

Article

Not peer-reviewed version

Three-Dimensional Unmanned Aerial Vehicle Path Planning in Simulated Rugged Mountainous Terrain Using Improved Enhanced Snake Optimizer (IESO)

[Wuke Li](#) , [Kongwen \(Frank\) Zhang](#) ^{*} , [Qi Xiong](#) , [Xiaoxiao Chen](#)

Posted Date: 17 June 2024

doi: 10.20944/preprints202406.1146.v1

Keywords: UAV; path planning; enhanced snake optimizer; dynamic boundary-based opposition learning; test function



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Three-Dimensional Unmanned Aerial Vehicle Path Planning in Simulated Rugged Mountainous Terrain Using Improved Enhanced Snake Optimizer (IESO)

Wuke Li ¹, Kongwen (Frank) Zhang ^{2,*} , Qi Xiong ³  and Xiaoxiao Chen ¹

¹ International College, Hunan University of Arts and Science, Changde, Hunan, 415000, China; liwuke@huas.edu.cn (W.L.), chenxiaoxiao920@huas.edu.cn (X.C.)

² School of Computing, University of the Fraser Valley, Abbotsford, British Columbia, V2S 7M7, Canada; Frank.Zhang@ufv.ca

³ School of Automation Science and Engineering, Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, 710049, China; xiongqi@stu.xjtu.edu.cn

* Correspondence: Frank.Zhang@ufv.ca; Tel.: 604-504-7441 x4401

Abstract: The challenging terrain and deep ravines characteristic of mountainous regions often result in slower path planning and suboptimal flight paths for Unmanned Aerial Vehicles (UAVs) when traditional meta-heuristic optimization algorithms are employed. This study investigates the application of an Improved Enhanced Snake Optimizer (IESO) for three-dimensional path planning in a simulated rugged mountainous terrain. The initialization process in the enhanced snake optimizer is refined by integrating the Chebyshev chaotic map. Additionally, a non-monotonic factor is introduced to modulate the "temperature," and a boundary condition is incorporated into the dynamic opposition learning mechanism. These modifications collectively reduce the likelihood of population convergence to local optima during optimization. The feasibility of IESO is validated through time complexity and global convergence analyses. Comparative simulation experiments benchmarked IESO against five state-of-the-art optimization algorithms across test functions and path-planning simulated scenarios. The experimental results indicate that the enhanced algorithm achieves superior optimization precision, faster convergence speeds, and improved quality in the planned trajectories by 30%.

Keywords: UAV; path planning; enhanced snake optimizer; dynamic boundary-based opposition learning; test function

1. Introduction

The complex mountainous terrain, characterized by significant elevation changes and limited transportation options, often renders certain areas difficult to access using conventional observation, surveillance, and logistical support methods. Due to their agility and flexibility, unmanned aerial vehicles (UAVs) have become extensively utilized in these challenging environments. Consequently, high-quality and rapid flight path planning in such complex terrains is critically important for enhancing the efficiency of UAV missions [1,2].

Recent research in UAV path planning has introduced many different methodologies, including meta-heuristic optimization algorithms (MAs) [3], the quality-oriented hybrid A* and Q-learning algorithm [4], the artificial potential field [5], trajectory planning with matrix alignment Dijkstra's algorithm [6,7]. Flores-Caballero et al. [8] approached path planning as a constrained global optimization problem, utilizing MAs in conjunction with the A* algorithm. However, their objective function was overly simplistic, concentrating exclusively on flight distance, and consequently, it did not achieve optimal practical outcomes. Shao et al. [9] proposed a hierarchical trajectory optimization framework that combines an adaptive particle swarm optimization algorithm with the Gaussian pseudospectral method. This methodology accounted for the UAV dynamics model and state constraints, thereby enabling the minimization of flight time. However, the simulated environment was overly simplistic and not applicable to complex terrains, such as mountainous regions or ravines. For large-scale collaborative searches, Chen et al. [10] utilized the ant colony optimization algorithm to efficiently plan paths for UAV swarms, reducing task completion times. Despite its rapid optimization, the algorithm exhibited limited convergence precision.

UAV path planning necessitates considering rugged terrain environments, such as mountainous regions, flight safety, and UAV-specific conditions, rendering it a quintessential multi-objective optimization problem. A literature review reveals that current research predominantly focuses on MAS due to their global search capabilities, which effectively address UAV motion constraints and path planning requirements, particularly in complex environments characterized by rugged mountainous terrain.

