Current Journal of Applied Science and Technology

Volume 42, Issue 17, Page 9-22, 2023; Article no.CJAST.102486
ISSN: 2457-1024
(Past name: British Journal of Applied Science & Technology, Past ISSN: 2231-0843, NLM ID: 101664541)

An Enhanced Dung Beetle Optimization Algorithm for Global Optimization

Zhenhui Lu [a*] and Jian Zhang [a]

[a] *Faculty of Civil Engineering and Mechanics, Jiangsu University, Zhenjiang-212013, P. R., China.*

Authors' contributions

This work was carried out in collaboration between both authors. Both authors read and approved the final manuscript.

Article Information

DOI: 10.9734/CJAST/2023/v42i174133

Open Peer Review History:
This journal follows the Advanced Open Peer Review policy. Identity of the Reviewers, Editor(s) and additional Reviewers, peer review comments, different versions of the manuscript, comments of the editors, etc are available here: https://www.sdiarticle5.com/review-history/102486

Original Research Article

Received: 27/04/2023
Accepted: 30/06/2023
Published: 01/07/2023

ABSTRACT

This paper proposes an enhanced dung beetle optimization (EDBO) algorithm in order to address the issues of the dung beetle optimization (DBO) algorithm which include easy convergence to the local optimal, slow convergence speed, and poor global search capability. The improvements in the EDBO are implemented via the following four aspects. Firstly, the SPM chaotic mapping designed through combing Sine mapping and Piece-Wise Linear Chaotic Mapping is introduced to initialize the population for increasing diversity of population. Secondly, the position update formula in the Golden Sine Algorithm (Golden-SA) is used to replace the formula for the mathematical model of dung beetle ball-rolling behavior without obstacle with the purpose of improving the convergence accuracy and accelerating the convergence speed. Thirdly, the spiral foraging strategy in the tuna swam optimization (TSO) is hybridized with the mathematical model of dung beetle breeding and foraging behavior. The hybridization not only balances the global exploration and local exploitation but also keeps the diversity of the population. Finally, the EDBO can enhance the capability of escaping the local optima and extending the search space by means of bringing in the two different sets of adaptive weight coefficients. The performance of the EDBO is evaluated and compared with

---

**Corresponding author: E-mail: lzh_jsdx_9207@163.com;*

other swarm intelligence optimization algorithms via the benchmark functions of different characteristics. The results demonstrate that the EDBO outperforms the classical DBO and other compared algorithms in terms of convergence speed and accuracy.

Keywords: Enhanced dung beetle optimization (EDBO); SPM chaotic mapping; golden sine algorithm; Spiral foraging strategy; adaptive weight coefficients.

1. INTRODUCTION

"The swarm intelligence (SI) optimization algorithm is a heuristic algorithm inspired by the biological behaviors which is used to find out the global optimal solution in the design space for optimization problems" [1,2]. The characteristics of the SI algorithms contain simple theoretical framework, excellent convergence accuracy and speed, and powerful global search ability. The common SI algorithms include Particle Swarm Optimization (PSO) [3], Grey Wolf Optimizer (GWO) [4], Cuckoo search (CS) [5], Wild Horse Optimizer (WHO) [6], Harris Hawks Optimization (HHO) [7], Northern Goshawk Optimization (NGO) [8], Coot Optimization Algorithm (COOT) [9], and so on. The dung beetle optimization (DBO) algorithm was first put forth by Jiankai Xue and Bo Shen in 2022 as a novel approach for handling the optimization issues [10]. The optimization performance of the DBO is superior to most of traditional optimization algorithms and has been used for handling the engineering optimization. Nevertheless, there are some drawbacks in the DBO including poor global search capability, premature convergence to the local optima and slow convergence speed.

In recent years, scholars have proposed improved metaheuristic algorithms by introducing several modification tactics to enhance the optimization performance due to the existing drawbacks of the metaheuristic algorithms. Zeng et al. [11] proposed "the improved Wild Horse Optimizer (IWHO)in which the diversity of the population was improved by using the SPM chaotic mapping for population initialization, the Golden Sine Algorithm was introduced to improve the convergence accuracy and speed, and the opposition-based learning and the Cauchy variation strategy were utilized to extend the search space and enhance the ability of avoiding getting trapped into the local optima". Han et al. [12] fused "the weight coefficients, optimal bootstrap position, and spiral search strategy in the Crow Search Algorithm to keep a balance between global exploration and local exploitation and improve the convergence rate". Liu et al. [13] combined "the Arithmetic Optimization Algorithm (AOA) with Gold-SA and additionally the Levy flight and Brownian mutation strategy proposed in this paper were introduced to improve the capability of the hybrid algorithm".

Thus, it is necessary to introduce several modification tactics to address the drawbacks of the DBO. This paper proposes an enhanced dung beetle optimization (EDBO) algorithm, which is reflected in four aspects:

(1) EDBO utilizes the SPM chaotic mapping to initialize the population aiming at increasing the diversity of population
(2) The individual update strategy in Golden Sine Algorithm is introduced in the EDBO to replace the formula for the mathematical model of dung beetle ball-rolling behavior without obstacle for the purpose of extending the search range and accelerating the convergence speed.
(3) The control coefficients of the spiral foraging strategy in the TSO are used to renew the formula for the mathematical model of dung beetle breeding and foraging behavior in order to reduce the probability of falling into the local optima and balance the global exploration and local exploitation.
(4) The two different sets of adaptive weight coefficients are introduced in the formula for the mathematical model of dung beetle stealing behavior to enhance the global searchability and avoid getting trapped in the local optima. The first one comes from the spiral foraging strategy of the TSO and the other one is newly proposed in the paper.

The rest of the paper is arranged as follows: The basic theory of the DBO is provided in Section 2; Section 3 introduces the proposed EDBO algorithm model at length; The efficacy of the proposed EDBO algorithm is evaluated in Section 4 by comparing with other optimization algorithms, including the classical DBO, on different unimodal benchmark functions and multimodal benchmark functions. Section 5 gives the conclusion of the paper.

