value of f_{72},f_{74}, f_{75}, and f_{76} functions is obtained only by LBO algorithm, and in f_{70} and f_{71} shares the first position with some other algorithms, which means the proposed algorithm reaches the best solution in 60% of the all functions of this group. Additionally, the obtained results in f_{73}, f_{77}, and f_{78} functions are also so close to the best answer. LBO is also the best algorithm in this comparison, like all four previous comparisons.

4.3 Analysis of the results

In the previous subsection, all eight algorithms were employed for 78 benchmark functions in five separated categories and the results were reported. The obtained results are compared in the following. Looking at Tables 8, 9, 10, 11, and 12, it is obvious that the algorithm is an appropriate optimizer for simple and even more complex convex optimization problems. Moreover, the performance of the algorithm is better than the other in the nonconvex optimization problems. The number of optimization problems that each algorithm has reached to the best cost is illustrated in Table 13, which is the first part of the comparision of the eight algorithms. Moreover, the number of problems in that the algorithms alone have determined the optimal values are shown. According to these results, LBO algorithm has succeeded to report the optimal values of 58 benchmark functions of 78 available functions and 16 functions alone reached the optimal values, making it substantially the best algorithm among others.

Looking at Fig. 6, the convergence curves of the 8 functions, selected randomly from all benchmark functions, are plotted to present the second part of eight algorithms' comparison. The LBO curves indicate the best path among others in most of eight functions. Based this figure, LBO keeps improving linearly without getting trapped in any local optimum and was one of the first algorithms that leads to optimal values. As shown in this figure, not only does the algorithm reach the best solutions in most of the 8 shown cost function curves, but the speed of the algorithm also is the best among the other, which is an important index to compare the algorithms.

In addition, to draw a reliable conclusion and illustrate the superiority of the proposed algorithm, a statistical test is conducted in the following. In this regard, the box plot analysis of eight optimization algorithms is demonstrated for eight benchmark functions in Fig. 7. Except f_{50} function, LBO algorithm demonstrates the best result in all of functions.

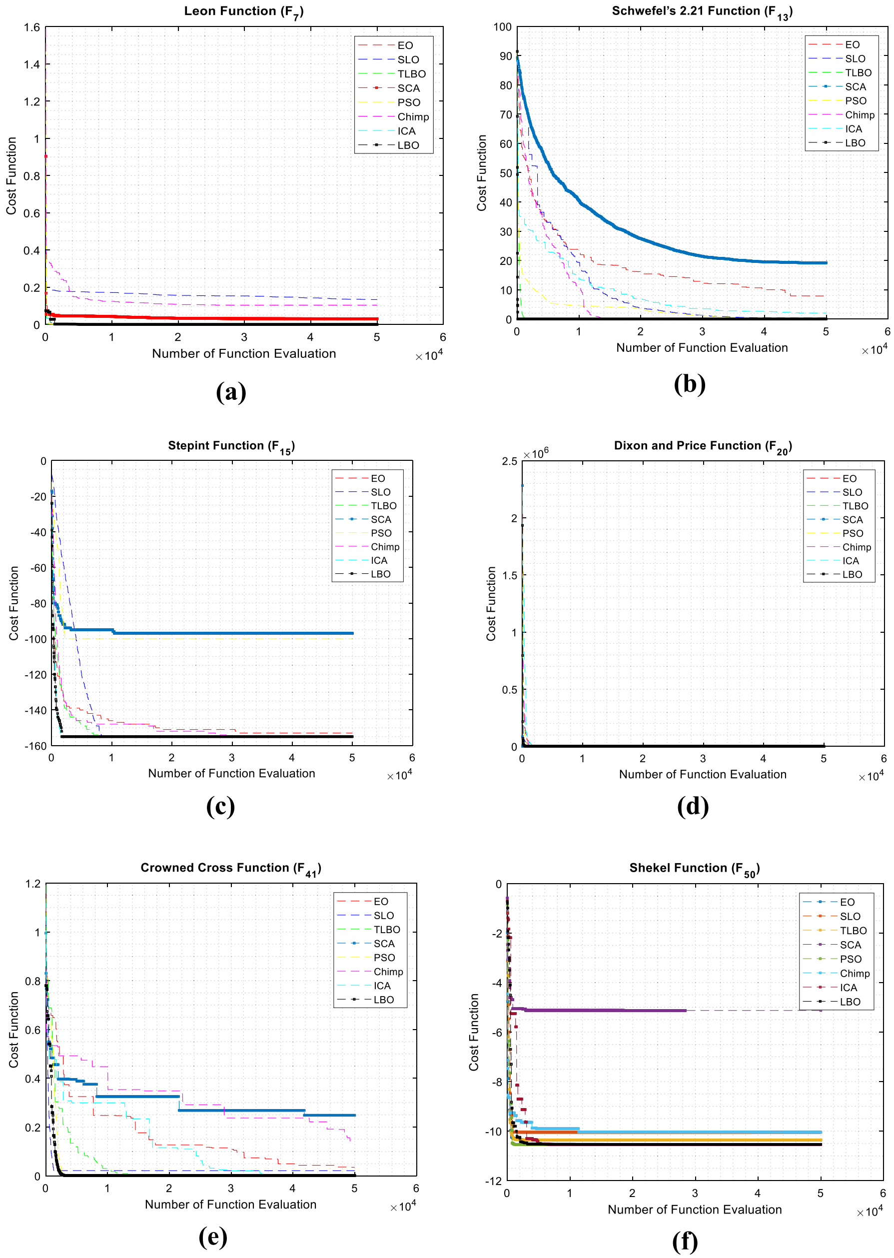
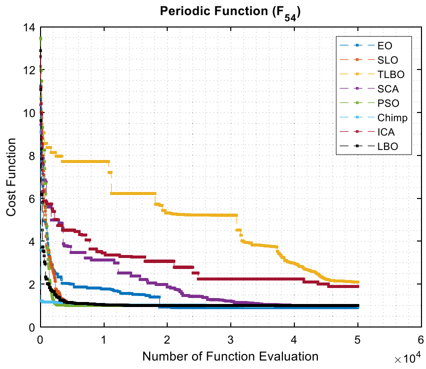
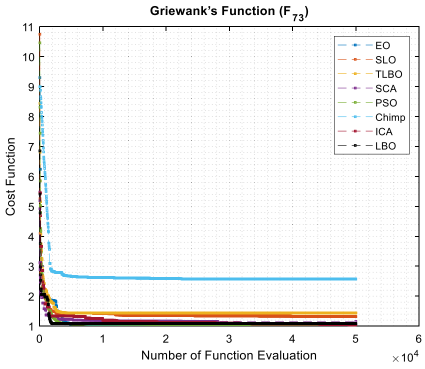


Fig. 6 Comparison of the various algorithms' convergence curves with LBO for 8 benchmark functions, including **a** f_7, **b** f_{13}, **c** f_{15}, **d** f_{20}, **e** f_{41}, **f** f_{50}, **g** f_{54}, and **h** f_{73}



(g)



(h)

Fig. 6 (continued)

5 Real-world engineering optimization case-studies

Variety of the real-world optimization problems is the main reason for the variety of optimizers; each optimization algorithm is able to properly solve a group of practical problems. In this regard, two different real-world optimization problems are solved by LBO in this section to prove the high performance of the proposed algorithm, EEDP and Covid-19 pandemic modeling. EEDP is one of the most important optimization problems for electrical engineers. The purpose of the EEDP is to choose the power optimal value for each power plant, in a way that first, the power demand by the consumer and other constraints imposed by the conditions such as the generation constraint of each power plant responds, and second, the total costs of power generation and emission of polluted gases are minimized. EEDP is one of the multi-objective optimization problems [77]. Moreover, Covid-19 pandemic is one of the most important concerns of human nowadays, and accurate forecasting is essential for deliberate decision-making by the authorities. Hence, LBO is employed to solve EEDP and predict the confirmed and recovered trend of Covid-19 in this section.