In this study, we examined the Enhanced Snake Optimizer (ESO) proposed by Yao et al. [11], which builds upon the original Snake Optimizer (SO) developed by Hashim and Hussien [12]. The ESO incorporates a randomized population initialization and mirroring imaging. This enhancement has been validated using classical test and engineering design functions, demonstrating its applicability to UAV path planning. However, in line with the No Free Lunch theorem [13], which posits that no single algorithm is universally optimal for all optimization problems, we acknowledge the limitations of ESO.

We propose three major ESO revisions at three stages that significantly improve optimization results, particularly in rugged terrains. Therefore, we name our method Improved Enhanced Snake Optimizer (IESO). First, the original SO employs a simple normal distribution random function to initialize the "population," which consists of the initial guess points. We replaced this with a Chebyshev chaotic map to improve the diversity and quality of the initial population. Second, we introduced a non-monotonic coefficient to adjust the "temperature," which controls the search merge and split processes. This modification addresses the SO's persistent "cold temperature" state, balancing global exploration and local exploitation. Third, we added a boundary value to address occasional anomalies from the dynamic opposition learning mechanism, a key enhancement of ESO over SO.

With these three major revisions to ESO, we tested IESO against five state-of-the-art optimization algorithms, namely Enhanced Snake Optimizer (ESO) [11], Snake Optimizer (SO) [12], Golden Jackal Optimization (GJO) [14], Honey Badger Algorithm (HBA) [15], Grey Wolf Optimizer (GWO) [16], demonstrating superior performance in precision, convergence speed and planned trajectory quality.

2. Methodologies: Improved Enhanced Snake Optimizer (IESO) Algorithm and Simulation Design

This section is organized into four subsections: 2.1 briefly reviews the Snake Optimizer (SO). 2.2 presents a concise overview of the Enhanced Snake Optimizer (ESO). 2.3 offers a detailed explanation of our Improved Enhanced Snake Optimizer (IESO). 2.4 discusses the design of the simulation experiments and the evaluation criteria.

2.1. Basic Concepts of Snake Optimizer (SO)

The Snake Optimizer (SO) was initially proposed by Hashim and Hussien [12] in 2022. It is an optimization algorithm inspired by the snake mating process, which is influenced by "temperature" (mode switch) and "food" (change position) availability. Snakes mate only when the temperature is low, and food is abundant. When food is scarce, the SO algorithm prioritizes the food search, representing the exploration phase (global search). Once food is sufficient, the algorithm progresses to the exploitation phase (local search). During "hot" temperatures ($T > 0.6$), mating is deferred. When temperatures are low ($T < 0.6$), snakes engage in mating, based on a random probability P_r (0, 1), whether to mate with the opposite sex or engage in same-sex aggression. When $P_r > 0.6$, mating behaviour occurs between sexes; while $P_r < 0.6$, males compete for mating rights in a fight mode, followed by a mating mode. Successful mating results in females laying eggs and hatching new snakes, thereby renewing the population (add more search points).

In SO, the "population" (N) represents the search points. The initial positions are generated using a random uniform distribution function, as defined in Equation 1. The population is then equally divided into two groups: males and females.

$$X_i = X_{min} + rand_i \times (X_{max} - X_{min}) \quad (1)$$

In SO, The "temperature" (T) determines the switch between global and local search modes. It is defined by Equation 2, where t is the current iteration and T is the maximum iteration and the same for the rest of this manuscript.

$$Temp = \exp^{-t/T} \quad (2)$$

In SO, a "food" parameter represents the feature position. The algorithm performs a broad, global search when no food is available. If food is present and the temperature is "hot", the algorithm refines the search point positions by eating food. When the temperature is "cold," the algorithm enters the exploitation phase, where males and females select optimal mates, resulting in convergence and repositioning new search points. The food quantity (Q) is defined in Eq. 3.

$$Q = c_1 \exp^{(t-T)/T}, c_1 = 0.5 \quad (3)$$

When $Q < 0.25$, the snakes search for food by selecting any random position and updating their position for it, which is called the exploration phase. The male version is presented as Eq. 4, and A_m is defined as Eq. 5. The female version is identical except for replacing the subscript m to f .

$$X_{i,m}(t+1) = X_{rand,m}(t) \pm c_2 \times A_m \times ((X_{max} - X_{min}) \times rand + X_{min}), c_2 = 0.05 \quad (4)$$

$$A_m = \exp^{-f_{rand,m}/f_{i,m}} \quad (5)$$

When $Q > 0.6$ and $T > 0.6$, the snake searches for food, keeping the global search. And it is presented in Eq. 6.

$$X_{i,j}(t+1) = X_{food} \pm c_3 \times Temp \times rand \times ((X_{food} - X_{i,j}(t))), c_3 = 2, \quad (6)$$

where $X_{i,j}$ is the i^{th} snake position (male or female), X_{food} is the position of the food.