2. THE CLASSICAL DBO ALGORITHM

"The Dung Beetle Optimization algorithm is a novel swarm intelligence optimization algorithm designed for handling both unconstrained and constrained optimization problems. The inspiration of the DBO comes from some dung beetle habits including ball-rolling, dancing, foraging, breeding, and stealing. The basic principle of this algorithm is that the DBO divides the dung beetle population into four subpopulations and then conducts the following four optimization processes which consist of rolling balls, foraging, breeding, and stealing" [10].

2.1 The Mathematical Model of Dung Beetle Ball-rolling Behavior

The dung beetle ball-rolling behavior is made up of the following two different situations. The first one is obstacle-free mode which means that the dung beetle will move forwards for search based on the navigation of sun without obstacle during the ball-rolling process. In this model, the position of dung beetles is updated by the following formula:

$$x_i^{g+1} = x_i^g + a \times k \times x_i^{g-1} + b \times \left|x_i^g - x_{worst}^g\right| \# \quad (1)$$

where g represents the number of the current iterations, x_i^g represents the position information of i^{th} dung beetle in the population at the g^{th} iteration, a represents a natural coefficient taking value of 1 or -1 in different situations where the value of 1 means no deviation and the value of -1 means deviation from the original direction, k represents an invariant quantity indicating the flexure coefficient in the range of $(0,0.2]$, b represents a fixed parameter within the range $(0,1)$, x_{worst}^g represents the global worst position in the current iteration, $\left|x_i^g - x_{worst}^g\right|$ represents the difference between i^{th} dung beetle and the global worst dung beetle which is used to simulate the changes in light intensity.

The second situation is obstructed mode which means that the dung beetles need to seek a new direction to move forwards by dancing when it encounters an obstacle and has difficulty in continuing to conduct the behavior of ball-rolling. A tangent function which only considers the values in the range of $[0,1]$ is used to mimic the behavior of dance as a method to determine a new ball-rolling direction. Thus, the position of dung beetle can be updated by the formula which is defined as follows:

$$x_i^{g+1} = x_i^g + \tan(\theta)\left|x_i^g - x_i^{g-1}\right| \# \quad (2)$$

where $\left|x_i^g - x_i^{g-1}\right|$ represents the difference between i^{th} dung beetle and $(i-1)^{th}$ dung beetle. It should be noted that the position of dung beetle is not be updated when $\theta = 0, \frac{\pi}{2}, or\ \pi$.

2.2 The mathematical Model of Dung Beetle Breeding Behavior

In nature, female dung beetles roll their dung balls to a safe place and then hide them. It is important for female dung beetles to choose a suitable place for laying their eggs so as to provide a safe habitat for their offspring. Inspired by this behavior, a frontier option strategy is proposed to determine the area where the eggs are produced. The strategy can be written as follows:

$$\begin{cases} X^{Lb^*} = max\{X^* \times (1-R), X^{Lb}\} \\ X^{Ub^*} = min\{X^* \times (1+R), X^{Ub}\} \end{cases} \# \quad (3)$$

where X^* represents the current local optimal position, X^{Lb^*} and X^{Ub^*} represent the bottom and top boundaries of the area where female dung beetles lay their eggs, X^{Lb} and X^{Ub} represent the lower and upper bounds of optimization problem, $R = 1 - g/G$, and G represents the upper limits of iterations.

Once the female dung beetles determine the spawning area, they produce their eggs in the area. It should be noted that each female dung beetle generates only one egg per iteration in the DBO. The boundary range is dynamically adjusted with the number of iterations increasing, thus the position where female dung beetles produce their eggs is also dynamic. The position update equation for the dung beetle breeding behavior can be defined as follows:

$$x_i^{g+1} = X^* + b_1 \times \left(x_i^g - X^{Lb^*}\right) + b_2 \times \left(x_i^g - X^{Ub^*}\right) \# \quad (4)$$

where x_i^g represents the location information of the i^{th} brood ball in the population at the g^{th} iteration, b_1 and b_2 represent two random and independent vectors which have the size of $1 \times D$ each, D represents the number of variables in the optimization problem. It is essential to ensure the position of the brood ball to be strictly restricted to a defined range.

2.3 The Mathematical Model of Dung Beetle Foraging Behavior

The foraging behavior is mainly designed for small dung beetles which emerge from the ground for searching food. Thus, it is necessary to determine the optimal foraging area for guiding the foraging dung beetles. The optimal foraging area is dynamically adjusted with the number of iterations increasing which can be expressed as follows:

$$\begin{cases} X^{Lb^b} = max\{X^b \times (1-R), X^{Lb}\} \\ X^{Ub^b} = min\{X^b \times (1+R), X^{Ub}\} \end{cases} \# \quad (5)$$

where X^b represents the global best position, X^{Lb^b} and X^{Ub^b} represent the lower and upper bounds of the optimal foraging region, other parameters have the same definition as in section 2.2. So, the location of small dung beetles can be updated by using the equation which is defined as follows:

$$x_i^{g+1} = x_i^g + C_1 \times \left(x_i^g - X^{Lb^b}\right) + C_2 \times \left(x_i^g - X^{Ub^b}\right) \# \quad (6)$$

where x_i^g represents the location information of the i^{th} small dung beetles in the population at the g^{th} iteration, C_1 is a random number with standard normal distribution, and C_2 represents a random vector within the range $(0,1)$ which has the size of $1 \times D$.

2.4 The Mathematical Model of Dung Beetle Stealing Behavior

There are some dung beetles called thieves which steal dung balls from other dung beetles in the population. Thus, the position update equation for the thieves can be defined as follows:

$$x_i^{g+1} = X^b + S \times t \times \left(\left|x_i^g - X^*\right| + \left|x_i^g - X^b\right|\right) \# \quad (7)$$

where x_i^g represents the location information of the i^{th} thief in the population at the g^{th} iteration, t represents a random vector obeying normal distribution with the size of $1 \times D$, and S represent a fixed parameter.

The flowchart of the DBO algorithm is given in Fig. 1.

3. ENHANCED DUNG BEETLE OPTIMIZATION ALGORITHM

Although the DBO algorithm is simple and has been applied to handle some unconstrained and constrained optimization problems, there are some shortcomings in it which include weak ability of global search and premature convergence to the local optimal. An enhanced dung beetle optimization algorithm has been proposed to overcome the existing deficiency.