5.1 Economic-environmental dispatch problem

In the following of this subsection, mathematical modeling of EEDP is expressed, and then, the results of LBO implementation for the three IEEE standard grid to optimize the generation cost and emission are presented and compared with the other methods.

5.1.1 EEDP formulation

This part describes the objective function and constraints of EEDP.

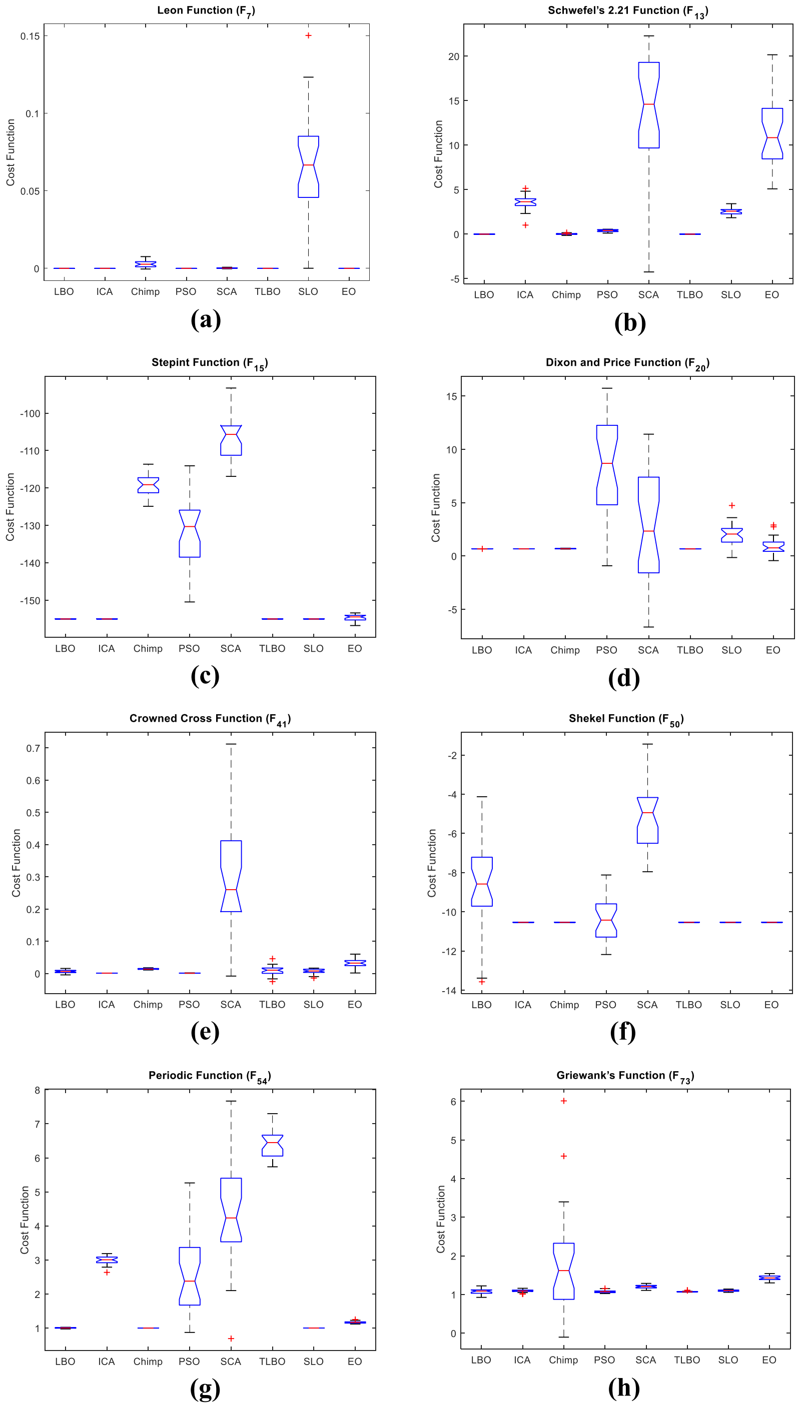


Fig. 7 Box plot analysis of all employed algorithms for 8 benchmark functions, including **a** f_7, **b** f_{13}, **c** f_{15}, **d** f_{20}, **e** f_{41}, **f** f_{50}, **g** f_{54}, and **h** f_{73}

Table 14 Characteristics of the test systems for EEDP

System	Number of units	Power demand (MW)
Case I [80]	3	500
Case II [81]	10	2000
Case III [82]	40	10,500

5.1.1.1 Objective function of EEDP Consider a power grid with N power plant. The cost function of the ith power plant for generate P_i MW power is presented as follows [78]:

$$F_{Pi}\left(P_i\right) = a_i P_i^2 + b_i P_i + c_i + \left| g_i \sin\left(h_i\left(P_i^{\min} - P_i\right)\right)\right|, i = 1, 2, \ldots, N. \tag{8}$$

where a_i, b_i, c_i, g_i, and h_i are constants and $P_i^{\min}$ is the minimum allowable power generation in the ith power plant. Also, the emission value for P_i MW power generation in the ith power plant as the second term of the objective function can be written as:

$$F_{E_i}\left(P_i\right) = \alpha_i P_i^2 + \beta_i P_i + \gamma_i + \eta_i \exp\left(\delta_i P_i\right), i = 1, 2, \ldots, N. \tag{9}$$

where α_i, β_i, γ_i, η_i, and δ_i are constants.

Finally, the total objective function of EEDP that the sum of the two objective functions includes the generation cost and emission amount for N power plants is obtained as the following,

$$F = \sum_{i=1}^{N} \left(F_{C_i} + p * F_{E_i}\right), \tag{10}$$

where p is the penalty coefficient to convert a multi-objective optimization into a single-objective optimization problem, and how to calculate it is stated in [79].

5.1.1.2 Constraints EEDP constraints that are considered are divided into two parts, which are described below.

(1) Generation constraints

The power generation of each power plant must be in the allowed range, which is represented as follows:

$$P_i^{\min} \le P_i \le P_i^{\max}, i = 1, 2, \ldots, N. \tag{11}$$

where $P_i^{\min}$ and $P_i^{\max}$ are the minimum and maximum allowable power generation of the ith power plant.

(2) The constraint of the balance of generation and demand

Table 15 Comparison of various methods for case I

	LBO	GA	PSO	BA	MSFLA	KKO
G_1	128.6695	128.997	128.984	128.8280	128.338	129.011
G_2	186.0869	192.683	192.645	192.5792	191.964	192.303
G_3	195.9175	190.110	190.063	190.2858	191.389	190.274
$F_c(\$)$	**25,453.99**	25,499.43	25,494.95	25,494.69	25,493.96	25,490.5
F_E(ton)	**310.0958**	311.273	311.150	311.15	311.1638	311.013
L_P	**11.6659**	11.6964	11.6919	11.6936	11.6927	11.687

Best-obtained results are shown in bold

Table 16 Comparison of various methods for case II

	LBO	SSA	KKO	LFA	FPA	SPEA2
G_1	78.966	54.2501	54.9923	54.9920	53.188	52.9761
G_2	79.999	79.0674	78.8914	78.7689	79.975	72.8130
G_3	84.539	80.9404	78.7946	87.7168	78.105	78.1128
G_4	83.035	80.6482	88.7479	78.1055	97.119	83.6088
G_5	134.953	159.6807	159.814	140.6272	152.74	137.2432
G_6	153.449	239.7595	160.555	157.0936	163.08	172.9188
G_7	296.7148	293.6209	262.174	299.9954	258.61	287.2023
G_8	314.644	299.3002	308.857	309.2219	302.22	326.4023
G_9	427.706	394.5042	430.307	439.3243	433.21	448.8814
G_{10}	430.124	397.5986	461.039	438.6947	466.07	423.9025
$F_c(\times 10^5\$)$	**1.13082**	1.16199	1.13481	1.13246	1.1337	1.1352
F_E(ton)	4114.90	**3922.6781**	3982.85	4139.89	3997.7	4109
L_P	**84.13**	–	84.17	84.37	84.3	–

Best-obtained results are shown in bold

Another constraint that must be considered while generating power is the balance of generation and demand. Since power losses are always associated with the transmission, it is necessary to consider the power losses in this constraint. Therefore, this constraint can be written as follows:

$$\sum_{i=1}^{N} P_i = D + L_P \tag{12}$$

where D is the total power demand and L_P is the power losses during the transmission. The total power losses L_P can be calculated as follows:

$$L_P = \sum_{i=1}^{N} \sum_{j=1}^{N} P_i B_{ij} P_j \tag{13}$$

where B_{ij} as power losses coefficient matrix is a constant matrix.