When $Q > 0.6$ and $T < 0.6$, the snake will enter the flight mode (FM) or mating mode (MM), which means the search points are merging and splitting, which are presented in Eq. 7 and 8.

$$X_{i,m}(t+1) = X_{i,m}(t) + c_3 \times FM \times rand \times Q \times (X_{best,f} - X_{i,m}(t)), c_3 = 2 \quad (7)$$

$$X_{i,f}(t+1) = X_{i,f}(t) + c_3 \times FF \times rand \times Q \times (X_{best,m} - X_{i,f}(t)), c_3 = 2, \quad (8)$$

where $X_{i,m}$ is the i^{th} male position, $X_{best,f}$ refers to the best female position. FM and FF are the fighting ability of a snake (male and female), and it is defined in Eq. 9 and 10.

$$FM = \exp^{-f_{best,f}/f_i} \quad (9)$$

$$FF = \exp^{-f_{best,m}/f_i} \quad (10)$$

The mating mode will create new search points and replace the worst current ones. The equations are given in Eq. 11 and 12. The mating abilities are given in Eq. 13 and 14.

$$X_{i,m}(t+1) = X_{i,m}(t) + c_3 \times M_m \times rand \times Q \times (X_{i,f}(t) - X_{i,m}(t)), c_3 = 2 \quad (11)$$

$$X_{i,f}(t+1) = X_{i,f}(t) + c_3 \times M_f \times rand \times Q \times (X_{i,m}(t) - X_{i,f}(t)), c_3 = 2 \quad (12)$$

$$M_m = \exp^{-f_{i,f}/f_{i,m}} \quad (13)$$

$$M_f = \exp^{-f_{i,m}/f_{i,f}} \quad (14)$$

The replace functions are given in Eq. 15 and 16.

$$X_{worst,m} = X_{min} + rand \times (X_{max} - X_{min}) \quad (15)$$

$$X_{worst,f} = X_{min} + rand \times X_{max} - X_{min} \quad (16)$$

2.2. Enhanced Snake Optimizer (ESO)

The Enhanced Snake Optimizer (ESO) was proposed by Yao et al. in 2023 [11]. The main modifications are the dynamic coefficients. First, the food quantity initialization function opens the fixed coefficient $c_1 = 0.5$ to a dynamic updated c_1^{new} , given as Eq. 17.

$$c_1^{new} = c_1 + \frac{1}{10} \times \cos\left(r_1^4 \times \frac{\pi}{2}\right), \quad (17)$$

where r_1 is a random number between (0,1).

Then, the fixed temperature coefficient $c_2 = 0.05$ to a new dynamic c_2^{new} as shown in Eq. .

$$c_2^{new} = c_2 + \frac{1}{1000} \times \cos\left(r_2^4 \times \frac{\pi}{2}\right), \quad (18)$$

where r_2 is a random number between (0,1). And, the fixed $c_3 = 2$.

Another major enhancement is the adoption of mirror imaging strategy [17] and opposition-based learning [18]. The upper boundary (UB) and lower boundary (LB) were added and are presented in Eq. 19 and 20.

$$\frac{(UB + LB)/2 - X}{X^* - (UB + LB)/2} = \frac{H}{H^*}, \quad (19)$$

where H^* and X^* are the mirror positions of H and X .

$$X^* = \frac{UB + LB}{2} + \frac{UB + LB}{2\delta} - \frac{X}{\delta}, \quad (20)$$

where $\delta = \frac{H}{H^*}$.

2.3. Improved Enhanced Snake Optimizer (IESO)

In our Improved Enhanced Snake Optimizer (IESO), we have implemented three significant enhancements based on the Enhanced Snake Optimizer (ESO). First, we adopted Chebyshev Population Initialization to initialize the snake's initial population. Second, we introduced a non-monotonic factor to modulate the temperature. Third, we applied a Dynamic Boundary-based Opposition Learning (DBOL) approach to define the upper and lower boundaries. The general flowchart is illustrated in Figure 1.

