Hybrid Leader Based Optimization: A New Stochastic Optimization Algorithm for Solving Optimization Applications

Pavel Trojovsky (pavel.trojovsky@uhk.cz)
University of Hradec Králové
Mohammad Dehghani
University of Hradec Králové

Research Article

Keywords:

Posted Date: February 16th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1312581/v1

Hybrid Leader Based Optimization: A New Stochastic Optimization Algorithm for Solving Optimization Applications

Mohammad Dehghani[1] **and Pavel Trojovský**[1,*]

[1]Department of Mathematics, Faculty of Science, University of Hradec Králové, Hradec Králové, Czech Republic
[*]pavel.trojovsky@uhk.cz

ABSTRACT

In this paper, a new optimization algorithm called Hybrid Leader-Based Optimization (HLBO) is introduced that is applicable in optimization challenges. The main idea of HLBO is to guide the algorithm population under the guidance of a hybrid leader. The stages of HLBO are modeled mathematically in two phases of exploration and exploitation. The efficiency of HLBO in optimization is tested by finding solutions to twenty-three standard benchmark functions of different types of unimodal and multimodal. The optimization results of unimodal functions indicate the high exploitation ability of HLBO in local search for better convergence to global optimal, while the optimization results of multimodal functions show the high exploration ability of HLBO in global search to accurately scan different areas of search space. The quality of the results obtained from HLBO is compared with the results of eight well-known algorithms.
The simulation results show the superiority of HLBO in convergence to the global solution as well as the passage of optimally localized areas of the search space compared to eight competing algorithms. In addition, the implementation of HLBO on four engineering design issues demonstrates the applicability of HLBO in real-world problem solving.

Introduction

Advances in science and technology have led to the emergence of new optimization challenges as well as the complexity of optimization problems. These cases indicate the need and importance of optimization with efficient tools to achieve optimal solutions. An optimization problem is identified and modeled with three main parts: decision variables, constraints, and objective function[1]. The goal in optimization is to achieve the best solution with respect to the constraints of the problem, among all solutions defined for an optimization problem[2]. Problem solving techniques in optimization applications fall into two groups of deterministic methods and stochastic methods. Deterministic methods using derivative information have acceptable performance in linear and convex spaces. However, these methods are incapable of dealing with high dimension and constraint problems, complex objective functions, nonlinear and non-convex spaces. Stochastic methods, by employing random operators and random scanning of the search space away from the difficulties of deterministic methods, have the ability to provide acceptable solutions to optimization problems. Simplicity in understanding, ease of implementation, no need for derivative information, the ability to cross local optimal areas, applicability in nonlinear, and non-convex spaces are some of the advantages that have led to the popularity and pervasiveness of random methods. Optimization algorithms are one of the most popular techniques in the stochastic approach to optimizing the problem[3]. How to achieve the solution in optimization algorithms begins with generating a certain number of candidate solutions (equal to the population of the algorithm). Evaluating the objective function of the problem based on these candidate solutions determines the quality of each solution. Using this information and the algorithm steps, these candidate solutions are improved in an iterative process. Once the algorithm is fully implemented, the best candidate solution that provides a better value for the objective function compared to other candidate solutions is identified. Given the fact that every optimization problem has a basic solution called global optimal, the point made in optimization studies is that optimization algorithms do not guarantee that they can achieve exactly the global optimal solution. Therefore, quasi-optimal is the name given to the solutions obtained from the optimization algorithms[4]. Efforts to reduce the differences between quasi-optimal solutions and global optimal solutions to find better solutions have paved the way for the design and development of numerous optimization algorithms.

Exploration and exploitation are capabilities that enable optimization algorithms to be efficient in finding solutions. Exploration is the ability to search globally in different areas of the search space while exploitation is the ability to search locally near the solutions obtained because there may be better solutions near those solutions. Balancing exploration and exploitation play a key role in the success of optimization algorithms in achieving optimal solutions[5]. The main research question in the study of optimization algorithms is whether there is still a need to introduce new optimization algorithms despite the fact that countless algorithms have been introduced so far. The No Free Lunch (NFL) theorem[6] answers this question. The concept of the NFL theorem explains that there is no guarantee that an algorithm with optimal performance in solving a set of objective functions and problems will be able to perform the same performance in all optimization applications. It is not possible to ensure that a particular algorithm is the best optimizer in all optimization topics. The NFL theorem encourages researchers to develop new algorithms to find better solutions to optimization problems. The NFL theorem has motivated researchers in this paper to develop a new optimization algorithm for optimization applications.

Innovation and scientific contribution of this study is in introducing and designing a new evolutionary algorithm called Hybrid Leader Optimization (HLBO). The fundamental idea in HLBO design is to guide the algorithm population based on a hybrid leader generated by three different members. The stages of HLBO are described in two phases of exploration and exploitation and are mathematically modeled. The efficiency of HLBO has been benchmarked by optimizing twenty-three objective functions of a variety of unimodal and multimodal types. To evaluate the capability of HLBO, its performance has been compared with eight well-known algorithms.

In this section and in the following section, the literature review is presented. The Hybrid Leader Optimization (HLBO) algorithm is introduced and modeled in the section "Hybrid Leader-Based Optimization". Simulation studies are included in the section "Simulation Studies and Results". The discussion of HLBO results is provided in the section "Discussion". Conclusions and several research subjects are provided for further study in the last section.

Lecture Review

Optimization algorithms are stochastic techniques to solve optimization applications that are based on the concepts of stochastic mechanisms, e.g., concretely on random methods of trial and error, modeling of natural processes, animal behavior, physical sciences, biology sciences, rules of games and other evolutionary processes[7]. The main idea applied in the design categorizes the optimization algorithms into five groups: evolutionary-based, swarm-based, physics-based, game-based, and human-based optimization algorithms.

Evolutionary-based algorithms have been developed using the concept of natural selection, the concepts of biological and genetic sciences, and random operators such as selection, crossover, and mutation. Genetic Algorithm (GA)[8] and Differential Evolutionary (DE)[9] are the most significant evolutionary algorithms whose main inspiration is modeling of the reproductive process. Simulation of the human immune system against diseases has paved the way for the design of an Artificial Immune System (AIS) algorithm[10].