Table 17 Comparison of various methods for case III

	LBO	SSA	KKO	MOMVO	MABC	FPA	MOSSA
G_1	113.998	114	114	113.888	114	43.405	110.7491
G_2	113.999	114	113.045	113.604	114	113.95	111.0225
G_3	119.952	120	119.744	118.1822	120	105.86	97.7975
G_4	167.663	169.6615	181.102	179.3608	169.368	169.65	176.4569
G_5	96.999	97	96.5081	97	97	96.659	86.08027
G_6	123.007	124.1793	139.796	139.7361	124.2571	139.02	107.3728
G_7	296.755	299.6333	299.686	299.1992	299.7111	273.28	256.9853
G_8	293.581	297.9084	298.619	287.2614	297.9148	285.17	284.4698
G_9	294.697	297.1041	289.447	293.1661	297.2603	241.96	288.4908
G_{10}	120.582	130	131.386	201.1565	130	131.26	130
G_{11}	296.957	298.4902	247.114	245.8292	298.4103	312.13	235.5396
G_{12}	295.103	298.0485	318.381	246.0955	298.0263	362.58	244.8612
G_{13}	428.375	433.6989	395.689	397.0952	433.5562	346.24	394.7111
G_{14}	417.1849	421.7318	393.82	395.2555	421.7284	306.06	394.4515
G_{15}	415.909	422.8917	305.891	394.5758	422.78	358.78	394.935
G_{16}	417.956	422.7761	394.283	394.92	422.7802	260.68	393.6546
G_{17}	435.428	439.4818	489.706	488.8086	439.4119	415.19	466.4393
G_{18}	435.180	439.3167	487.897	488.5656	439.4031	423.94	417.0333
G_{19}	435.061	439.4325	500.104	423.0749	439.4133	549.12	506.681
G_{20}	436.139	439.3283	455.719	426.6439	439.4134	496.7	458.2062
G_{21}	436.014	439.5036	434.334	437.0451	439.4467	539.17	495.378
G_{22}	435.443	439.5325	434.86	440.1489	439.4469	546.46	517.2692
G_{23}	436.685	439.8736	446.6	514.0343	439.7724	540.06	498.5927
G_{24}	436.263	439.3167	451	444.3748	439.7716	514.5	477.2586
G_{25}	436.545	440.2088	491.259	434.8894	440.1118	453.46	461.2753
G_{26}	437.098	440.2306	435.771	437.2822	440.1113	517.31	436.3836
G_{27}	26.820	28.8355	11.079	13.3596	28.9933	14.881	10.30714
G_{28}	26.426	28.9969	10.3466	10.2467	28.9937	18.79	10.44341
G_{29}	26.865	28.8005	12.2337	12.4183	28.9939	26.611	10.66527
G_{30}	96.999	97	96.6001	95.4617	97	59.581	97.33416
G_{31}	171.897	172.3405	189.436	189.2573	172.3318	183.48	162.7609
G_{32}	170.865	172.3671	175.188	187.1736	172.3316	183.39	176.2818
G_{33}	170.653	172.2762	189.992	188.4748	172.3319	189.02	170.2161
G_{34}	199.999	200	199.679	199.8411	200	198.73	199.1574
G_{35}	199.999	200	199.89	199.5351	200	198.77	199.3051
G_{36}	302.519	200	199.905	200	200	182.23	199.1051
G_{37}	100.046	100.8786	108.554	107.1915	100.8384	39.673	106.7655
G_{38}	99.583	100.6951	109.71	109.6094	100.8384	81.596	101.9738
G_{39}	99.538	100.7199	108.639	110	100.8386	42.96	105.6719
G_{40}	435.197	439.3512	421.912	426.2376	439.4127	537.17	508.0384
$F_c(\times10^5\$)$	1.28937	1.29996	1.25852	1.2547	1.29995	**1.2317**	1.241695
$E_c(\times10^5\ \text{ton})$	**1.69348**	1.76652	2.10837	2.0991	1.76682	2.0846	2.352652

5.1.2 EEDP results using LBO

To comprehensively evaluate the performance of LBO for EEDP, the proposed algorithm is implemented for the three small, medium, and large grids, which are IEEE standard three, ten, and forty buses, respectively. In each of the three problems, the obtained optimal power generation value, the value of each objective function, and the transmission losses are presented and compared with similar methods. The characteristics of the three EEDPs, including the number of the power plant and the power demand, are given in Table 14.

By optimizing the generation of each grid introduced in Table 14, the optimal generation value of each power plant is calculated. The optimal power generation values in each grid cover the power demand and transmission losses. At the same time, the intended objective functions are optimized. Tables 15, 16, and 17 related to the grid include three, ten, and forty power plants, respectively, which show the optimal values for the power plants, the objective functions, and the power losses.

The obtained results from LBO and five other algorithms include GA [80], PSO [80], Bat Algorithm (BA) [83], Modified Shuffled Frog Leaping Algorithm (MSFLA) [84], and Kho-Kho Optimization (KKO) [78] are presented for comparison in Table 15. As can be seen from the results, all three values of generation costs, emissions, and power losses have been improved in the proposed algorithm compared to other algorithms. The results of the LBO and several of the most important algorithms that have recently studied on this grid are compared in Table 16. The compared algorithms with LBO results in the ten-bus grid are Squirrel Search Algorithm (SSA) [85], KKO [78], Lightning Flash Algorithm (LFA) [86], Flower Pollination Algorithm (FPA) [79], and Strength Pareto Evolutionary Algorithm 2 (SPEA2) [81]. According to this comparison, the total generation cost and the power losses have been improved in LBO. However, it has not been very successful in reducing emissions. Finally, the obtained results for the 40-bus grid using LBO and several other algorithms, including SSA [85], KKO [78], Multi-Objective Multi-Verse Optimization (MOMVO) [87], Modified Artificial Bee Colony (MABC) algorithm [88], FPA [79], and Multi-Objective Squirrel Search Algorithm (MOSSA) [82] are also presented in Table 17. According to the results for the 40-bus grid, emissions have dropped significantly. The generation costs in LBO are not the best result among the existing algorithms but are still comparable. In addition, according to the IEEE standard, the power losses have not been considered for this grid. The optimal value for the production of power plants has been determined by the LBO, and the generation costs, the emissions, and the power losses obtained by LBO are the lowest values among the others. The most important reason is to be clear about the value of the penalty coefficient; all methods considered this value same. The penalty coefficient of the first grid is expressed in several references. But in the other two grids, the exact value of the penalty coefficient is not stated, and different values may be considered in researches. This can be deduced from the results. However, by analyzing the results in the ten-bus and forty-bus grids, the superiority of the LBO method is clear. Also, the minimum value of transmission losses (L_P) in the examined grids is determined by LBO.