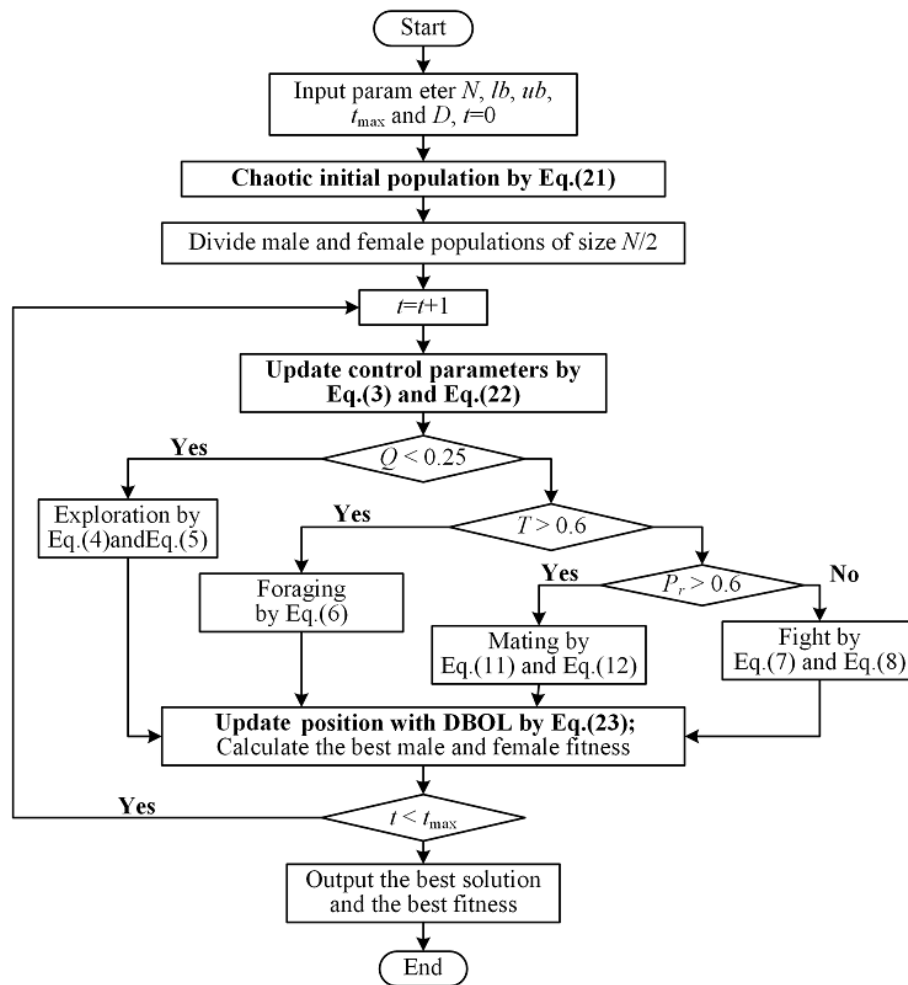


Figure 1. IESO Flowchart

2.3.1. Chebyshev Population Initialization

The SO and ESO randomly populate the initial population, leading to unstable target accuracy and varying convergence speeds across multiple searches. Achieving a more evenly distributed initial population within the solution space enhances the probability of locating the optimal value. Chaotic search, renowned for its randomness, ergodicity, and uniqueness, is preferred over random search strategies for generating initial populations due to its thorough coverage of the solution space. Therefore, this paper employs the Chebyshev chaotic map [19] for initializing the snake population to ensure a more comprehensive occupation of the entire solution space. The mathematical model of the Chebyshev chaotic map is as follows Eq. 21:

$$X_i(t+1) = \cos\left(t \cos^{-1}(X_i(t))\right) \quad (21)$$

This comparison, illustrated in Figure 2a,b, demonstrates that the Chebyshev chaotic map produces a more uniformly distributed initial population. The Chebyshev chaotic map exhibits enhanced search capabilities, particularly when solutions to constrained problems are situated at the boundaries of the search space.