Swarm-based algorithms are inspired by the behaviors and strategies of animals, insects, birds, and other swarming activities in nature. The most widely used and famous techniques of this group are Particle Swarm Optimization (PSO)[11], Ant Colony Optimization (ACO)[12], Artificial Bee Colony (ABC)[13], Firefly Algorithm (FA)[14]. The strategy of birds and fish in finding food sources using individual and collective information has been the basic inspiration in designing PSO. The ACO's main idea has been the ability of ant colonies to find the shortest path between the nest and the food source, taking advantage of its pheromone properties and accumulation. Utilizing the collective intelligence and smart behavior of the bee colony to search and find food has been the fundamental inspiration in ABC design. The light emitted by fireflies can be used for a variety of reasons, such as attracting prey and hunting, attracting other members of the group (attracting the opposite sex), and as a communication strategy. This fascinating light of fireflies has been a remarkable and interesting phenomenon, the inspiration of which has led to the development of the FA. Searching strategies and behaviors of animals, birds, and insects to find food sources or prey hunting have been the main ideas in the design of various techniques such as Grey Wolf Optimization (GWO) algorithm[15], Pelican Optimization Algorithm (POA)[16], Marine Predator Algorithm (MPA)[17], Orca Predation Algorithm (OPA)[18], Whale Optimization Algorithm (WOA)[19], Reptile Search Algorithm (RSA)[20], and Tunicate Search Algorithm (TSA)[21].

Physics-based algorithms have been developed on the base of using some physical processes and modeling of physical forces and laws. Simulated Annealing (SA) is the name of the most familiar physics-based algorithm based on simulation of the cooling of a molten metal in the refrigeration process[22]. The use of gravity force along with Newton's laws of motion have been the basic principles employed in Gravitational Search Algorithm (GSA) design[23]. Flow regimes and classical fluid mechanics have been a fundamental inspiration in developing Flow Regime Algorithm (FRA)[24]. Mathematical modeling of the nuclear reaction process in two stages of nuclear fusion and nuclear fission is employed in the design of Nuclear Reaction Optimization (NRO)[25]. The application of three concepts in cosmology, including wormholes, black holes, and white holes, has been the basis of the Multi-Verse Optimizer (MVO) design[26].

Game-based algorithms are inspired by player behaviors, rules governing individual and group games. The strategy used by different players to put the puzzle pieces together and solve it has been the idea of designing the Puzzle Optimization Algorithm (POA)[27]. Simulation of the coaching process, holding competitions, and teams interacting with each other during a competitive season of volleyball has led to the design of the Volleyball Premier League (VPL) optimization method[28]. Mathematizing the competition between teams and groups playing a tug-of-war game and trying to win has been the main idea in the development of Tug of War Optimization (TWO) approach[29].

Human-based algorithms are developed based on the simulation of human activities and behaviors in performing various tasks. Among the approaches of this group can be mentioned Teaching-Learning-Based Optimization (TLBO) based on modeling the interactions of a teacher and learners in the classroom[30], Poor and Rich Optimization (PRO) based on the modeling of the efforts of the rich and poor groups to improve their economic situation[31], and Human Behavior-Based Optimization (HBBO) based on the modeling of human thoughts and behaviors[32].

Hybrid Leader-Based Optimization

In this section, the concepts of the proposed Hybrid Leader-Based Optimization (HLBO) approach are stated and the HLBO mathematical formulation is presented.

Inspiration and main idea of HLBO

In population-based algorithms, each member of the population is a searcher in the problem-solving space and therefore a candidate solution. Based on the algorithm steps and information transfer, the population members are able to improve their position to provide better solutions. The dependence of the algorithm population update process on specific members (such as the best member of the population and the worst member of the population) may prevent the algorithm from searching globally in the problem-solving space. These conditions can lead to the rapid convergence of the algorithm towards the local optimal solution and as a result, the algorithm fails to identify the main optimal area in the search space. Therefore, overreliance on the process of updating the algorithm population to certain members reduces the exploration ability within the algorithm. In the proposed HLBO method, a unique hybrid leader is employed to update and guide each member of the algorithm population in the search space. This hybrid leader is generated based on three different members including the best member, one random member, and the corresponding member.

Mathematical Model of HLBO

The HLBO population is similar to other population-based algorithms that can be mathematically modeled using a matrix according to Equation (1).

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times m} = \begin{bmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Nj} & \cdots & x_{Nm} \end{bmatrix}_{N\times m}, \tag{1}$$

where X is the HLBO population, X_i is the ith candidate solution, $x_{i,j}$ is the value of jth variable determined by the ith candidate solution, N is the size of HLBO population, and m is the number of problem variables.

The position of each member of the population is initially initialized randomly by considering the constraints of the problem variables based on Equation (2).

$$x_{i,j} = lb_j + r \cdot (ub_j - lb_j)\ , \tag{2}$$

where r is a random real number from the interval $[0,1]$, lb_j and ub_j are the lower and upper bound of the jth problem variables respectively.

The objective function of the problem is evaluated based on each of the candidate solutions determined by the members of the population, which is specified in Equation (3) using a vector.

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix}_{N\times 1}, \tag{3}$$

where F represents the vector of the objective functions and F_i denotes the objective function value delivered from the ith candidate solution.

The values obtained for the objective function are a measure of the quality of the candidate solutions. The member that provides the best value for the objective function is known as the best member (X_{best}) and the member that provides the worst value for the objective function is known as the worst member (X_{worst}). These values are updated in each algorithm iteration. What distinguishes optimization algorithms from each other is the process used to update the algorithm population. Two important and influential indicators in the performance of optimization algorithms that should be considered in the process of updating and changing the position in the search space are exploration (global search) and exploitation (local search).

Phase 1: Exploration (global search)

Exploration is a feature that enables members of the algorithm population to accurately scan different areas of the search space to be able to find the original optimal area. Excessive reliance on specific members of the population (such as the best member) in the process of updating the algorithm population position prevents the global search of the algorithm in the search space and reduces the algorithm's ability to explore. This dependence in the update process can lead to early convergence of the algorithm to the local optimal and as a result the algorithm fails to identify the main optimal area in the search space. However, some population members, like the best member, have useful information that should not be overlooked. HLBO uses a hybrid leader to update members of the population. This hybrid leader is produced for each member of the population at each repetition. In constructing a random leader, three members of the population, including

(i) the corresponding member (the same member to be led by this hybrid leader).