3.1 SPM Chaotic Mapping

The DBO generates initial population randomly in the design space which can cause the loss of the population diversity and overconvergence in the subsequent iteration process. The chaos models have been confirmed to be effective for increasing diversity of population in swarm intelligence optimization algorithm. Common chaos models include Tent [14], Logistic [15], Henon [16], and Kent [17] chaos mapping whose basic method is to involves mapping chaotic sequences into individual search spaces. Two significant factors including simplicity and ergodicity should be considered when it comes to selecting a suitable chaotic mapping to generate a random sequence. Thus, the paper chooses an efficient chaotic mapping called SPM designed through combing Sine mapping and Piece-Wise Linear Chaotic Mapping which has superior chaotic and ergodic properties [11]. The formula is defined as follows:

$$x_{i+1} = \begin{cases} mod\left(\frac{x_i}{\eta} + \mu \sin(\pi x_i) + r, 1\right), 0 \le x_i < \eta \\ mod\left(\frac{x_i/\eta}{0.5-\eta} + \mu \sin(\pi x_i) + r, 1\right), \eta \le x_i < 0.5 \\ mod\left(\frac{1-x_i/\eta}{0.5-\eta} + \mu \sin\left(\pi(1-x_i)\right) + r, 1\right), 0.5 \le x_i < 1-\eta \\ mod\left(\frac{1-x_i}{\eta} + \mu \sin\left(\pi(1-x_i)\right) + r, 1\right), 1-\eta \le x_i < 1 \end{cases} \# \quad (8)$$

With the control parameter $\eta \in (0,1)$ and $\mu \in (0,1)$, the system is in a chaotic state. SPM Chaotic Mapping the value of control parameter and the initial value of x_0 at first. Then a random sequence within the range of $(0,1)$ are generated after a certain number of cyclic iterations. The initial population dung beetles based on the random sequence obtained by SPM Chaotic Mapping improve diversity of population. This paper chooses the Logistic mapping and Cubic mapping for comparison under the condition where it set the same iteration number 5000 and the same initial value x_0. The frequency distribution histograms of the three different chaotic mappings are presented in Fig. 2 It can be seen from Fig. 2 that SMP chaotic mapping displays better chaotic performance and ergodicity. Fig. 3 presents the population distribution scatter map of the three different chaotic mappings. The SPM mapping distributes more uniformly while the individuals of the Logistic and Cubic mappings distributed more around boundaries which causes the loss of the population diversity.

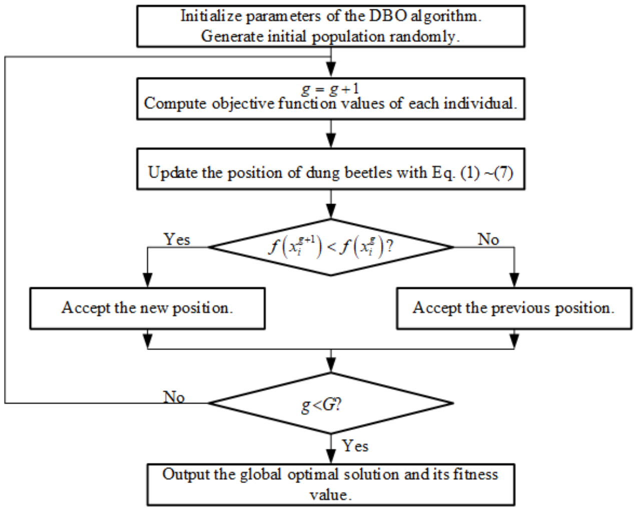


Fig. 1. The flowchart of the DBO algorithm

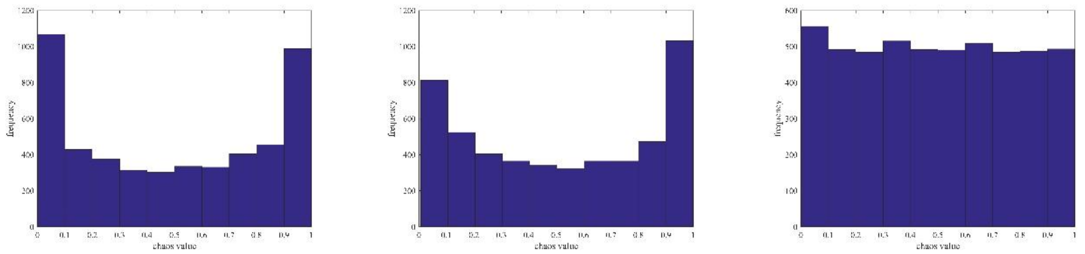


Fig. 2. Chaotic Mapping Histogram. (a) Logistic mapping; (b) Cubic mapping; (c) SPM mapping

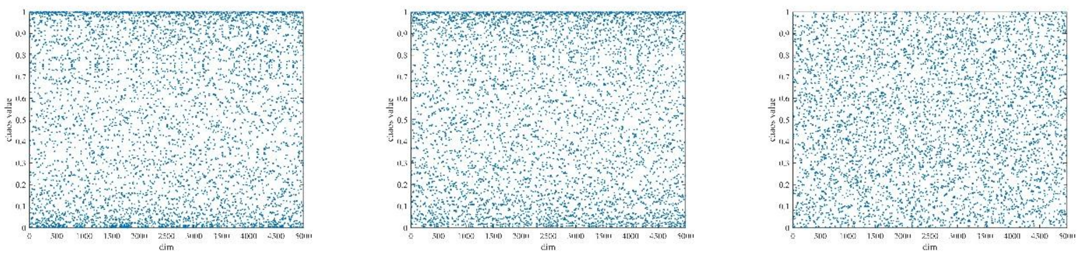


Fig. 3. Chaotic Mapping Scatter map: (a) Logistic mapping; (b) Cubic mapping; (c) SPM mapping