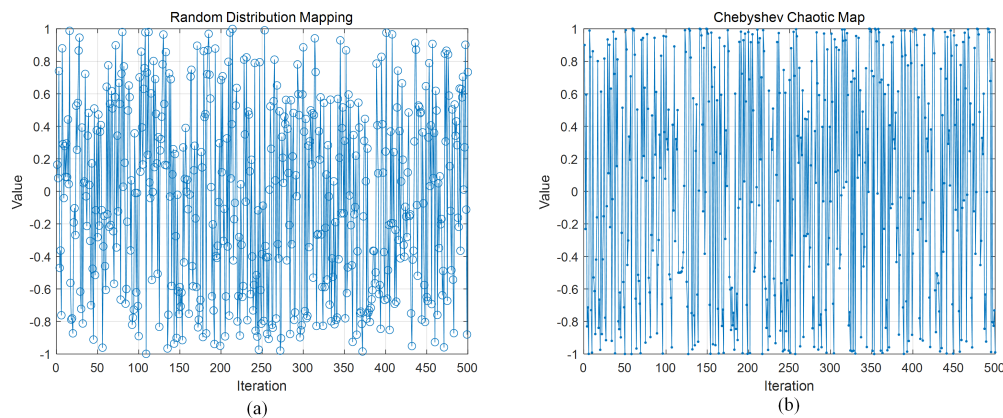


Figure 2. Initial value distribution by (a) Normal Random distribution and (b) Chebyshev Chaotic map.

2.3.2. Non-Monotone Temperature Factor

In SO and ESO, the temperature factor influences the behaviour of the snake swarm in a monotonic manner. Figure 2 demonstrates that the temperature factor solely depends on the number of iterations, with a threshold value of approximately 220. Before this threshold, the temperature coefficient T consistently remains above 0.6, indicating a "warm" state and foraging activity. Conversely, as the optimization process progresses into its later stages, the temperature shifts to a "cold" state and remains "cold." This transition indicates that after a significant number of iterations, both SO and ESO are prone to local convergence failure.

To address this issue, we introduce a non-monotonic temperature factor by incorporating a random coefficient a into each iteration's temperature coefficient T . This adjustment allows the modified temperature coefficient to balance more effectively between local development modes, as shown in Equation 22.

$$T = a * e^{-\frac{t}{t_{\max}}} \quad (22)$$

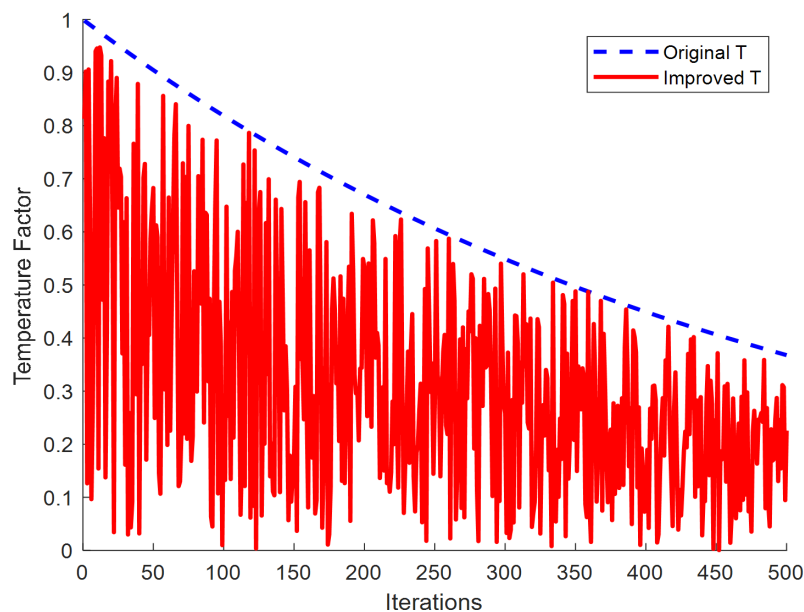


Figure 3. Temperature factor.

2.3.3. Dynamic Boundary-Based Opposition Learning (DBOL)

After carefully examining the mirror imaging strategy used in ESO [17] and the original opposition-based learning (OBL) [18], it is evident that the ESO mirror method effectively enhances the diversity and quality of the population by calculating the opposition solutions for the current position and then selecting the optimal solution.

However, the performance exhibits a certain degree of uncertainty. While opposition solutions may have greater potential for global optimization, the fixed search upper and lower boundaries limit the ability to estimate the direction of new solutions. We propose a Dynamic Boundary-based Opposition Learning (DBOL) to address this issue. DBOL leverages the fact that elite individuals carry more effective information than ordinary ones to refine the opposition process. It fully utilizes the valuable experiences of historical individuals while maintaining a global perspective for new individuals. Through greedy selection, a superior solution is chosen between the current and opposition solutions to participate in the next iteration. The DBOL defines the boundary as follows: Eq. 23:

$$X_i^{j*} = r_1 (UB_i^j + LB_i^j) - X_i^j, \quad (23)$$

where r_1 is a random value in the interval $(0, 1)$, $UB_i^j = \max(X_i^j)$, $LB_i^j = \min(X_i^j)$, and UB_i^j and LB_i^j are the upper and lower dynamic boundary in the j^{th} dimension, respectively.