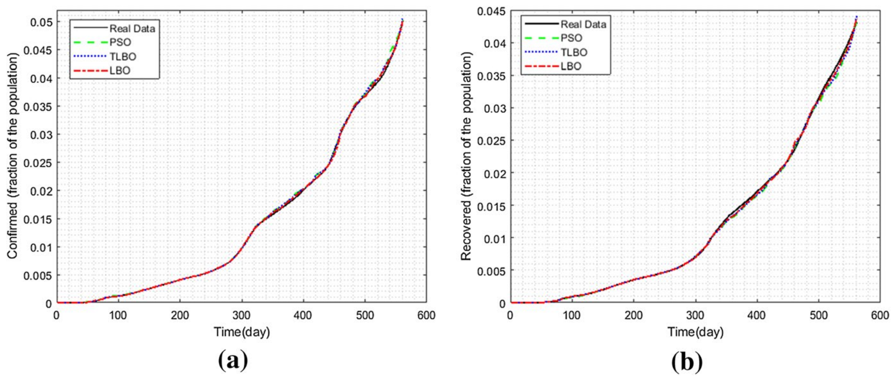


Fig. 8 Compression real and prediction of confirmed and recovered data of Iran by top-performing and LBO algorithms, **a** confirmed and **b** recovered individuals

5.2 Prediction of the COVID-19 pandemic

Despite the fact that the proposed optimization algorithm succeeded to optimize several benchmark functions in Sect. 4, the evaluation of the algorithm is essential for real-world problems. Therefore, the mathematical model of Covid-19 is expressed in this section, and the prediction of this pandemic will be evaluated by LBO. The prediction of the Covid-19 pandemic model is vital to recognize and control the pandemic course. The accurate prediction helps authorities employ appropriate strategies to prevent the spread of this infection, such as vaccination and social distance.

5.2.1 Mathematical modeling of Covid-19

There is plenty of mathematical models for Covid-19 that have been recently proposed. Among them, however, one of the most popular models will be employed in this section, named SIDARTHE mathematical model [62]. The state-space model of Covid-19 is given in (14)–(21).

$$\dot{S}(t) = -S(t)(\alpha I(t) + \beta D(t) + \gamma A(t) + \delta R(t)) \tag{14}$$

$$\dot{I}(t) = S(t)(\alpha I(t) + \beta D(t) + \gamma A(t) + \delta R(t)) - (\varepsilon + \zeta + \lambda)I(t) \tag{15}$$

$$\dot{D}(t) = \varepsilon I(t) - (\eta + \rho)D(t) \tag{16}$$

$$\dot{A}(t) = \zeta I(t) - (\theta + \mu + \kappa)A(t) \tag{17}$$

$$\dot{R}(t) = \eta D(t) + \theta A(t) - (\nu + \xi)R(t) \tag{18}$$

$$\dot{T}(t) = \mu A(t) + \nu R(t) - (\sigma + \tau)T(t) \tag{19}$$

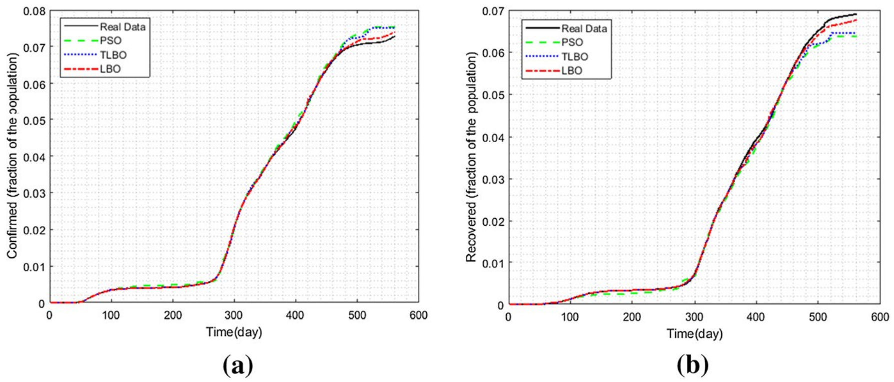


Fig. 9 Compression real and prediction of confirmed and recovered data of Italy by top-performing and LBO algorithms, **a** confirmed and **b** recovered individuals

$$\dot{H}(t) = \lambda I(t) + \rho D(t) + \kappa A(t) + \xi R(t) + \sigma T(t) \tag{20}$$

$$\dot{E}(t) = \tau T(t) \tag{21}$$

where this model consists of eight states, including susceptible ($S(t)$), infected ($I(t)$), diagnosed ($D(t)$), ailing ($A(t)$), recognized ($R(t)$), threatened ($T(t)$), healed ($H(t)$), and extinct ($E(t)$) cases. Besides, the positive parameters of model, denoted by Greek letters, are represented in NOMENCLATURE. The purpose of modeling is to determine these 16 parameters so that the modeling accuracy is maximized. Thus, LBO is used to specify the parameters of the model in the next subsection.

5.2.2 Evaluation of Covid-19 prediction

The mathematical model of Covid-19 was presented in the last subsection. LBO is utilized to specify the parameters of the mentioned model in the following. In this problem, the mathematical modeling has been changed to an optimization problem and Mean Square Error (MSE) is used as the cost function. Besides, the real data used in this modeling are available on https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases. The mathematical modeling has been done for two different countries, including Iran and Italy. In this regard, the confirmed and recovered individuals' data are predicted for 561 days, from 22 January 2020 to 4 August 2021. Mathematical modeling must change over time due to changing the behaviors of society against the pandemic. Therefore, all data are divided into several 20-day periods, tuning the model is being done only by the first 15 days, and then the model is being evaluated by the last five days of each 20-day period. The results of the confirmed and recovered individuals' prediction are shown in Figs. 8 and 9 for Iran and Italy, respectively. It should be noted that the proportion of confirmed and recovered individuals' number to the total population of the country are considered in the prediction process, 80 million and 60 million population for Iran and Italy in turn. Obviously, the mortality rate is considerably affected by patients' age and the improvement of treatment methods. Thus, the death rate has not been considered in the

Table 18 Estimated parameters of SIDARTHE mathematical local model for Iran's data by LBO algorithm

	11 Feb. 2020	02 Mar. 2020	22 Mar. 2020	11 Apr. 2020	01 May 2020	21 May 2020	10 Jun 2020	30 Jun 2020	20 Jul. 2020	09 Aug. 2020	29 Aug. 2020	18 Sep. 2020	08 Oct. 2020	28 Oct. 2020
α	2.7E−16	1.4E−06	0.7235	0.0897	2.2E−16	0.3460	0.2058	0.0108	0.0924	0.0658	2.2E−16	0.0644	0.0360	2.2E−16
β	2.9E−16	2.8E−16	1	0.2024	2.2E−16	0.7345	0.1526	0.4631	0.1772	0.5277	2.2E−16	0.1585	2.2E−16	0.1867
γ	2.2E−16	2.3E−16	0.4134	0.0707	0.6031	0.3766	0.1568	0.4364	0.4859	1	0.5922	0.3342	0.4156	0.9291
δ	2.3E−16	0.2357	0.0224	0.4933	0.1243	0.3966	0.6610	0.0553	0.5223	0.2718	0.2769	0.0084	0.1996	2.2E−16
ε	2.6E−16	2.8E−16	0.0160	0.6127	0.1925	2.2E−16	0.6504	0.3631	0.2430	0.6766	0.5643	2.2E−16	0.4772	0.2667
θ	0.4881	0.1188	0.8453	0.2061	0.1340	0.6712	0.0465	0.6835	0.0169	0.4758	2.2E−16	0.1458	0.5936	0.0878
ζ	2.6E−16	0.0586	0.8134	2.2E−16	0.3495	0.8306	2.2E−16	0.3141	0.3388	1	0.3978	0.2770	0.1802	0.4103
η	0.0088	2.4E−16	1.001	0.2536	0.237	0.4566	0.9181	0.0757	0.4131	0.1547	0.0560	0.0981	0.1765	0.0419
μ	2.2E−16	0.0065	0.6694	2.2E−16	0.0091	2.2E−16	0.8232	0.5039	0.1056	0.0452	0.3094	2.2E−16	0.9677	0.5046
ν	2.5E−16	2.2E−16	1.250	0.1914	1	0.2581	0.4109	0.7649	0.1341	0.1752	2.2E−16	0.0302	2.2E−16	0.0012
τ	0.0325	2.2E−16	1.036	2.2E−16	2.2E−16	0.7191	0.5041	2.2E−16	2.2E−16	0.4020	0.7553	2.2E−16	0.3663	2.2E−16
λ	2.6E−16	2.3E−16	0.5225	2.2E−16	1	0.6530	0.2232	0.2305	0.3053	0.8168	0.2063	0.0751	0.6328	0.3341
κ	2.2E−16	0.0124	2.75E−	2.2E−16	2.2E−16	0.2339	0.5434	0.5815	0.5375	0.3015	0.3073	0.0725	0.4006	0.1396
ξ	2.2E−16	0.0141	0.0563	0.0162	2.2E−16	2.2E−16	2.2E−16	0.2273	2.2E−16	0.0005	2.2E−16	0.1933	2.2E−16	0.0068
ρ	2.7E−16	0.1450	1.001	0.3667	0.0194	0.1203	0.4505	0.0902	0.6075	0.9841	0.3453	2.2E−16	0.0333	2.2E−16
σ	2.6E−16	0.0021	2.22E−	2.2E−16	0.1628	0.4678	0.8191	0.4698	0.2350	1	0.5177	0.5950	2.2E−16	0.3194