3.2 Golden Sine Algorithm

Golden sine algorithm has strong ability of global search and meanwhile the golden partition coefficient is introduced to enhance the capability of local search [18]. Thus, Golden Sine Strategy can keep a good balance between global exploration and local exploitation. The formula of golden sine algorithm can be expressed as follows:

$$x_i^{g+1} = x_i^g \times |\sin(R_1)| - R_2 \times \sin(R_1) \times \left|m_1 \times P_i^g - m_2 \times x_i^g\right| \# \quad (9)$$

where R_1 represents a random number in the interval $[0,2\pi]$ which determines the movement distance of i^{th} individual in the next iteration, R_2 represents a random number in the interval $[0,\pi]$ which determines the movement direction of i^{th} individual in the next iteration, m_1 and m_2 represent golden partition coefficients which is used to reduce the search space and guide the current individual to the global optimal. The coefficients m_1 and m_2 can be calculated by the equations as follows:

$$\begin{aligned} m_1 &= h_1 \times \tau + h_2 \times (1-\tau) \\ m_2 &= h_1 \times (1-\tau) + h_2 \times \tau \# \\ \tau &= \left(\sqrt{5}-1\right)/2 \end{aligned} \quad (10)$$

where a and b represent initial golden values, and τ represents the golden ratio.

In the DBO algorithm, the position update approach in the condition where dung beetles conduct ball-rolling behavior without obstacle is poor in the capability of local search. Therefore, this paper replaces the equation (1) with equation (9) to improve the DBO performance.

3.3 Tuna Swarm Optimization

The tuna swarm optimization (TSO) is a novel swarm-based metaheuristic algorithm inspired by the two different foraging behaviors of tuna swarm which consist of spiral foraging and parabolic foraging. [19]. The paper only focuses on spiral foraging behavior of tuna swarm which can keep a balance between global exploration and local exploitation on the premise of ensuring speed of convergence. The mathematical model of spiral foraging behavior can be expressed as follows:

$$x_i^{g+1} = \begin{cases} \alpha_1\left(X^b + \beta \times \left|X^b - x_i^g\right|\right) + \alpha_2 \times x_i^g, i = 1 \\ \alpha_1\left(X^b + \beta \times \left|X^b - x_i^g\right|\right) + \alpha_2 \times x_{i-1}^g, i = 2,3,\cdots,NP \end{cases} \# (11)$$

$$\begin{aligned} \alpha_1 &= p + (1-p) \times \frac{g}{G} \\ \alpha_2 &= (1-p) - (1-p) \times \frac{g}{G} \# \end{aligned} \quad (12)$$

$$\begin{aligned} \beta &= e^{ql} \times \cos(2\pi q) \\ l &= e^{3\cos\left(((G-g+1)/g)\pi\right)} \# \end{aligned} \quad (13)$$

where α_1 represents a weight coefficient which is applied to control the tendency of current individuals to get close to optimal individual, α_2 represents a weight coefficient which is used to control the tendency of current individuals to approach to previous individual, p represents a fix parameter used to determine the extent that tuna go after the optimal individual and previous individual in the early stage, and q represents a uniformly distributed within the range [0,1].

In the DBO algorithm, the position update approach in the condition where dung beetles conduct dung beetle breeding behavior are based on current optimal position which could decrease the population diversity and premature convergence to the local optimal. Meanwhile, the position update approach in the condition where dung beetles conduct dung beetle foraging behavior are strictly controlled by the current global optimal position which the capability of global exploration could decrease during the iteration process and it is also easy to get trapped into the local optimal position.

It can be seen from equation (11) that β is significant for tuna to conduct spiral foraging strategy. Thus, the paper introduces the specific coefficient into the DBO algorithm for the purpose of extending the searching capability of dung beetle populations, and balancing the global exploration and local exploitation on the premise of keep the population diversity. The updated equation for the mathematical model of dung beetle breeding behavior and foraging behavior can be expressed as follows:

$$x_i^{g+1} = X^* + \beta_1 \times \left(x_i^g - X^{Lb^*}\right) + \beta_2 \times \left(x_i^g - X^{Ub^*}\right) \# \quad (14)$$

$$x_i^{g+1} = x_i^g + \beta_3 \times \left(x_i^g - X^{Lb^b}\right) + \beta_4 \times \left(x_i^g - X^{Ub^b}\right) \# \quad (15)$$

The paper introduces two different sets of adaptive weight coefficients into the mathematical model of dung beetle stealing behavior to generate new position update formula. One of the sets of adaptive weight coefficients originates from spiral foraging strategy of the TSO. The updated formula can be expressed as follows:

$$x_i^{g+1} = \alpha_1 \times X^b + \alpha_2 \times S \times t \times (|x_i^g - X^*| + |x_i^g - X^b|) \# \quad (16)$$

In addition, the paper newly proposed a set of adaptive weight coefficients which is introduced for updating the equation (7). The new position update formula can be expressed as follows:

$$x_i^{g+1} = \varphi_1 \times X^b + \varphi_2 \times S \times t \times (|x_i^g - X^*| + |x_i^g - X^b|) \# \quad (17)$$

$$\varphi_1 = 1 - \frac{g}{G} \# \quad \varphi_2 = \frac{g}{G} \quad (18)$$

Adaptive weight coefficients are added into equation (7) in order to improve the ability to jump out of local optima, extend the search space and keep a balance between global exploration and local exploitation.

3.4 The Pseudo Code of EDBO

The Pseudo Code of EDBO is illustrated in the Algorithm 1. The EDBO can be outlined as follows: 1) utilize SPM chaotic mapping to initialize the dung beetle population and set the related parameters of the EDBO; 2) calculate the fitness values of the dung beetle individuals in the population according to the objective function; 3) update the position information of each dung beetle individual according to corresponding formula; 4) check whether each individual in the population is out of the boundary of the design space for the optimization; 5) update the current optimal solution and its corresponding fitness values; 6) repeat the above steps until the EDBO satisfies convergence criteria (g meets the upper limits of iterations); 7) output the global optimal solution and its corresponding value.

4. NUMERICAL EXPERIMENT AND ANALYSIS

The efficacy of the proposed EDBO algorithm is evaluated through a range of experiments on some benchmark functions in this section.