2.4. Simulation Experiments Design and the Evaluation Criteria

2.4.1. Simulated Experiment

This study's simulation experiments are divided into two parts: The first part involves selecting the benchmark test functions with a search dimension ($D=1000$). The objective is to compare different algorithms' convergence speeds and optimization precision to ascertain whether IESO demonstrates a superior capability to evade local optima when dealing with large-scale optimization problems. The second part entails creating a simulated rugged mountainous terrain to test the path planning. In this experience, the performance of various algorithms is assessed when solving three-dimensional path planning problems based on the three optimization criteria (discussed in 2.4.2). It aims to further validate the feasibility of IESO in determining optimal flight paths within intricate, complex landscapes.

We selected thirteen testing functions shown in Table 1.

Table 1. Formula and parameters of 13 benchmark test functions.

Num.	Name	Formula	Range	f_{min}
f_1	Sphere's Function	$f(x) = \sum_{i=1}^n x_i^2$	[-100,100]	0
f_2	Schwefel's Problem 2.22	$f(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i$	[-10,10]	0
f_3	Schwefel's Problem 1.2	$f(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$	[-100,100]	0
f_4	Schwefel's Problem 2.21	$f(x) = \max_i \{ x_i , 1 \leq i \leq n \}$	[-100,100]	0
f_5	Generalized Rosenbrock's Function	$f(x) = \sum_{i=1}^{n-1} \left[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right]$	[-30,30]	0
f_6	Step Function	$f(x) = \sum_{i=1}^d x_i ^{i+1}$	[-100,100]	0
f_7	Quartic Function	$f(x) = \sum_{i=1}^n ix_i^4 + \text{random}[0, 1]$	[-1.28,1.28]	0
f_8	Generalized Schwefel's Problem 2.26	$f(x) = 418.9829n - \sum_{i=1}^n x_i \sin(\sqrt{ x_i })$	[-500,500]	-2094
f_9	Generalized Rastrigin's Function	$f(x) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$	[-10,10]	0
f_{10}	Ackley's Function	$f(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) - \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right) + 20 + e$	[-10,10]	0
f_{11}	Generalized Griewank's Function	$f(x) = 1 + \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right)$	[-10,10]	0
f_{12}	Generalized Penalized Function 1	$f(x) = \frac{\pi}{n} \left[10 \sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 (1 + 10 \sin^2(\pi y_{i+1})) + (y_n - 1)^2 \right] + \sum_{i=1}^n u(x_i, 10, 100, 4)$	[-10,10]	0
f_{13}	Generalized Penalized Function 2	$f(x) = 0.1 \left[\sin^2(3\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2 (1 + \sin^2(3\pi x_{i+1})) + (x_n - 1)^2 (1 + \sin^2(2\pi x_n)) \right] + \sum_{i=1}^n u(x_i, 5, 100, 4)$	[-10,10]	0

We selected five algorithms, namely GJO, GWO, HBA, ESO and SO and their parameters are listed in Table 2.

Table 2. Algorithm parameters.

Algorithms	Parameters
IESO	$T_{h1} = 0.25, T_{h2} = 0.6, T_p = 50$
ESO	$T_{h1} = 0.25, T_{h2} = 0.6, c_{start1} = 0.5, c_{start2} = 0.05, c_{start3} = 2; c_{end1} = c_{end2} = c_{end3} = 0.5$
SO	$T_{h1} = 0.25, T_{h2} = 0.6$
GJO	$c_1 = 1.5, \epsilon = 1.5$
HBA	$C = 2, \epsilon = 6$
GWO	$a \in [0, 2]$

We randomly generated the digital elevation model of a simulated rugged mountainous terrain, with map dimensions of 100 km by 150 km by 3 km. The maximum elevation difference is at least 2 km, representing a terrain with significant undulations and numerous ravines. Arbitrary no-fly zones were also added to increase the complexity. Flight waypoints were set to evaluate the effectiveness of the flight paths generated by different algorithms.