	17 Nov 2020	07 Dec. 2020	27 Dec. 2020	16 Jan. 2021	05 Feb. 2021	25 Feb. 2021	17 Mar. 2021	06 Apr. 2021	26 Apr. 2021	16 May 2021	05 Jun 2021	25 Jun 2021	15 Jul. 2021	04 Aug. 2021
α	0.3916	0.0810	2.2E−16	0.2644	0.6827	1.006	0.8907	0.6999	0.629	0.3864	0.4052	0.8902	0.9020	1
β	2.2E−16	0.1153	2.2E−16	0.8681	1.0020	1.032	0.8932	0.780	0.2753	2.2E−16	0.2713	0.3440	0.300	0.4134
γ	2.2E−16	0.0033	0.6689	0.4388	1.0001	1.23	2.8E−16	1	2.2E−16	0.4710	0.4957	0.2350	0.0253	0.0224
δ	0.0815	0.4577	0.2901	0.4865	0.9965	0.6344	1	0.4134	0.7742	0.9378	2.2E−16	0.9847	2.8E−7	0.0160
ε	0.1716	0.6877	0.2320	0.6113	2.7E−16	2.7E−16	2.5E−16	0.1224	0.329	1	0.3836	2.2E−16	0.6242	0.8453
θ	0.4498	0.2153	2.2E−16	2.2E−16	0.5870	0.680	0.8720	0.010	2.2E−16	0.6735	0.5554	0.0372	2.2E−16	2.2E−16

Table 18 (continued)

	17 Nov 2020	07 Dec. 2020	27 Dec. 2020	16 Jan. 2021	05 Feb. 2021	25 Feb. 2021	17 Mar. 2021	06 Apr. 2021	26 Apr. 2021	16 May 2021	05 Jun 2021	25 Jun 2021	15 Jul. 2021	04 Aug. 2021
ζ	0.1904	2.2E−16	0.3495	0.0013	2.2E−16	2.2E−16	2.2E−16	0.8443	1	0.7170	0.0403	0.9528	0.0289	1
η	0.2041	0.4360	1	0.2795	0.2321	2.2E−16	1.001	1.069	0.7408	0.2204	0.1081	0.6959	1	0.096
μ	2.2E−16	0.2265	2.2E−16	0.0787	1	2.2E−16	1	1	1	2.2E−16	0.5237	2.2E−16	2.2E−16	0.6404
ν	0.3711	0.329	0.1263	0.4050	0.1049	0.7997	2.3E−16	0.3399	0.329	0.5419	0.1591	0.6368	1.0001	1.417
τ	2.2E−16	0.2253	0.1056	2.2E−16	0.2072	2.2E−16	2.2E−16	1	0.2253	2.2E−16	2.2E−16	2.2E−16	2.2E−16	0.3402
λ	0.0084	2.2E−16	2.2E−16	0.9345	0.693	0.6216	0.1224	0.4254	2.2E−16	0.3409	0.1808	0.9749	1.4100	1.3621
κ	0.3116	0.6242	0.4942	0.7576	1	0.1224	0.410	0.547	0.6242	0.8241	0.3656	1	0.5110	0.6119
ξ	2.2E−16	0.0870	0.6383	0.7124	0.6841	0.010	0.8213	0.6870	1	0.1630	0.3213	0.1841	2.2E−16	1.001
ρ	0.4547	0.2426	0.0163	0.2607	0.3620	0.8443	2.9E−16	0.8036	0.0035	2.2E−16	0.6076	0.1224	0.8459	2.2E−16
σ	0.0001	0.016	0.5888	0.6985	1	1.069	1	0.9329	2.7E−16	0.0481	0.1116	0.740	0.9799	1.230

Table 19 Estimated parameters of SIDARTHE mathematical local model for Italy's data by LBO algorithm

	11 Feb. 2020	02 Mar. 2020	22 Mar. 2020	11 Apr. 2020	01 May 2020	21 May 2020	10 Jun 2020	30 Jun 2020	20 Jul. 2020	09 Aug. 2020	29 Aug. 2020	18 Sep. 2020	08 Oct. 2020	28 Oct. 2020
α	2.2E−16	2.3E−16	0.4341	2.6E−16	2.2E−16	2.2E−16	0.0096	0.0066	0.2373	0.0006	0.1207	0.0544	2.2E−16	0.1284
β	2.2E−16	2.2E−16	0.0078	0.1365	0.1823	0.0865	2.2E−16	0.0087	3.1E−16	0.6521	0.0988	0.0832	0.6494	0.7039
γ	2.2E−16	2.3E−16	2.2E−16	0.1824	0.0345	2.2E−16	0.0283	0.0049	0.6851	2.2E−16	0.3673	0.3506	0.2091	2.2E−16
δ	2.9E−16	0.5224	0.8430	0.0636	2.2E−16	0.0754	0.2651	1	0.0405	0.1136	0.2578	0.0067	0.5152	0.3261
ε	2.2E−16	2.2E−16	0.7170	2.7E−16	0.0246	2.2E−16	0.0048	0.2600	2.2E−16	0.0358	0.0174	0.0130	0.0126	0.0059
θ	2.2E−16	0.0814	1	0.9944	0.1532	2.2E−16	0.0891	0.4134	2.2E−16	2.2E−16	0.0522	2.2E−16	2.2E−16	0.7881
ζ	2.2E−16	0.1898	0.4925	2.6E−16	0.0049	0.0486	0.0554	0.0224	2.2E−16	0.1004	0.0451	2.6E−16	0.0699	0.0273
η	2.2E−16	2.2E−16	2.2E−16	0.0151	0.1312	2.2E−16	2.2E−16	0.0160	0.2864	2.2E−16	0.3142	0.7721	2.2E−16	0.1950
μ	2.2E−16	5.3E−06	0.0608	0.0369	0.0756	0.0893	0.6067	0.8453	3.1E−16	0.1901	0.2877	2.2E−16	0.8340	0.6702
ν	0.5836	0.0065	2.2E−16	2.6E−16	2.2E−16	0.5502	0.3220	0.8950	0.0638	2.3E−16	0.2512	0.2067	0.9067	0.4398
τ	0.5014	2.2E−16	6.5E−02	0.0165	2.2E−16	0.1541	0.0690	0.4990	0.5355	0.2943	0.0193	0.3127	0.3054	0.5106
λ	2.2E−16	0.0020	2.2E−16	0.0086	2.2E−16	2.2E−16	0.0212	0.0176	1	0.4230	0.0015	0.0191	3.4E−05	4.8E−06
κ	2.2E−16	0.0075	6.4E−01	0.0697	0.0795	0.7581	0.1840	0.4522	0.8853	2.2E−16	0.0691	2.2E−16	0.0669	0.1126
ξ	2.2E−16	2.2E−16	0.1099	0.0439	0.0512	2.2E−16	0.6955	0.7003	0.0173	2.3E−16	0.2359	2.2E−16	0.1278	0.5741
ρ	2.2E−16	2.2E−16	2.2E−16	2.3E−16	0.0684	2.2E−16	0.0030	0.0618	2.2E−16	0.1769	2.2E−16	0.1770	0.0532	1
σ	2.2E−16	0.0070	0.4269	0.4779	0.4303	2.2E−16	0.2114	0.1166	0.3306	0.4653	0.1667	2.2E−16	0.3077	0.5667
	17 Nov 2020	07 Dec. 2020	27 Dec. 2020	16 Jan. 2021	05 Feb. 2021	25 Feb. 2021	17 Mar. 2021	06 Apr. 2021	26 Apr. 2021	16 May 2021	05 Jun 2021	25 Jun 2021	15 Jul. 2021	04 Aug. 2021
α	0.1066	2.2E−16	0.0148	0.0379	0.3678	0.9923	0.9855	0.1245	0.1250	0.6394	0.9688	0.9951	1.022	1.0160
β	1.1E−01	0.1331	2.2E−16	0.1638	0.4967	1.0020	0.9912	0.9826	0.7044	0.0991	0.9281	0.9950	0.9988	0.9948
γ	2.2E−16	2.2E−16	0.0979	0.1576	0.4075	1.0057	1.0146	2.2E−16	0.0577	1.0055	0.6890	1.0088	1.0037	0.9848
δ	0.4575	0.8654	0.3442	0.2385	2.2E−16	1.0026	0.5647	2.2E−16	0.1202	1.0036	0.9906	0.9831	1.0006	1.0092
ε	2.2E−16	0.1102	0.2001	0.9780	0.5269	2.2E−16	0.6031	0.3414	0.2657	0.9975	0.9078	1.0005	1.6E−15	3.0E−16
θ	0.3538	0.2987	0.2613	0.0745	0.2153	1.007	0.4186	0.2216	0.0605	1.0190	0.2093	1.0009	0.9995	2.4E−16