4.1 Benchmark Functions

In order to validate capacity for optimization of the EDBO, 10 benchmark functions were selected to conduct simulation experiment. The benchmark function can be divided into two different categories which include unimodal function, and multimodal function. Unimodal benchmark functions such as $f_1 \sim f_6$ listed in Table 1 which contain only one single global optimum solution are utilized to test the speed and exactness of convergence of the algorithm. Multimodal benchmark functions such as $f_7 \sim f_{10}$ listed in Table 2 which contain a global optimum solution and several local optimum solutions are used to gauge the performance of the algorithm to overstep the local optimum.

4.2 Comparison Algorithm and Experimental Parameters Settings

In this paper, several swarm intelligence optimization algorithms were selected for comparison to verify the robustness of the proposed the EDBO algorithm which include GWO [4], HHO [7], TSO [19], DBO [10]. Table 3 presents the parameter settings whose values were recommended in their respective paper. To ensure the fairness of the comparison between EDBO and other comparative algorithms, the experiments need to be implemented in the same environment where the size of population was $N = 30$ and the number of maximum iterations were $G = 500$. The experiment was conducted on Windows 10 operating system, 64-bit OS, Intel(R) Xeon(R) Silver 4210 CPU @ 2.20GHz, 96GB. The simulation software was Matlab 2021a.

Algorithm1 The Pseudo Code of the EGDBO algorithm.
Input: The size of population *N*, the number of maximum iterations *G*, the objective function *f*, the problem bounds Lb and Ub, the problem dimension D.
/* Initialization */
1: Initialize population $i = 1,2,\cdots,N$ and define relevant parameter of algorithm.
2: Calculate the fitness of every individual and obtain the optimal solution X^b.
/* **Main loop** */
3: **while** $(g \leq G)$ **do**
4: f**or** $i = 1{:}N$
/* **The ball-rolling behavior** */

if $i ==$ ball-rolling dung beetle
if $\delta_1 < 0.9$
Update the position according to Eq. (9)
else
Update the position according to Eq. (2)
end if
end if
/* **The breeding behavior** */
if $i ==$ brood ball
if $\delta_2 < 0.2$
Update the position according to Eq. (4)
else
Update the position according to Eq. (14)
end if
end if
/* **The foraging behavior** */
if $i ==$ small dung beetle
if $\delta_3 < 0.2$
Update the position according to Eq. (6)
else
Update the position according to Eq. (15)
end if
end if
/* **The stealing behavior** */
if $i ==$ thief
if $\delta_4 < 0.5$
Update the position according to Eq. (16)
else
Update the position according to Eq. (17)
end if
end if
end for
if the newly generated position is better than previous position
Accept the new position.
else
Accept the previous position.
end if
$g = g + 1$;
end while
Output: Optimal position X^b and its corresponding fitness value.

Table 1. Unimodal benchmark functions

Function name	Function	Dim	Range	f_{min}
Sphere	$f_1(x) = \sum_{i=1}^{n} x_i^2$	30	$[-100,100]$	0
Schwefel 2.22	$f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	$[-10,10]$	0

Function name	Function	Dim	Range	f_{min}
Schwefel 1.2	$f_3 = \sum_{i=1}^{n}\left(\sum_{j=1}^{i} x_j\right)^2$	30	$[-100,100]$	0
Schwefel 2.21	$f_4(x) = max\{\lvert x_i\rvert, 1 \le i \le n\}$	30	$[-100,100]$	0
Cjgar	$f_5(x) = x_1^2 + 10^6 \sum_{i=2}^{n} x_i^2$	30	$[-100,100]$	0
Zakharov	$f_6(x) = \sum_{i=1}^{n} x_i^2 + \left(\sum_{i=1}^{n} ix_i\right)^2 + \left(\sum_{i=1}^{n} ix_i\right)^4$	30	$[-5,10]$	0

Table 2. Multimodal benchmark functions

Function name	Function	Dim	Range	f_{min}
Rastrigin	$f_7(x) = \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i) + 10]$	30	$[-5.12,5.12]$	0
Ackley	$f_8(x) = -20exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - exp\left(\frac{1}{n}\sum_{i=1}^{n} 10\cos(2\pi x_i)\right) + 20 + e$	30	$[-32,32]$	0
Griewank	$f_9(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	$[-600,600]$	0
Apline	$f_{10}(x) = \sum_{i=1}^{n}\lvert x_i\sin(x_i) + 0.1x_i\rvert$	30	$[-10,10]$	0

Table 3. The parameters setting of the compared algorithms

Algorithm	Parameter	Setting
GWO	a	Uniformly lowered from 2 to 0
HHO	Interval of E_0	$[-1,1]$
TSO	z and a	0.05 and 0.7
DBO	k, λ, b, and S	0.1, 0.1, 0.3, and 0.5
EDBO	h_1, h_2, and p	$-\pi$, π, and 0.85

4.3 Experimental Results and Discussion

It should be noted that the proposed EDBO and several compared algorithms need to be repeated for 30 times independently for the purpose of reducing the influence of randomness. This section selects three different statistical tools as the performance indicators which include best-seeking optimum (Best), the mean value (Mean), and the standard deviation (Std). The mathematical expressions are defined as follows:

$$\text{Best} = min\{f_1, f_2, \cdots, f_R\} \# \quad (19)$$

$$\text{Mean} = \frac{1}{R}\sum_{i=1}^{R} f_i \#(18)$$

$$\text{Std} = \sqrt{\frac{1}{R-1}\sum_{i=1}^{R}(f_i - \text{Mean})^2} \# \quad (20)$$

where R represents the number of independent runs for optimization experiment, and f_i represents the optimum of the i^{th} run.