(ii) the best member,

(iii) a random member of the population is influential.

The participation coefficient of each of these three members in the production of the hybrid leader is based on the quality of that member in providing a better value for the objective function. The quality of each member of the population in presenting the candidate solution is calculated using Equation (4).

$$q_i = \frac{F_i - F_{worst}}{\sum_{j=1}^{N}(F_j - F_{worst})}\ , \quad i \in \{1,2,\ldots N\}\ . \tag{4}$$

Then, using the results of Equation (4), the participation coefficients for each member are calculated using Equation (5).

$$PC_i = \frac{q_i}{q_i + q_{best} + q_k}\ , \qquad PC_{best} = \frac{q_{best}}{q_i + q_{best} + q_k}\ , \qquad PC_k = \frac{q_k}{q_i + q_{best} + q_k}\ , \tag{5}$$

where $i,k \in \{1,2,\ldots,N\}$, $k \neq i$, q_i is the quality of the ith candidate solution, F_{worst} is the value of the objective function of the worst candidate solution, PC_i, PC_{best}, PC_k are the participation coefficients of the ith member, the best member, and the kth member (k is an integer determined randomly from the set $\{1,2,\ldots,N\}$), respectively, in producing the hybrid leader.

After determining the participation coefficients, the hybrid leader is generated for each member of the population using Equation (6).

$$HL_i = PC_i \cdot X_i + PC_{best} \cdot X_{best} + PC_k \cdot X_k\ , \tag{6}$$

where HL_i is the hybrid leader for the ith member and X_k is a randomly selected population member which the index k is the row number of this member in the population matrix. The new position for each member of the population in the search space under the guidance of the hybrid leader is calculated using Equation (7). This new position is acceptable to the corresponding member if the value of the objective function is improved from the previous position, otherwise it remains in the previous position. These update conditions are modeled in Equation (8).

$$x_{i,j}^{new,P1} = \begin{cases} x_{i,j} + r \cdot (HL_{i,j} + I \cdot x_{i,j})\ , & F_{HL_i} < F_i; \\ x_{i,j} + r \cdot (xL_{i,j} - HL_{i,j})\ , & \text{else}\ , \end{cases} \tag{7}$$

$$X_i = \begin{cases} X_i^{new,P1}\ , & F_i^{new,P1} < F_i; \\ X_i\ , & \text{else}\ , \end{cases} \tag{8}$$

where $X_i^{new,P1}$ is the new position of the ith member, $x_{i,j}^{new,P1}$ is its jth dimension, $F_i^{new,P1}$ is its objective function value based on the first phase of HLBO, I is an integer which is selected randomly between 1 and 2, and F_{H_i} is the value of the objective function obtained from hybrid leader of the ith member.

Phase 2: Exploitation (local search)

Exploitation is an ability for members of the algorithm population that enables them to search locally for finding better solutions near the obtained solutions. Therefore, in HLBO a neighborhood around each member of the population is considered that allows that member to change position by searching locally in that area and finding a position with a better value for the objective function. This local search is modeled to improve and increase HLBO exploitation ability using Equation (9). In this phase, the newly calculated position is also acceptable if it improves the value of the objective function, which is simulated in Equation (10).

$$x_{i,j}^{new,P2} = x_{i,j} + (1-2r) \cdot R \cdot \left(1 - \frac{t}{T}\right) \cdot x_{i,j} \, , \tag{9}$$

$$X_i = \begin{cases} X_i^{new,P2} \, , & F_i^{new,P2} < F_i; \\ X_i \, , & \text{else} \, , \end{cases} \tag{10}$$

where $X_i^{new,P2}$ is the new position of the ith member, $x_{i,j}^{new,P2}$ is its jth dimension, $F_i^{new,P2}$ is its objective function value based on the second phase of HLBO, R is the constant equal to 0.2, t is the iteration counter, and T is the maximum number of iterations.

Repetition Process, Pseudo-Code, and Flowchart of HLBO

By implementing the first and second phases, all HLBO members are updated and an iteration of the algorithm is completed. The algorithm enters the next iteration and the HLBO population update process continues based on the exploration and exploitation phases according to Equations (4) to (10). This process continues until the end of the algorithm, and finally the best candidate solution experienced during the iterations is introduced as the solution to the problem. The HLBO pseudocode is presented in Algorithm 1.

Algorithm 1. Pseudo-code of HLBO.

Start HLBO.
Input the optimization problem information.
Adjust N and T.
Initialize the HLBO population position and evaluate the objective function.
For $i = 1$ to N
For $t = 1$ to T
Phase 1: Exploration phase
Calculate quality q_i of candidate solutions using Equation (4).
Calculate participation coefficients PC_i, PC_k, and PC_{best} using Equation (5).
Create hybrid leader HL_i using Equation (6).
Calculate new position of the ith member using Equation (7).
Update the ith member using Equation (8).
Phase 2: Exploitation phase
Calculate new position of the ith member using Equation (9).
Update the ith member using Equation (10).
End.
Update the best found candidate solution.
End.
Output: The best candidate solution obtained by HLBO.
End HLBO.

Computational Complexity of HLBO

The HLBO initialization and preparation process has a computational complexity equal to $O(Nm)$, where N refers to the number of population members and m is the number of variables in the problem. In each iteration, for each member, a hybrid leader must be generated, resulting in the computational complexity of generating the hybrid leaders equal to $O(NmT)$, where T is the maximum number of iterations of the algorithm. The HLBO update process has two phases of exploration and exploitation, which in both phases the objective function is evaluated. As a result, the computational complexity of HLBO update process equals $O(2NmT)$. Thus, the total computational complexity of HLBO is equal to $O(Nm(1+3T))$.