The experimental setup utilized a Windows 10, 64-bit operating system with an AMD Ryzen 7 4800H CPU running at 2.9 GHz and 16GB of RAM. The algorithms were implemented on the Matlab2020b platform.

2.4.2. Evaluation Criteria

Considering the intricate nature of UAV three-dimensional path planning, this study formulates the path planning objective function by incorporating factors such as flight distance, path smoothness, and flight safety. The shortest flight path is defined as Eq. 24

$$F_f = \sum_{k=1}^D \sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2 + (z_{k+1} - z_k)^2} \quad (24)$$

, where D represents the number of waypoints and (x_k, y_k, z_k) denotes the coordinates of the k^{th} waypoints. Ensuring smoothness involves minimizing abrupt changes in yaw angle and altitude. Given mountainous areas' rugged and diverse terrain, flight paths must adhere to maximum climb angles and rates. The path smoothness is given as Eq. 25, 26, and 27.

$$\varphi_k = \arctan\left(\frac{\|l_k l_{k+1}\|}{l_k l_{k+1}}\right) \quad (25)$$

$$\phi_k = \arctan\left(\frac{z_{k+1} - z_k}{\sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2}}\right) \quad (26)$$

$$F_e = \sum_{k=1}^{D-2} \varphi_k + \sum_{k=1}^{D-1} (\phi_k - \phi_{k-1}), \quad (27)$$

where l_k represents the distance between two waypoints. Eq 25 and 26 represent the yaw angle φ_k and pitch angle ϕ_k , respectively.

A meticulously designed path should prioritize the safe operation of UAVs. The objective function design must incorporate a safety margin, enabling UAVs to avoid obstacles or restricted no-fly zones. As shown in Figure 4 the obstacle within the airspace has a center coordinate of O_p and radius R_p . The perpendicular distance d_p from the flight node of the UAV to the obstacle should exceed the safety

distance S_f , ensuring the UAV operates outside designated shaded regions for safety. The flight safety formula is provided as Eq. 28

$$F_s = \begin{cases} 0, & d_p \geq S \\ S - d_p, & R_p < d_p < S \\ \infty, & d_p < R_p \end{cases} \quad (28)$$

By integrating these cost functions with weights, an objective function F for the multi-objective path planning problem is formed in Eq. 29

$$F = \omega_1 F_f + \omega_2 F_e + \omega_3 F_s \quad (29)$$

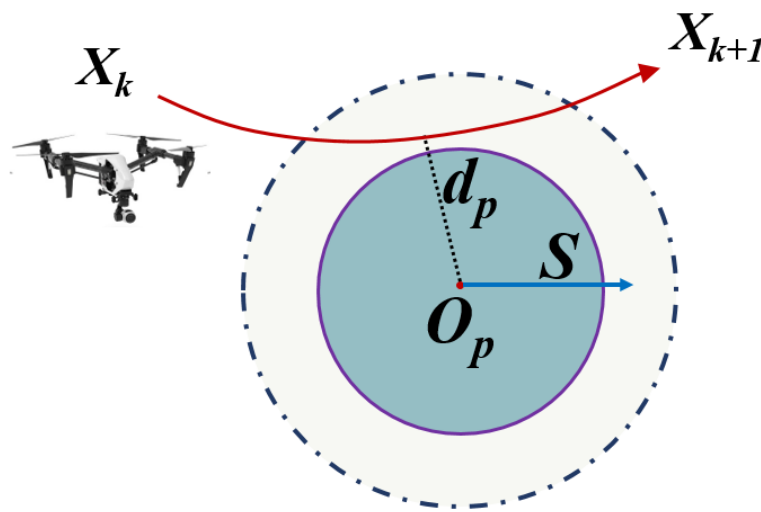


Figure 4. Flight safety to avoid obstacles or restricted no-fly zones.

3. Results and Discussion

The test results across six different algorithms are summarized, with the algorithm achieving the lowest average optimization result for each test function highlighted in bold, which is shown in Table 3. In the rank-sum test, the symbols "+", "-", and "=" denote that IESO optimization is superior, inferior, or equivalent, respectively, compared to other algorithms. Moreover, the statistical analysis indicates that, overall, the optimization performance of IESO surpasses that of most tested algorithms. The integration of multiple strategies significantly enhances the convergence precision and stability of IESO beyond that of the standard SO, preliminarily confirming the effectiveness of these strategies. For any given test function, none of the algorithms significantly outperforms IESO. The optimization curves for various algorithms on the test function with $D=1000$ are illustrated in Figure 4.