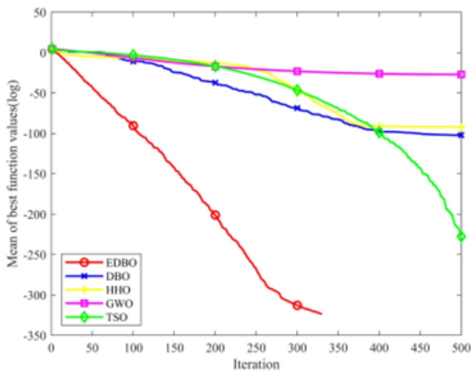
The experimental results of the three performance indicators for six unimodal benchmark functions and four multimodal benchmark functions after 30 independent runs are showed in Table 4. The EDBO outperforms other compared algorithms. It can be seen that for the unimodal benchmark functions f_1~f_6, the EDBO can obtain the theoretical optimal value with the mean value and standard deviation of 0, which demonstrates that the EDBO possesses strong robustness and stability. For multimodal benchmark functions f_7~f_9, the performance of the EDBO, DBO, HHO, and TSO are comparable and the three different performance indicators are significantly better than GWO. For f_{10}, the EDBO converges to theoretical optimal value with mean value and standard deviation of 0. The EDBO proposed in the paper is an improvement based on the DBO. Thus, it is necessary to compare the performance between the EDBO and the DBO. It can be seen from the Table 4 that the EDBO is significantly better than the DBO in terms of the three performance indicators which suggests that the amelioration strategy used in the paper can effectively enhance the performance of the classical DBO.

Table 4. The experimental results of unimodal and multimodal benchmark functions

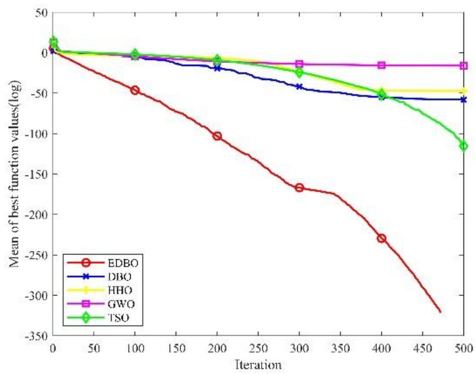
Function	Indicator	EDBO	DBO	HHO	GWO	TSO
f_1	Best	0.00E+00	1.03E-159	3.86E-114	2.29E-29	7.77E-274
	Mean	0.00E+00	6.24E-103	1.31E-92	9.76E-28	2.99E-228
	Std	0.00E+00	3.36E-102	7.17E-92	1.64E-27	0.00E+00
f_2	Best	0.00E+00	2.42E-82	3.99E-59	4.90E-18	2.46E-134
	Mean	0.00E+00	8.88E-59	3.51E-48	9.64E-17	4.62E-116
	Std	0.00E+00	4.38E-58	1.88E-47	6.42E-17	2.35E-115
f_3	Best	0.00E+00	8.43E-136	6.77E-96	3.11E-08	7.89E-257
	Mean	0.00E+00	3.05E-48	9.88E-70	9.67E-06	5.72E-215
	Std	0.00E+00	1.67E-47	5.28E-69	1.70E-05	0.00E+00
f_4	Best	0.00E+00	2.46E-80	5.36E-57	9.58E-08	2.95E-139
	Mean	0.00E+00	7.62E-51	1.84E-49	6.16E-07	1.27E-115
	Std	0.00E+00	3.20E-50	5.79E-49	3.96E-07	3.57E-115
f_5	Best	0.00E+00	2.28E-156	1.49E-113	1.77E-23	6.63E-261
	Mean	0.00E+00	7.01E-108	1.73E-85	6.38E-22	8.70E-219
	Std	0.00E+00	3.84E-107	9.48E-85	7.93E-22	0.00E+00
f_6	Best	0.00E+00	4.04E-120	7.34E-86	9.50E-11	2.80E-229
	Mean	0.00E+00	1.67E-19	6.80E-41	2.00E-07	8.67E-192
	Std	0.00E+00	9.14E-19	3.72E-40	5.20E-07	0.00E+00
f_7	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	0.00E+00	1.59E+00	0.00E+00	2.42E+00	0.00E+00
	Std	0.00E+00	6.95E+00	0.00E+00	3.09E+00	0.00E+00
f_8	Best	8.88E-16	8.88E-16	8.88E-16	6.48E-14	8.88E-16
	Mean	8.88E-16	8.88E-16	8.88E-16	1.01E-13	8.88E-16
	Std	0.00E+00	0.00E+00	0.00E+00	1.92E-14	0.00E+00
f_9	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	0.00E+00	0.00E+00	0.00E+00	6.53E-03	0.00E+00
	Std	0.00E+00	0.00E+00	0.00E+00	1.10E-02	0.00E+00
f_{10}	Best	0.00E+00	9.08E-82	3.16E-61	2.57E-16	5.97E-131
	Mean	0.00E+00	1.09E-04	4.59E-51	6.66E-04	1.01E-34
	Std	0.00E+00	2.62E-04	1.75E-50	7.34E-04	5.49E-34

The convergence curves of the proposed EDBO and other four compared algorithms for six unimodal benchmark functions and four multimodal benchmark functions are displayed in Fig. 4 with the purpose of comparing the convergence accuracy and rate of different algorithms more intuitively.

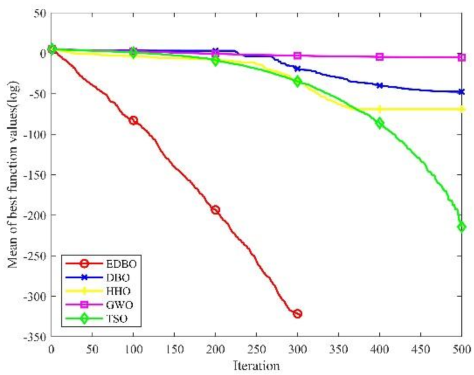
In Fig. 4, the horizontal axis denotes the number of iterations and the vertical axis represents the average fitness values which have been expressed in logarithmic form with the base of 10 to better display the trend of convergence. It can be seen from Fig. 4 that the proposed EDBO exhibits faster convergence rate and higher convergence accuracy than the compared algorithms. By observing several convergence curves based on unimodal benchmark functions f_1 ~ f_6 , it can be seen that the convergence performance of the EDBO is far superior to that of the compared algorithms and the convergence curves present a near-linear descent to the theoretical optimal values or approximate theoretical optimal values which indicate that the EDBO can determine the region immediately where the global optimal solution may exist and move towards it. Next, it can be summarized from the convergence curves of the multimodal benchmark functions that the EDBO outperforms the compared algorithms on the part of the convergence accuracy and rate. The convergence curves of function f_{10} is similar to that of unimodal benchmark functions. In the convergence curves of function f_7~f_9, the EDBO converges with a sharp decline in a straight line to obtain theoretical optimal values which suggest that the proposed EDBO can jump out of the local optima effectively. Finally, by comparing the convergence curves of EDBO and the convergence curves of the DBO, it can be seen that the convergence accuracy and rate of EDBO are significantly better than that of the DBO, which indicates that the modification tactic proposed in the paper are useful for enhancing the convergence performance of the DBO.