Simulation Studies and Results

This section is devoted to simulation studies and evaluation of the proposed HLBO performance in optimization. HLBO has been implemented to provide optimal solutions of twenty-three standard benchmark functions of three main types (complete definitions, domains and tables of suitable values of parameters of functions F_1 to F_{23} can be found in the paper[33]) unimodal function (functions F_1 to F_7), high-dimensional multimodal functions (functions F_8 to F_{13}), and fixed-dimensional multimodal functions (functions F_1 to F_7). The optimization results obtained from HLBO are compared with the performance of eight well-known algorithms including PSO, MPA, GA, WOA, TLBO, TSA, GSA, and GWO. The HLBO and the eight mentioned algorithms in twenty independent implementations are employed in optimizing the benchmark functions while each iteration contains 1000 iterations. The optimization results are reported using four statistical indicators: mean, best, standard deviation, and median. Moreover, the rank of each algorithm in providing a better solution for each benchmark function as well as for each group of objective functions is specified. Table 1 lists the adjusted values of the control parameters of the eight competitor algorithms.

Algorithm	Parameter	Value
MPA	Binary vector	$U=0$ or $U=1$.
	Random vector	R is a vector of uniform random numbers in $[0,1]$.
	Constant number	$P=0.5$.
	Fish Aggregating Devices	$FADs=2$.
TSA	c_1, c_2, c_3	Random numbers, which lie in the interval $[0,1]$.
	$Pmin$	1
	$Pmax$	4
WOA	l is a random number in $[-1,1]$.	
	r is a random vector in $[0,1]$.	
	Convergence parameter a	a: Linear reduction from 2 to 0.
GWO	Convergence parameter a	a: Linear reduction from 2 to 0.
TLBO	random number	$rand$ is a random number from the interval $[0,1]$.
	T_F: teaching factor	$T_F = \text{round}\,(1+rand)$.
GSA	$Alpha$	20
	G_0	100
	$Rnorm$	2
	$Rnorm$	1
PSO	Velocity limit	10% of dimension range.
	Topology	Fully connected.
	Inertia weight	Linear reduction from 0.9 to 0.1.
	Cognitive and social constant	$(C_1, C_2) = (2,2)$.
GA	Type	Real coded.
	Mutation	Gaussian ($Probability = 0.05$)
	Crossover	Whole arithmetic ($Probability = 0.8$,
	Selection	Roulette wheel (Proportionate)

Table 1. Adjusted values of the control parameters of eight competitor algorithms.

Evaluation of unimodal benchmark functions

The results of optimization of F_1 to F_7 benchmark functions using HLBO and competitor algorithms are released in Table 2. Experimental results show that HLBO provides the global optimal for F_1 and F_6. HLBO is the best optimizer against competitor algorithms in optimizing F_2, F_3, F_4, F_5, and F_7. What can be deduced from the analysis of the reported results is that HLBO is highly efficient in addressing unimodal optimization problems compared to eight competitor algorithms.

Evaluation of high-dimensional multimodal benchmark functions

The employment results of HLBO and eight competitor algorithms in optimizing F_8 to F_{13} benchmark high-dimensional multimodal functions are reported in Table 3. HLBO has managed to find the global optimum in optimizing the F_9 and F_{11}. HLBO is the first best optimizer for handling F_8 and F_{10}. In the case of F_{12} the WOA algorithm and in solving the F_{13} the GSA algorithm are the first best optimizers, while HLBO is the second best optimizer for these functions. Analysis of simulation results shows HLBO capability in solving high dimensional multimodal optimization problems.

Evaluation of fixed-dimensional multimodal benchmark functions

The results of implementing HLBO and competitor algorithms on benchmark F_{14} to F_{23} benchmark functions are presented in Table 4. What is evident from the simulation results is that HLBO is the first best optimizer in solving F_{14} to F_{23} benchmark functions compared to competitor algorithms. The presented experimental results show that HLBO has a superior performance over similar algorithms in dealing with multimodal optimization problems.

The convergence curves behavior of HLBO and competitor algorithms in achieving solutions for objective functions F_1 to F_{23} is presented in Figure 1.

Discussion

Optimization algorithms by utilizing exploration for global search and exploitation for local search, have the ability to handle optimization problems. To analyze the exploitation ability of HLBO in local search, the unimodal objective functions are favorable with only one main peak. In this type of optimization issues, the main challenge is the convergence towards the global optima. The optimization results of unimodal functions using HLBO indicate the exploitation ability of the proposed method in converging to the global optimal solution. In particular, HLBO has demonstrated its high local search ability by converging to the global optimal in handling the functions F_1 and F_6. High dimensional multimodal functions due to having multiple local optimal solutions are suitable options for measuring the exploration ability of optimization algorithms to global search and find the main optimal area. The main challenge in solving these problems is to accurately scan the search space and prevent the algorithm from getting stuck in the optimal local areas. The results of implementing HLBO on high-dimensional multimodal functions show that the proposed approach has an acceptable exploration ability in scanning the search space and finding the optimal area. The exploratory power of HLBO in identifying the optimal region, especially in the F_9 and F_{11} functions, is evident that it has been able to provide the global optimal. In addition to having the right quality of exploration and exploitation, having the right balance between these two indicators is the key to the success of optimization algorithms. Fixed dimensional multimodal functions have been selected to evaluate the ability of HLBO to strike a balance between exploration and exploitation. In this type of problem, it is important to simultaneously find the main optimal area based on global search and converge as much as possible to the global optimal based on local search. The optimization results of this type of function using the proposed approach show the high capability of HLBO in balancing exploration and exploitation to discover the optimal area and converge towards the global optimal.

Conclusion and future works

In this paper, a new optimization algorithm called Hybrid Leader Optimization (HLBO) is introduced. The use of a hybrid leader generated by three different members was HLBO's idea in updating the algorithm population in the search space. The HLBO implementation process was mathematically modeled in two phases of exploration and exploitation. Twenty-three objective functions were employed to evaluate the performance of HLBO in achieving optimal solutions for optimization problems. The results of the unimodal functions indicated the high exploitation ability of HLBO to search locally and converge towards global optima. The results of optimizing multimodal functions showed the high exploration ability of HLBO to search globally and discover the optimal area without getting caught up in local optimal. The results of HLBO compared to the performance of eight well-known algorithms showed that HLBO has a superior performance by providing appropriate solutions in most cases due to the appropriate balance between exploration and exploitation. The proposed HLBO opens up several research subjects for further work in the future. Specific research potentials are the development of binary and multimodal versions of HLBO. The HLBO employment on optimization topics in various sciences as well as real-world applications are other suggestions for future studies.