Table 19 (continued)

	17 Nov 2020	07 Dec. 2020	27 Dec. 2020	16 Jan. 2021	05 Feb. 2021	25 Feb. 2021	17 Mar. 2021	06 Apr. 2021	26 Apr. 2021	16 May 2021	05 Jun 2021	25 Jun 2021	15 Jul. 2021	04 Aug. 2021
ζ	0.1313	2.2E−16	0.1728	0.0216	0.6507	3.2E−16	0.4856	0.3590	0.1993	1.0005	0.0007	0.9205	0.0285	3.2E−16
η	0.2200	0.3127	2.2E−16	0.1528	0.3020	1.0098	0.7565	0.2233	2.2E−16	0.9763	0.0173	0.2524	0.9921	1.0114
μ	0.2442	2.2E−16	0.0501	0.2982	2.2E−16	2.2E−16	0.2218	0.2888	0.5896	0.0733	0.8844	1.0042	2.2E−16	1.0051
ν	0.5765	0.2984	0.1870	0.0390	2.2E−16	2.4E−16	0.8562	2.2E−16	0.3014	2.5E−16	0.2477	2.9E−16	1.7E−15	0.0692
τ	0.8037	0.0745	2.2E−16	0.1645	1.0090	0.9992	0.9779	2.2E−16	2.2E−16	0.0528	0.6757	2.7E−16	2.2E−16	2.2E−16
λ	2.2E−16	2.2E−16	0.0001	4.1E−05	2.2E−16	1.0207	0.4962	0.6690	0.3597	0.8821	0.8734	0.9945	0.9879	0.9935
κ	0.1561	0.1321	0.0621	0.1001	0.3299	0.8729	0.8949	0.3408	0.8248	1.0022	2.2E−16	0.9974	1.0025	1.0012
ξ	0.1036	0.0580	2.2E−16	0.0663	0.9089	1.0008	1.0113	0.2691	0.4545	0.6473	2.2E−16	1.0007	2.4E−15	2.3E−16
ρ	0.2706	0.8357	0.2203	0.3604	0.6099	1.0112	0.9967	2.2E−16	0.2643	0.9963	0.9349	1.0001	0.0289	3.3E−16
σ	2.2E−16	0.0739	0.6586	2.2E−16	0.7642	1.0052	1.0032	0.2210	0.5306	1.0057	0.7801	0.9949	0.6293	1.0180

Table 20 MAE and MSE values of Iran and Italy's confirmed and recovered data prediction by three algorithms

	LBO	TLBO	PSO
Iran			
MAE	1.3065E−04	1.6921E−04	2.0277E−04
MSE	5.1065E−08	9.2653E−08	1.3633E−07
Italy			
MAE	3.0045E−04	6.1144E−04	1.0236E−03
MSE	3.0498E−07	1.7024E−06	2.6902E−06

prediction process. In these figures, the results of the first 15 days of each 20-day period are related to modeling output and the last 5 days results are the results of the prediction based on the obtained model. Additionally, the results of PSO and TLBO algorithms have been compared with LBO algorithm.

Apparently, the three optimization algorithms in predicting the confirmed and recovered individuals of Iran show approximately the same performance. However, the performance of LBO in forecasting of Italy's data is dramatically better than other two algorithms, especially in the last parts of the prediction. The estimated parameter of the model by LBO is expressed in Table 18 for Iran and Table 19 for Italy. As mentioned earlier, the modeling has been conducted every 20 days. Therefore, 28 separated models are estimated for this data.

After modeling the Covid-19 in Iran and Italy based on the first 15 days of each 20-day period, the number of the confirmed and recovered individuals were predicted by obtained model. Then, by comparing the real data and estimated data, Mean Absolute Error (MAE) and Mean Square Error (MSE) have been calculated for Iran's and Italy's data for three algorithms which are reported in Table 20. According to this table, LBO showed considerable better performance than others. In other words, according to the population size of both countries, the LBO algorithm illustrates less error than the TLBO algorithm, as the second algorithm, which is more than 3000 and 1800 individuals per day for Iran's and Italy's data, respectively.

6 Conclusion

In this paper, a novel optimization algorithm called Ladybug Beetle Optimization (LBO) algorithm is proposed aimed utilizing at engineering applications, which is inspired by the behavior of ladybugs in winter to find a warm location. At the beginning of winter, the swarm of ladybugs searches coordinately for a warm location, and their movements are affected by the other ladybugs. Also, while searching, some ladybugs may deviate and be annihilated due to extreme cold. The objective function is considered to be the inverse value of heat for each ladybug location. Therefore, the optimization problem is formulated as a standard minimization problem. Firstly, the initial positions of the population have been selected randomly in the algorithm. Then, according to the other ladybugs, especially better members of the ladybug population, the position of each ladybug is updated. Some ladybugs update randomly to make a balance between the exploitation and exploration aspects of the algorithm. Finally, the positions of the population are evaluated to sort them and ignore the worst members in each iteration. To evaluate the performance of LBO, the proposed algorithm was employed on 78 well-known benchmark functions, including the unimodal, multimodal, and CEC-C06 2019 functions, and the results were analyzed. LBO results were also compared with several high-performance metaheuristic optimization algorithms, which shows LBO succeeded to achieve best performance in comparison among other algorithms in most of the benchmark functions. The proposed algorithm has reached the optimal values of 70.5% of the used benchmark functions and is the only algorithm that achieved the best solution of 21.8% of them. Besides, LBO was used for the economic-environmental dispatch problem (EEDP) as a real-world and multi-objective optimization problem. The results obtained from all three power grids in solving the EEDP including three, ten, and forty busses were so promising. So, most of the results obtained from LBO were better than other algorithms. Compared to other algorithms, LBO won first place in the generation costs and emission in the three-bus grid, the generation costs in the ten-bus grid, and the emission in the forty-bus grid. Also, according to the results, LBO had the lowest power losses in the grids. Additionally, LBO was used for the prediction of the COVID-19 model as a real-world problem. According to the results, LBO indicated better performance with less error compared to top-performing algorithms. Consequently, the results show that the novel proposed LBO has the acceptable performance in the challenging benchmark functions as well as the real-world problems. However, the proposed algorithm is not able to optimize the fifth group of benchmark functions, which means the algorithm needs some modifications in some aspects. Overall, the proposed method is a high-performance algorithm, which is able to optimize a variety of optimization problems. Therefore, it is predicted that the multi-objective version of LBO would be also a powerful algorithm against multi-objective optimization problems. Thus, LBO optimizer will be modified as a multi-objective algorithm in future research.