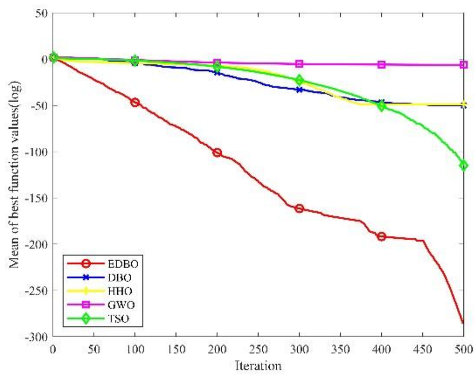
(a) f_1 average convergence curve



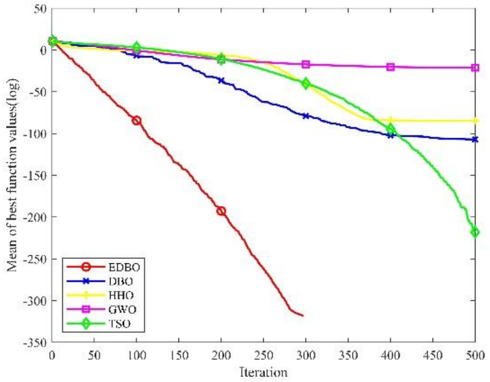
(b) f_2 average convergence curve



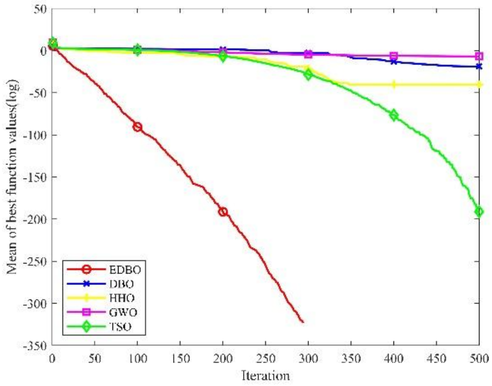
(c) f_3 average convergence curve



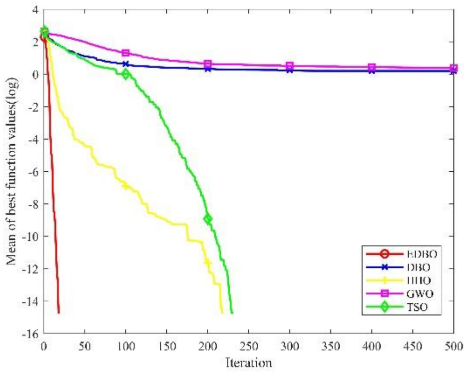
(d) f_4 average convergence curve



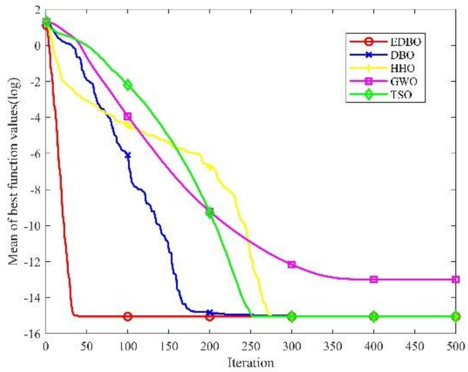
(e) f_5 average convergence curve



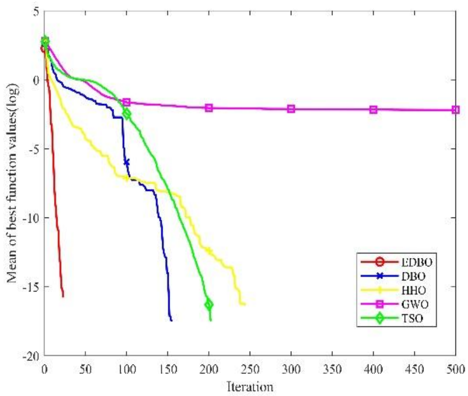
(f) f_6 average convergence curve



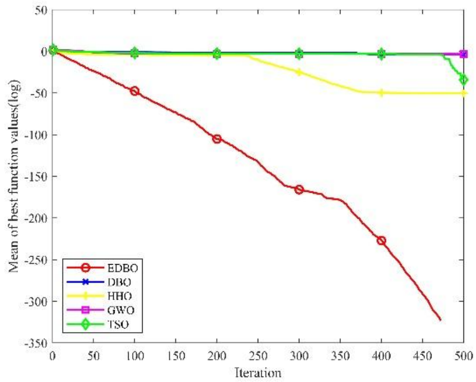
(g) f_7 average convergence curve



(h) f_8 average convergence curve



(i) f_9 average convergence curve



(j) f_{10} average convergence curve

Fig. 4. Average convergence curve of the benchmark function

Table 5. α-values of Wilcoxon sign-rank test

Function	EDBO vs			
	DBO	**HHO**	**GWO**	**TSO**
f_1	1.21E-12	1.21E-12	1.21E-12	1.21E-12
f_2	1.21E-12	1.21E-12	1.21E-12	1.21E-12
f_3	1.21E-12	1.21E-12	1.21E-12	1.21E-12
f_4	1.21E-12	1.21E-12	1.21E-12	1.21E-12
f_5	1.21E-12	1.21E-12	1.21E-12	1.21E-12
f_6	1.21E-12	1.21E-12	1.21E-12	1.21E-12
f_7	1.61E-01	N/A	4.47E-12	N/A
f_8	N/A	N/A	1.17E-12	N/A
f_9	N/A	N/A	6.62E-04	N/A
f_{10}	1.21E-12	1.21E-12	1.21E-12	1.21E-12
$+/-/=$	$7/0/3$	$7/0/3$	$10/0/0$	$7/0/3$

In addition, the Wilcoxon sign-rank test with a significant level $\alpha = 0.05$ is included in the paper for finding out whether the proposed EDBO has a significant performance difference compared with the classical DBO, HHO, GWO, and TSO on the basis of 30 independent runs in the 10 benchmark functions. The results of Wilcoxon sign-rank test are displayed in Table 5.