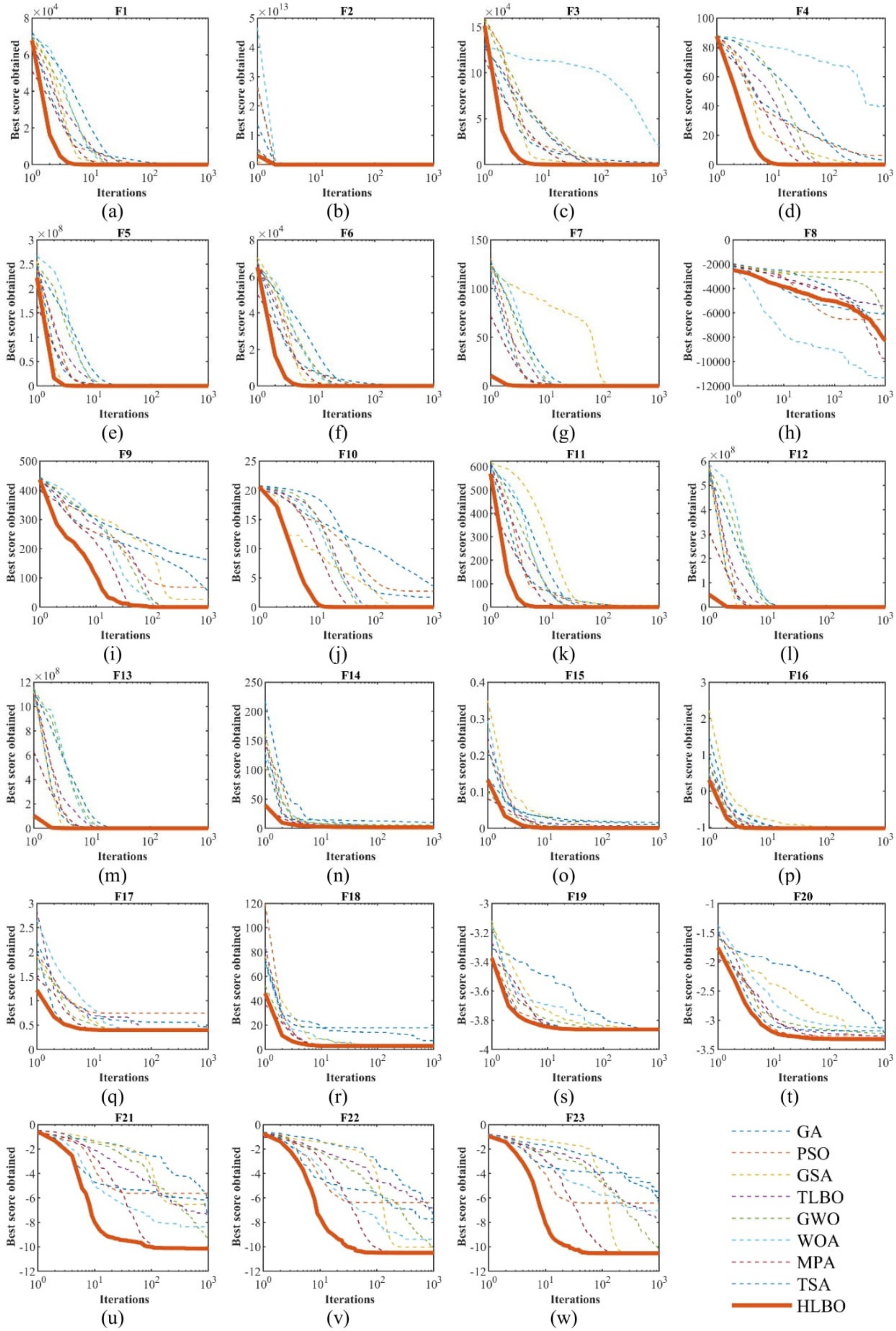

Figure 1. Convergence curves of the HLBO and competitor algorithms in optimizing objective functions F_1 to F_{23}.

		GA	PSO	GSA	TLBO	GWO	WOA	TSA	MPA	HLBO
F_1	mean	13.39109	1.77E-05	2.03E-17	1.34E-59	1.09E-58	1.79E-64	8.21E-33	1.7E-18	0
	best	6.905378	2E-10	8.2E-18	9.36E-61	7.73E-61	1.25E-65	1.14E-62	3.41E-28	0
	std	5.552616	5.86E-05	7.1E-18	2.05E-59	4.09E-58	2.75E-64	2.53E-32	6.76E-18	0
	med	11.04546	9.92E-07	1.78E-17	4.69E-60	1.08E-59	6.29E-65	3.89E-38	1.27E-19	0
	rank	9	8	7	3	4	2	5	6	1
F_2	mean	2.479574	0.341155	2.37E-08	5.55E-35	1.3E-34	1.57E-51	5.02E-39	2.78E-09	9.3E-222
	best	1.591137	0.001741	1.59E-08	1.32E-35	1.55E-35	1.14E-57	8.26E-43	4.25E-18	2.3E-223
	std	0.642826	0.669594	3.96E-09	4.71E-35	2.2E-34	5.95E-51	1.72E-38	1.08E-08	0
	med	2.463873	0.130114	2.33E-08	4.37E-35	6.38E-35	1.9E-54	8.26E-41	3.18E-11	2.1E-222
	rank	9	8	7	4	5	2	3	6	1
F_3	mean	1537.012	589.5083	279.3581	7.01E-15	7.41E-15	7.56E-09	3.2E-19	0.37704	3.1E-167
	best	1014.689	1.614937	81.91242	1.21E-16	4.75E-20	3.38E-09	7.29E-30	0.032038	6.4E-197
	std	367.2429	1524.005	112.2994	1.27E-14	1.9E-14	2.38E-09	9.9E-19	0.201772	0
	med	1510.715	54.15445	291.5324	1.86E-15	1.59E-16	7.2E-09	9.81E-21	0.378658	1.9E-181
	rank	9	8	7	3	4	5	2	6	1
F_4	mean	2.094404	3.9636	3.26E-09	1.58E-15	1.26E-14	0.001285	2.01E-22	3.66E-08	4.8E-206
	best	1.389849	1.605533	2.09E-09	6.42E-16	3.43E-16	5.88E-05	1.87E-52	3.42E-17	9.4E-208
	std	0.337071	2.203987	7.5E-10	7.14E-16	2.32E-14	0.00062	5.96E-22	6.45E-08	0
	med	2.09854	3.26186	3.34E-09	1.54E-15	7.3E-15	0.001417	3.13E-27	3.03E-08	8.7E-207
	rank	8	9	5	3	4	7	2	6	1
F_5	mean	310.4517	50.26629	36.10878	145.675	26.86252	27.17731	28.76917	42.50033	26.28159
	best	160.5013	3.647051	25.83811	120.7932	25.22966	26.45099	28.53831	41.58682	24.7708
	std	120.4671	36.52536	32.46201	19.73667	0.881999	0.626574	0.364803	0.616877	0.956216
	med	279.5174	28.70268	26.07475	142.9438	26.71803	26.93543	28.54912	42.49068	26.53765
	rank	9	7	5	8	2	3	4	6	1
F_6	mean	14.55074	20.25179	0	0.45	0.642403	0.071533	3.84E-20	0.390896	0
	best	6.0042	5	0	0	1.57E-05	0.014645	6.74E-26	0.274582	0
	std	5.834957	12.77601	0	0.510418	0.301212	0.078194	1.5E-19	0.080285	0
	med	13.5	19	0	0	0.621487	0.029317	6.74E-21	0.406648	0
	rank	7	8	1	5	6	3	2	4	1
F_7	mean	0.00568	0.113422	0.020694	0.00313	0.000819	0.00193	0.000277	0.002182	0.000126
	best	0.002111	0.029593	0.01006	0.001362	0.000248	4.24E-05	0.000104	0.001429	2.43E-05
	std	0.002433	0.045875	0.011363	0.001351	0.000503	0.003342	0.000123	0.000466	7.41E-05
	med	0.005365	0.107872	0.016995	0.002912	0.000629	0.00098	0.000367	0.002181	0.000113
	rank	7	9	8	6	3	4	2	5	1
Sum rank		58	57	40	32	28	26	20	39	7
Mean rank		8.285714	8.142857	5.714286	4.571429	4	3.714286	2.857143	5.571429	1
Total rank		9	8	7	5	4	3	2	6	1