Data availability The MATLAB and python source code of the LBO algorithm that support the findings of this study are available in https://github.com/Saadat-Safiri/LBO-algorithm-matlab-code and https://github.com/Saadat-Safiri/LBO-algorithm-python-code, respectively.

Declarations

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

References

1. Yang X-S, He X (2016) Nature-inspired optimization algorithms in engineering: overview and applications. Nat-Inspired Comput Eng:1–20
2. He S, Wu QH, Saunders JR (2009) Group search optimizer: an optimization algorithm inspired by animal searching behavior. IEEE Trans Evol Comput 13(5):973–990
3. Zhang X, Wen S (2021) Hybrid whale optimization algorithm with gathering strategies for high-dimensional problems. Expert Syst Appl 179:115032
4. Saafan MM, El-Gendy EM (2021) IWOSSA: an improved whale optimization salp swarm algorithm for solving optimization problems. Expert Syst Appl 176:114901

5. Mostafa Bozorgi S, Yazdani S (2019) IWOA: an improved whale optimization algorithm for optimization problems. J Comput Des Eng 6(3):243–259
6. Deng W, Xu J, Zhao H (2019) An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem. IEEE Access 7:20281–20292
7. Abd Elaziz M, Attiya I (2021) An improved Henry gas solubility optimization algorithm for task scheduling in cloud computing. Artif Intell Rev 54(5):3599–3637
8. Adnan RM, Mostafa RR, Kisi O, Yaseen ZM, Shahid S, Zounemat-Kermani M (2021) Improving streamflow prediction using a new hybrid ELM model combined with hybrid particle swarm optimization and grey wolf optimization. Knowl-Based Syst 230:107379
9. Miranda V, Alves R (2013) Differential evolutionary particle swarm optimization (deepso): a successful hybrid. In: 2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence, IEEE, pp 368–374
10. Nasir M, Sadollah A, Aydilek İB, Ara AL, Nabavi-Niaki SA (2021) A combination of FA and SRPSO algorithm for combined heat and power economic dispatch. Appl Soft Comput 102:107088
11. Kıran MS, Gündüz M, Baykan ÖK (2012) A novel hybrid algorithm based on particle swarm and ant colony optimization for finding the global minimum. Appl Math Comput 219(4):1515–1521
12. Chou J-S, Truong D-N (2021) A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean. Appl Math Comput 389:125535
13. Askari Q, Younas I, Saeed M (2020) Political Optimizer: a novel socio-inspired meta-heuristic for global optimization. Knowl-Based Syst 195:105709
14. Zervoudakis K, Tsafarakis S (2020) A mayfly optimization algorithm. Comput Ind Eng 145:106559
15. Kaveh A, Zaerreza A (2020) Shuffled shepherd optimization method: a new meta-heuristic algorithm. Eng Comput
16. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks. IEEE, vol 4, pp 1942–1948
17. Yang X-S, Deb S (2009) Cuckoo search via Lévy flights. In: 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC). IEEE, pp 210–214
18. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67
19. Pozna C, Precup R-E, Horvath E, Petriu EM (2022) Hybrid Particle filter-particle swarm optimization algorithm and application to fuzzy controlled servo systems. IEEE Trans Fuzzy Syst
20. Xie L, Han T, Zhou H, Zhang Z-R, Han B, Tang A (2021) Tuna swarm optimization: a novel swarm-based metaheuristic algorithm for global optimization. Comput Intell Neurosci, vol 2021
21. Al-Khateeb B, Ahmed K, Mahmood M, Le D-N (2021) Rock hyraxes swarm optimization: a new nature-inspired metaheuristic optimization algorithm. Comput Mater Continua 68(1):643–654
22. Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–73
23. Lampinen J, Storn R (2004) Differential evolution. In: New optimization techniques in engineering. Springer, pp 123–166
24. Hashim FA, Hussien AG (2022) Snake Optimizer: a novel meta-heuristic optimization algorithm. Knowl-Based Syst 242:108320
25. Kaveh A, Bakhshpoori T (2016) Water evaporation optimization: a novel physically inspired optimization algorithm. Comput Struct 167:69–85
26. Eskandar H, Sadollah A, Bahreininejad A, Hamdi M (2012) Water cycle algorithm–a novel metaheuristic optimization method for solving constrained engineering optimization problems. Comput Struct 110:151–166
27. Faramarzi A, Heidarinejad M, Stephens B, Mirjalili S (2020) Equilibrium optimizer: a novel optimization algorithm. Knowl-Based Syst 191:105190
28. Tahani M, Babayan N (2019) Flow Regime Algorithm (FRA): a physics-based meta-heuristics algorithm. Knowl Inf Syst 60(2):1001–1038
29. Rao RV, Savsani VJ, Vakharia D (2011) Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. Comput Aided Des 43(3):303–315
30. Askari Q, Saeed M, Younas I (2020) Heap-based optimizer inspired by corporate rank hierarchy for global optimization. Expert Syst Appl 161:113702
31. Chou J-S, Nguyen N-M (2020) FBI inspired meta-optimization. Appl Soft Comput 93:106339
32. Moosavian N, Roodsari BK (2014) Soccer league competition algorithm: a novel meta-heuristic algorithm for optimal design of water distribution networks. Swarm Evol Comput 17:14–24
33. Dhiman G et al (2021) MOSOA: a new multi-objective seagull optimization algorithm. Expert Syst Appl 167:114150