The symbol '+' indicates the EDBO is superior to the compared algorithms; the symbol '-' indicates the EDBO is inferior to the compared algorithms; the symbol '=' indicates the EDBO is similar to the compared algorithms, and 'N/A' presents the EDBO and the compared algorithm have a comparable performance. According to Table 5, the EDBO differs significantly from DBO, HHO, GWO, and TSO for function f_1~$f_6$and f_{10}. There is no significant performance difference between the EDBO and DBO, HHO, and TSO for function f_7~f_9. In general, the convergence performance of EDBO is superior to that of the compared algorithms.

5. CONCLUSION

This paper proposes the EDBO algorithm to address the existing drawbacks of the classical DBO algorithm which include the poor capability to conduct global search and escape the local optimum. The SPM chaotic mapping model is utilized to initialize the population and keep diversity. The individual update strategy in Golden Sine Algorithm is introduced in the EDBO to enhance the ability of search. Additionally, the spiral foraging strategy in tuna swarm optimization is implemented to modify the position update equation for the mathematical model of dung beetle breeding and foraging behavior with the purpose of extending the search range and keeping a balance the global exploration and local exploitation. Finally, the two different sets of adaptive weight parameters are added into the position update equation for the mathematical model of dung beetle stealing behavior in order to increase the likelihood of jumping out of the local optima and enhance the later search ability. The performance of the EDBO is compared with four different swarm intelligence optimization algorithms, including DBO, HHO, GWO, and TSO, on six unimodal benchmark functions and four multimodal benchmark functions. The three performance indicators, including 'Best', 'Mean', and 'Std', and Wilcoxon sign-rank test are used to verify the effectiveness of modification tactics. According to the outcomes of the simulation experiment, the EDBO is superior to other compared algorithms in terms of convergence accuracy and speed and meanwhile enhances the capability of jumping out of the local optima.

Nevertheless, the current research is still insufficient. In the future, we will consider further improving the convergence performance of the EDBO so that it can be utilized to handle complex multi-objective optimization problem and engineering optimization problems.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Tang J, Liu G, Pan Q. A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends[J].

IEEE/CAA Journal of Automatica Sinica, 2021;8(10):1627-1643.

2. Al-Betar M A, Awadallah M A, Faris H, et al. Natural selection methods for grey wolf optimizer[J]. Expert Systems with Applications. 2018;113:481-498.
3. Kennedy J, Eberhart R. Particle swarm optimization[C]//Proceedings of ICNN'95-international conference on neural networks. IEEE. 1995;4:1942-1948.
4. Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer[J]. Advances in engineering software, 2014;69:46-61.
5. Yang XS, Deb S. Cuckoo search: recent advances and applications[J]. Neural Computing and applications. 2014;24: 169-174.
6. Naruei I, Keynia F. Wild horse optimizer: A new meta-heuristic algorithm for solving engineering optimization problems[J]. Engineering with computers. 2022; 38(Suppl 4):3025-3056.
7. Heidari AA, Mirjalili S, Faris H, et al. Harris hawks optimization: Algorithm and applications[J]. Future generation computer systems. 2019;97:849-872.
8. Dehghani M, Hubálovský Š, Trojovský P. Northern goshawk optimization: A new swarm-based algorithm for solving optimization problems[J]. IEEE Access. 2021;9:162059-162080.
9. Naruei I, Keynia F. A new optimization method based on COOT bird natural life model[J]. Expert Systems with Applications. 2021;183:115352.
10. Xue J, Shen B. Dung beetle optimizer: A new meta-heuristic algorithm for global optimization[J]. The Journal of Supercomputing. 2023;79(7):7305-7336.
11. Zeng C, Qin T, Tan W, et al. Coverage Optimization of Heterogeneous Wireless Sensor Network Based on Improved Wild Horse Optimizer[J]. Biomimetics. 2023; 8(1):70.
12. Han X, Xu Q, Yue L, et al. An improved crow search algorithm based on spiral search mechanism for solving numerical and engineering optimization problems[J]. IEEE Access. 2020;8:92363-92382.
13. Liu Q, Li N, Jia H, et al. A hybrid arithmetic optimization and golden sine algorithm for solving industrial engineering design problems[J]. Mathematics. 2022; 10(9):1567.
14. Liu M, Zhang Y, Guo J, et al. An Adaptive Lion Swarm Optimization Algorithm Incorporating Tent Chaotic Search and Information Entropy[J]. International Journal of Computational Intelligence Systems. 2023;16(1):39.
15. Si Q, Li C. Indoor Robot Path Planning Using an Improved Whale Optimization Algorithm[J]. Sensors. 2023;23(8): 3988.
16. Shouyu LI, Qing HE, Nisuo DU. Butterfly optimization algorithm for chaotic feedback sharing and group synergy[J]. Journal of Frontiers of Computer Science & Technology. 2022;16(7):1661.
17. Wang Y, Chen S, Wang Y. Chaos encryption algorithm based on kent mapping and AES combination[C]//2018 international conference on network, communication, computer engineering (NCCE 2018). Atlantis Press. 2018;588-592.
18. Tanyildizi E, Demir G. Golden sine algorithm: A novel math-inspired algorithm[J]. Advances in Electrical and Computer Engineering. 2017;17(2): 71-78.
19. Xie L, Han T, Zhou H, et al. Tuna swarm optimization: A novel swarm-based metaheuristic algorithm for global optimization[J]. Computational intelligence and Neuroscience. 2021; 2021:1-22.

Peer-review history:
The peer review history for this paper can be accessed here:
https://www.sdiarticle5.com/review-history/102486