Table 2. Evaluation results of unimodal functions.

		GA	PSO	GSA	TLBO	GWO	WOA	TSA	MPA	HLBO
F_8	mean	-8184.33	-6908.59	-2849.04	-7803.51	-5885.06	-7687.48	-5669.59	-3652.11	-8246.45
	best	-9717.68	-8501.44	-3969.23	-9103.77	-7227.05	-8597.11	-5706.3	-4419.9	-8763.3
	std	795.1489	836.7303	540.3636	986.6122	984.5011	1105.161	21.8595	474.5854	300.4366
	med	-8117.25	-7098.95	-2671.33	-7735.22	-5774.63	-8290.68	-5669.63	-3632.65	-8306.6
	rank	2	5	9	3	6	4	7	8	1
F_9	mean	62.41581	57.06498	16.26904	10.67825	8.53E-15	0	0.005888	152.7027	0
	best	36.86623	27.85883	4.974795	9.873963	0	0	0.004776	128.2306	0
	std	15.21593	16.51676	4.660198	0.396658	2.08E-14	0	0.000696	15.18568	0
	med	61.67858	55.22468	15.42187	10.88785	0	0	0.005871	154.6214	0
	rank	7	6	5	4	2	1	3	8	1
F_{10}	mean	3.222042	2.154811	3.57E-09	0.263217	1.71E-14	3.91E-15	6.38E-11	8.31E-10	1.95E-15
	best	2.757203	1.155151	2.64E-09	0.156415	1.51E-14	8.88E-16	8.14E-15	1.68E-18	8.88E-16
	std	0.361677	0.549389	5.27E-10	0.07285	3.15E-15	2.65E-15	2.6E-10	2.8E-09	1.67E-15
	med	3.120322	2.170083	3.64E-09	0.261541	1.51E-14	4.44E-15	1.1E-13	1.05E-11	8.88E-16
	rank	9	8	6	7	3	2	4	5	1
F_{11}	mean	1.230294	0.046297	3.737798	0.587714	0.003753	0.00302	1.55E-06	0	0
	best	1.14127	7.29E-09	1.519288	0.310117	0	0	4.23E-15	0	0
	std	0.062769	0.051838	1.670263	0.16909	0.007344	0.013507	3.38E-06	0	0
	med	1.227231	0.029473	3.424268	0.582026	0	0	8.77E-07	0	0
	rank	7	5	8	6	4	3	2	1	1
F_{12}	mean	0.047029	0.480718	0.036287	0.020551	0.037212	0.007729	0.050167	0.082564	0.011386
	best	0.018364	0.000145	5.57E-20	0.002032	0.019307	0.001142	0.035428	0.077912	0.0036
	std	0.028483	0.602662	0.06087	0.028645	0.013874	0.008983	0.009857	0.002388	0.004855
	med	0.04179	0.1556	1.48E-19	0.015181	0.032991	0.003919	0.050935	0.082136	0.011033
	rank	6	9	4	3	5	1	7	8	2
F_{13}	mean	1.20862	0.508413	0.002085	0.329141	0.576374	0.193317	2.658936	0.565279	0.184032
	best	0.49809	9.99E-07	1.18E-18	0.038266	0.297822	0.029662	2.63175	0.280295	0.136248
	std	0.333737	1.251681	0.005476	0.198935	0.170438	0.150912	0.009864	0.187798	0.025522
	med	1.218053	0.043997	2.14E-18	0.282962	0.578323	0.152006	2.66175	0.579854	0.179199
	rank	8	5	1	4	7	3	9	6	2
Sum rank		39	38	33	27	27	14	32	36	8
Mean rank		6.5	6.333333	5.5	4.5	4.5	2.333333	5.333333	6	1.333333
Total rank		8	7	5	3	3	2	4	6	1

Table 3. Evaluation results of high-dimensional multimodal functions.