34. Hashim FA, Hussain K, Houssein EH, Mabrouk MS, Al-Atabany W (2021) Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems. Appl Intell 51(3):1531–1551
35. Khishe M, Mosavi MR (2020) Chimp optimization algorithm. Expert Syst Appl 149:113338
36. Shabani A, Asgarian B, Salido M, Gharebaghi SA (2020) Search and rescue optimization algorithm: a new optimization method for solving constrained engineering optimization problems. Expert Syst Appl 161:113698
37. Fouad MM, El-Desouky AI, Al-Hajj R, El-Kenawy E-SM (2020) Dynamic group-based cooperative optimization algorithm. IEEE Access 8:148378–148403
38. Cui Z et al (2019) A pigeon-inspired optimization algorithm for many-objective optimization problems. Sci China Inf Sci 62(7):70212:1-70212:3
39. Harifi S, Khalilian M, Mohammadzadeh J, Ebrahimnejad S (2019) Emperor Penguins Colony: a new metaheuristic algorithm for optimization. Evol Intel 12(2):211–226
40. Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H (2019) Harris hawks optimization: algorithm and applications. Futur Gener Comput Syst 97:849–872
41. Wang H et al (2019) Heterogeneous pigeon-inspired optimization. Sci China Inf Sci 62(7):1–9
42. Shadravan S, Naji H, Bardsiri VK (2019) The Sailfish Optimizer: a novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems. Eng Appl Artif Intell 80:20–34
43. Shayanfar H, Gharehchopogh FS (2018) Farmland fertility: a new metaheuristic algorithm for solving continuous optimization problems. Appl Soft Comput 71:728–746
44. Malik H, Iqbal A, Joshi P, Agrawal S, Bakhsh FI (2021) Metaheuristic and evolutionary computation: algorithms and applications. Springer
45. Gandomi AH, Yang X-S, Talatahari S, Alavi AH (2013) Metaheuristic applications in structures and infrastructures. Newnes
46. Kaveh A (2017) Applications of metaheuristic optimization algorithms in civil engineering. Springer
47. Yang X-S (2010) Engineering optimization: an introduction with metaheuristic applications. Wiley
48. Osiogo F et al (2021) COVID-19 pandemic: demographic and clinical correlates of disturbed sleep among 6041 Canadians. Int J Psychiatry Clin Pract 25(2):164–171
49. da Silva RG, Ribeiro MHDM, Mariani VC, dos Santos Coelho L (2020) Forecasting Brazilian and American COVID-19 cases based on artificial intelligence coupled with climatic exogenous variables. Chaos Solitons Fractals 139:110027
50. Panwar H, Gupta P, Siddiqui MK, Morales-Menendez R, Singh V (2020) Application of deep learning for fast detection of COVID-19 in X-Rays using nCOVnet. Chaos Solitons Fractals 138:109944
51. Garcia LP et al. (2020) Estimating underdiagnosis of covid-19 with nowcasting and machine learning: experience from Brazil. *medRxiv*
52. Colubri A et al (2019) Machine-learning prognostic models from the 2014–16 Ebola outbreak: data-harmonization challenges, validation strategies, and mHealth applications. EClinicalMedicine 11:54–64
53. Chockanathan U, DSouza AM, Abidin AZ, Schifitto G, Wismüller A (2019) Automated diagnosis of HIV-associated neurocognitive disorders using large-scale Granger causality analysis of resting-state functional MRI. Comput Biol Med 106:24–30
54. Toğaçar M, Ergen B, Cömert Z (2020) Covid-19 detection using deep learning models to exploit social mimic optimization and structured chest x-ray images using fuzzy color and stacking approaches. Comput Biol Med 121:103805
55. Shaibani MJ, Emamgholipour S, Moazeni SS (2021) Investigation of robustness of hybrid artificial neural network with artificial bee colony and firefly algorithm in predicting COVID-19 new cases: case study of Iran. Stoch Environ Res Risk Assessm:1–16
56. Khalilpourazari S, Doulabi HH, Çiftçioğlu AÖ, Weber G-W (2021) Gradient-based grey wolf optimizer with Gaussian walk: application in modelling and prediction of the COVID-19 pandemic. Expert Syst Appl 177:114920
57. Hosseini E, Ghafoor KZ, Sadiq AS, Guizani M, Emrouznejad A (2020) Covid-19 optimizer algorithm, modeling and controlling of coronavirus distribution process. IEEE J Biomed Health Inform 24(10):2765–2775
58. Ndaïrou F, Area I, Nieto JJ, Torres DF (2020) Mathematical modeling of COVID-19 transmission dynamics with a case study of Wuhan. Chaos Solitons Fractals 135:109846
59. Ivorra B, Ferrández MR, Vela-Pérez M, Ramos AM (2020) Mathematical modeling of the spread of the coronavirus disease 2019 (COVID-19) taking into account the undetected infections. The case of China. Commun Nonlinear Sci Numer Simul 88:105303
60. Bhatnagar MR (2020) COVID-19: mathematical modeling and predictions. ResearchGate 10

61. Gozalpour N, Badfar E, Nikoofard A (2021) Transmission dynamics of novel coronavirus SARS-CoV-2 among healthcare workers, a case study in Iran. Nonlinear Dyn 105(4):3749–3761
62. Giordano G et al (2020) Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy. Nat Med 26(6):855–860
63. Mckenna DD et al (2015) The beetle tree of life reveals that C oleoptera survived end-P ermian mass extinction to diversify during the C retaceous terrestrial revolution. Syst Entomol 40(4):835–880
64. Dallai R, Lino-Neto J, Dias G, Nere PH, Mercati D, Lupetti P (2018) Fine structure of the ladybird spermatozoa (Insecta, Coleoptera, Coccinellidae). Arthropod Struct Dev 47(3):286–298
65. Gordon RD (1985) The Coccinellidae (Coleoptera) of America north of Mexico. J New York Entomol Soc 93(1)
66. Vandenberg NJ (2002) 93. Coccinellidae Latreille 1807. Am Beetles 2:371–389
67. Majerus ME (2009) Ladybugs. In: Encyclopedia of insects. Elsevier, pp 547–551
68. Sarwar M (2016) Recognition of some lady beetles (Coleoptera: Coccinellidae) deadly sighted for insect and mite pests in agroecosystems. Int J Entomol Res 1(2):29–34
69. Sarwar M, Saqib SM (2010) Rearing of predatory seven spotted ladybird beetle Coccinella septempunctata L.(Coleoptera: Coccinellidae) on natural and artificial diets under laboratory conditions. Pak J Zoolo 42(1)
70. Hodek I, Honek A, Van Emden HF (2012) Ecology and behaviour of the ladybird beetles (Coccinellidae). Wiley
71. Lipowski A, Lipowska D (2012) Roulette-wheel selection via stochastic acceptance. Physica A 391(6):2193–2196
72. Price K, Awad N, Ali M, Suganthan P (2018) Problem definitions and evaluation criteria for the 100-digit challenge special session and competition on single objective numerical optimization. In: Technical Report: Nanyang Technological University
73. Abdullah JM, Ahmed T (2019) Fitness dependent optimizer: inspired by the bee swarming reproductive process. IEEE Access 7:43473–43486
74. Mirjalili S (2016) SCA: a sine cosine algorithm for solving optimization problems. Knowl-Based Syst 96:120–133
75. Wang P, Zhu Z, Huang S (2013) Seven-spot ladybird optimization: a novel and efficient metaheuristic algorithm for numerical optimization. Sci World J 2013
76. Atashpaz-Gargari E, Lucas C (2007) Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In: 2007 IEEE Congress on Evolutionary Computation. IEEE, pp 4661–4667
77. Huang B, Liu L, Zhang H, Li Y, Sun Q (2019) Distributed optimal economic dispatch for microgrids considering communication delays. IEEE Trans Syst Man Cybern Syst 49(8):1634–1642
78. Srivastava A, Das DK (2020) A new Kho-Kho optimization Algorithm: an application to solve combined emission economic dispatch and combined heat and power economic dispatch problem. Eng Appl Artif Intell 94:103763
79. Abdelaziz AY, Ali ES, Abd Elazim S (2016) Implementation of flower pollination algorithm for solving economic load dispatch and combined economic emission dispatch problems in power systems. Energy 101:506–518
80. Devi AL, Krishna OV (2008) Combined economic and emission dispatch using evolutionary algorithms-a case study. ARPN J Eng Appl Sci 3(6):28–35
81. Basu M (2011) Economic environmental dispatch using multi-objective differential evolution. Appl Soft Comput 11(2):2845–2853
82. Sakthivel V, Suman M, Sathya P (2021) Combined economic and emission power dispatch problems through multi-objective squirrel search algorithm. Appl Soft Comput 100:106950
83. Gherbi YA, Bouzeboudja H, Gherbi FZ (2016) The combined economic environmental dispatch using new hybrid metaheuristic. Energy 115:468–477
84. Elattar EE (2019) Environmental economic dispatch with heat optimization in the presence of renewable energy based on modified shuffle frog leaping algorithm. Energy 171:256–269
85. Ponnuvel SV, Murugesan S, Duraisamy SP (2020) Multi-objective squirrel search algorithm to solve economic environmental power dispatch problems. Int Trans Electr Energy Syst 30(12):e12635
86. Kheshti M, Kang X, Bie Z, Jiao Z, Wang X (2017) An effective lightning flash algorithm solution to large scale non-convex economic dispatch with valve-point and multiple fuel options on generation units. Energy 129:1–15
87. Sundaram A (2020) Multiobjective multi-verse optimization algorithm to solve combined economic, heat and power emission dispatch problems. Appl Soft Comput 91:106195

88. Secui DC (2015) A new modified artificial bee colony algorithm for the economic dispatch problem. Energy Convers Manage 89:43–62