References

1. Ray, T. & Liew, K. M. Society and civilization: an optimization algorithm based on the simulation of social behavior. *IEEE Transactions on Evol. Comput.* **7**, 386–396 (2003).
2. Kaidi, W., Khishe, M. & Mohammadi, M. Dynamic levy flight chimp optimization. *Knowledge-Based Syst.* **235**, 107625 (2022).
3. Sergeyev, Y. D., Kvasov, D. & Mukhametzhanov, M. On the efficiency of nature-inspired metaheuristics in expensive global optimization with limited budget. *Sci. reports* **8**, 1–9 (2018).
4. Iba, K. Reactive power optimization by genetic algorithm. *IEEE Transactions on power systems* **9**, 685–692 (1994).
5. Wang, J.-S. & Li, S.-X. An improved grey wolf optimizer based on differential evolution and elimination mechanism. *Sci. reports* **9**, 1–21 (2019).
6. Wolpert, D. H. & Macready, W. G. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation* **1**, 67–82 (1997).
7. Li, C. *et al.* Integrated optimization algorithm: A metaheuristic approach for complicated optimization. *Inf. Sci.* **586**, 424–449 (2022).
8. Goldberg, D. E. & Holland, J. H. Genetic algorithms and machine learning. *Mach. Learn.* **3**, 95–99 (1988).

		GA	PSO	GSA	TLBO	GWO	WOA	TSA	MPA	HLBO
F_{14}	mean	0.998729	2.173656	3.591704	2.264374	3.74098	3.106213	1.798816	0.998803	0.998
	best	0.998004	0.998004	0.999508	0.998391	0.998004	0.998004	0.9979	0.998098	0.998
	std	0.002476	2.93651	2.779161	1.149558	3.969687	3.533635	0.527526	0.00032	0
	med	0.998022	0.998004	2.986658	2.275231	2.982105	0.998353	1.912608	0.998898	0.998
	rank	2	5	8	6	9	7	4	3	1
F_{15}	mean	0.005396	0.001684	0.002403	0.00317	0.00637	0.000664	0.000408	0.003936	0.000307
	best	0.000776	0.000307	0.000805	0.002206	0.000307	0.000313	0.000264	0.00027	0.000307
	std	0.008102	0.004932	0.001195	0.000394	0.009401	0.000349	7.59E-05	0.005051	4.28E-13
	med	0.002074	0.000307	0.002312	0.003185	0.000308	0.000521	0.00039	0.0027	0.000307
	rank	8	4	5	6	9	3	2	7	1
F_{16}	mean	-1.0316	-1.03162	-1.03162	-1.03162	-1.03162	-1.03162	-1.03158	-1.03158	-1.03163
	best	-1.03163	-1.03163	-1.03163	-1.03163	-1.03163	-1.03163	-1.03161	-1.0316	-1.03163
	std	4.43E-05	3.18E-05	3.18E-05	3.18E-05	3.18E-05	3.18E-05	3.46E-05	4.09E-05	2.16E-16
	med	-1.03162	-1.03163	-1.03163	-1.03163	-1.03163	-1.03163	-1.0316	-1.0316	-1.03163
	rank	5	2	2	2	4	3	6	7	1
F_{17}	mean	0.436996	0.785471	0.397915	0.397915	0.397916	0.397916	0.400117	0.401406	0.397887
	best	0.397888	0.397887	0.397887	0.397887	0.397887	0.397887	0.398052	0.398869	0.397887
	std	0.140738	0.721739	8.57E-05	8.57E-05	8.58E-05	8.55E-05	0.004474	0.004437	0
	med	0.397939	0.398027	0.397887	0.397887	0.397888	0.397888	0.399052	0.398869	0.397887
	rank	7	8	2	2	3	4	5	6	1
F_{18}	mean	4.360454	3.00021	3.00021	3.00021	3.000221	3.000219	3.001828	3.00021	3
	best	3.000001	3	3	3	3	3	3	3	3
	std	6.039901	0.000646	0.000646	0.000646	0.000645	0.000644	0.000896	0.000646	1.13E-15
	med	3.001083	3	3	3	3.000007	3.000003	3.001797	3	3
	rank	9	3	4	2	7	6	8	5	1
F_{19}	mean	-3.8543	-3.86274	-3.86274	-3.86134	-3.86213	-3.86064	-3.80656	-3.86266	-3.86278
	best	-3.86278	-3.86278	-3.86278	-3.8625	-3.86278	-3.86278	-3.8366	-3.8627	-3.86278
	std	0.014863	0.000119	0.000119	0.001383	0.001709	0.00259	0.015283	0.000119	1.43E-07
	med	-3.86236	-3.86278	-3.86278	-3.862	-3.86276	-3.86167	-3.8066	-3.8627	-3.86278
	rank	8	2	3	6	5	7	9	4	1
F_{20}	mean	-2.82387	-3.26191	-3.32196	-3.20114	-3.25235	-3.22295	-3.31948	-3.32106	-3.32199
	best	-3.31342	-3.322	-3.322	-3.26174	-3.32199	-3.32198	-3.3212	-3.3213	-3.322
	std	0.385953	0.070609	0.000102	0.031828	0.07654	0.090399	0.003075	0.000132	2.81E-05
	med	-2.96828	-3.32166	-3.322	-3.2076	-3.26231	-3.19517	-3.3206	-3.3211	-3.32199
	rank	9	5	2	8	6	7	4	3	1
F_{21}	mean	-4.60396	-5.53917	-5.4486	-9.19008	-9.44514	-8.87627	-5.50204	-9.90434	-10.1532
	best	-8.52131	-10.1532	-10.1532	-9.66387	-10.1532	-10.1531	-9.50209	-10.1532	-10.1532
	std	1.924706	3.076268	3.094017	0.120742	1.739512	2.263499	1.256581	0.559202	4.25E-10
	med	-4.3747	-5.10077	-3.76931	-9.1532	-10.1525	-10.1512	-5.50209	-10.1532	-10.1532
	rank	9	6	8	4	3	5	7	2	1
F_{22}	mean	-5.11736	-7.63223	-9.76638	-10.0486	-10.4024	-9.33722	-5.91343	-10.2858	-10.4029
	best	-9.11064	-10.4029	-10.4029	-10.4029	-10.4028	-10.4028	-9.06249	-10.4029	-10.4029
	std	1.969583	3.541651	1.708382	0.39829	0.000443	2.179987	1.7549	0.24536	1.92E-05
	med	-5.0294	-10.4024	-10.4029	-10.1836	-10.4025	-10.4012	-5.06249	-10.4028	-10.4029
	rank	9	7	5	4	2	6	8	3	1
F_{23}	mean	-6.56207	-6.16474	-10.0188	-9.26419	-10.1302	-9.4522	-9.80976	-10.1408	-10.5364
	best	-10.2216	-10.5364	-10.5364	-10.534	-10.5363	-10.5363	-10.3683	-10.5364	-10.5364
	std	2.617229	3.734885	1.593807	1.676546	1.814378	2.221872	1.606421	1.14013	5.39E-06
	med	-6.5629	-4.50554	-10.5364	-9.67172	-10.536	-10.535	-10.3613	-10.5364	-10.5364
	rank	8	9	4	7	3	6	5	2	1
Sum rank		74	51	43	47	51	54	58	42	10
Mean rank		7.4	5.1	4.3	4.7	5.1	5.4	5.8	4.2	1
Total rank		8	5	3	4	5	6	7	2	1

Table 4. Evaluation results of fixed-dimensional multimodal functions.

9. Storn, R. & Price, K. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *J. global optimization* **11**, 341–359 (1997).

10. Castro, L. N. D. & Timmis, J. I. Artificial immune systems as a novel soft computing paradigm. *Soft computing* **7**, 526–544 (2003).

11. Kennedy, J. & Eberhart, R. Particle swarm optimization. In *Proceedings of ICNN'95 - International conference on neural networks*, 1942–1948 (IEEE, 1995).

12. Dorigo, M. & Stützle, T. *Handbook of Metaheuristics*, chap. Ant Colony Optimization: Overview and Recent Advances, 311–351 (Cham: Springer International Publishing, 2019).

13. Karaboga, D. & Basturk, B. Artificial bee colony (abc) optimization algorithm for solving constrained optimization problems. In *Foundations of Fuzzy Logic and Soft Computing. IFSA 2007. Lecture Notes in Computer Science*, 789–798 (Springer, Berlin, Heidelberg, 2007).

14. Yang, X.-S. Firefly algorithms for multimodal optimization. In *Stochastic Algorithms: Foundations and Applications. SAGA 2009. Lecture Notes in Computer Science*, 169–178 (Springer, Berlin, Heidelberg, 2009).

15. Mirjalili, S., Mirjalili, S. M. & Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **69**, 46–61 (2016).

16. Trojovský, P. & Dehghani, M. Pelican optimization algorithm: A novel nature-inspired algorithm for engineering applications. *Sensors* **22**, 855 (2022).

17. Faramarzi, A., Heidarinejad, M., Mirjalili, S. & Gandomi, A. H. Marine predators algorithm: A nature-inspired metaheuristic. *Expert. Syst. with Appl.* **152**, 113377 (2020).

18. Jiang, Y., Wu, Q., Zhu, S. & Zhang, L. Orca predation algorithm: A novel bio-inspired algorithm for global optimization problems. *Expert. Syst. with Appl.* **188**, 116026 (2022).

19. Mirjalili, S. & Lewis, A. The whale optimization algorithm. *Adv. engineering software* **95**, 51–67 (2016).

20. Abualigah, L., Elaziz, M. A., Sumari, P., Geem, Z. W. & Gandomi, A. H. Reptile search algorithm (rsa): A nature-inspired meta-heuristic optimizer. *Expert. Syst. with Appl.* **191**, 116158 (2022).

21. Kaur, S., Awasthi, L. K., Sangal, A. L. & Dhiman, G. Tunicate swarm algorithm: A new bio-inspired based metaheuristic paradigm for global optimization. *Eng. Appl. Artif. Intell.* **90**, 103541 (2020).

22. Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. Optimization by simulated annealing. *Science* **220**, 671–680 (1983).

23. Rashedi, E., Nezamabadi-Pour, H. & Saryazdi, S. Gsa: a gravitational search algorithm. *Inf. sciences* **179**, 2232–2248 (2009).

24. Tahani, M. & Babayan, N. Flow regime algorithm (fra): a physics-based meta-heuristics algorithm. *Knowl. Inf. Syst.* **60**, 1001–1038 (2019).

25. Wei, Z., Huang, C., Wang, X., Han, T. & Li, Y. Nuclear reaction optimization: A novel and powerful physics-based algorithm for global optimization. *IEEE Access* **7**, 66084–66109 (2019).

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

27. Zeidabadi, F. A. & Dehghani, M. Poa: Puzzle optimization algorithm. *Int. J. Intell. Eng. Syst.* **15**, 273–281 (2022).

28. Moghdani, R. & Salimifard, K. Volleyball premier league algorithm. *Appl. Soft Comput.* **64**, 161–185 (2018).

29. Kaveh, A. & Zolghadr, A. A novel meta-heuristic algorithm: tug of war optimization. *Iran Univ. Sci. & Technol.* **6**, 469–492 (2016).

30. Rao, R. V., Savsani, V. J. & Vakharia, D. Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. *Comput. Des.* **43**, 469–492 (2011).

31. Moosavi, S. H. S. & Bardsiri, V. K. Poor and rich optimization algorithm: A new human-based and multi populations algorithm. *Eng. Appl. Artif. Intell.* **86**, 165–181 (2019).

32. Ahmadi, S.-A. Human behavior-based optimization: a novel metaheuristic approach to solve complex optimization problems. *Neural Comput. Appl.* **28**, 233–244 (2017).

33. Yao, X., Liu, Y. & Lin, G. Evolutionary programming made faster. *IEEE Transactions on Evol. computation* **3**, 82–102 (1999).

Funding

This research was funded by the Project of Excellence of Faculty of Science, University of Hradec Králové, grant number 2210/2022-2023.

Author contributions

Conceptualization, M.D. and P.T.; methodology, P.T.; software, M.D.; validation, P.T. and M.D.; formal analysis, M.D.; investigation, P.T.; resources, P.T.; data curation, M.D.; visualization, P.T.; funding acquisition, P.T. All authors have read and agreed to the published version of the manuscript.

Competing interests

Te authors declare no competing interests.

Additional information

All data generated or analysed during this study are included in this published article [and its supplementary information files].

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- DATAHLBO.rar