Table 3. Test results of 13 benchmark functions (D=1000)

Num.	GJO	SO	GWO	HBA	ESO	IESO
f_1	5.76E-55(7.15E-55)/+	9.32E-37(2.69E-36)/+	9.32E-28(2.82E-28)/+	5.00E-137(7.43E-137)/+	3.38E-197(0.00E+00)/+	0.00E-00(0.00E-00)
f_2	1.29E-32(1.18E-32)/+	6.02E-34(1.75E-33)/+	6.02E-17(1.27E-16)/+	4.20E-72(9.64E-72)/+	5.40E-92(1.45E-91)/+	7.52E-240(0.00E-00)
f_3	1.37E-17(2.54E-17)/+	9.89E-17(1.69E-17)/+	9.89E-06(6.81E-07)/+	2.81E-101(4.02E-101)/+	1.10E-150(2.94E-150)/+	0.00E-00(0.00E-00)
f_4	2.75E-16(7.85E-16)/+	7.57E-09(1.25E-08)/+	7.57E-07(7.28E-07)/+	2.47.00E-137(7.43E-137)/+	2.35E-197(6.41E-77)/+	7.88E-135(2.94E-134)
f_5	2.75E+01(7.85E-01)/+	7.57E-03(1.25E-03)/+	2.43E-01(7.28E-01)/+	2.47E-137(7.43E-137)/+	2.35E-197(6.41E-01)/+	1.21E-07(1.64E-06)
f_6	2.76E-04(3.82E-01)/+	3.10E-05(5.92E-05)/+	3.10E-28(2.82E-02)/+	2.56E-137(7.43E-137)/+	2.79E-197(7.51E-02)/+	9.52E-08(1.08E-08)
f_7	2.95E-55(1.66E-04)/+	4.96E-03(3.33E-04)/+	4.96E-28(2.82E-04)/+	2.85E-137(7.43E-137)/+	1.85E-197(1.88E-04)/+	1.82E-04(1.03E-04)
f_8	-3.70E+03(1.04E+03)/+	-9.70E+03(2.27E+03)/+	-9.70E-28(8.73E+02)/+	-9.02E-137(7.43E-137)/+	-8.62E+03(0.00E-00)/+	-1.08E+04(1.38E+03)
f_9	0.00E-00(0.00E-00)/=	0.00E-00(0.00E-00)/=	2.18E-02(5.68E-14)/+	0.00E-00(0.00E-00)/=	0.00E-00(0.00E-00)/=	0.00E-00(0.00E-00)
f_{10}	6.13E-55(1.83E-15)/+	1.87E-15(2.48E-15)/+	1.86E-13(1.48E-14)/+	8.88E-00(0.00E-00)/=	8.88E-16(0.00E-00)/=	8.88E-16(0.00E-00)
f_{11}	0.00E-00(0.00E-00)/=	0.00E-00(0.00E-00)/=	0.00E-00(0.00E-00)/=	0.00E-00(0.00E-00)/=	0.00E-00(0.00E-00)/=	0.00E-00(0.00E-00)
f_{12}	2.95E-02(1.66E-02)/+	4.96E-07(3.33E-07)/+	4.96E-02(7.30E-02)/+	2.85E-04(1.78E-03)/+	1.85E-09(5.88E-09)/=	2.53E-09(3.26E-09)
f_{13}	1.75E-00(2.20E-01)/+	2.98E-05(8.17E-05)/+	2.95E-05(1.74E-01)/+	3.80E-01(2.76E-01)/+	1.43E-01(1.72E-01)/+	7.13E-08(7.36E-08)
Wilcoxon Test Summary +/ = / -	12/1/0	10/3/0	12/1/0	9/4/0	8/5/0	Wilcoxon

* The data format: Average / Standard Deviation / Wilcoxon Test Sign (+/-/=)

Among the thirteen test functions, our IESO algorithm significantly outperforms other algorithms. This advantage is attributed to the non-monotonic decrease in the temperature factor, which fosters interaction among the foraging, combat, and mating modes. Specifically, the foraging mode utilizes the current global best position to generate high-quality solutions throughout the mid-to-high iterations, accelerating convergence. The only exception is f_{12} , where ESO slightly outperforms our IESO. Its numerous local minima characterize this function, posing a considerable challenge for global optimization algorithms. The analysis suggests that introducing the mirroring mechanism has somewhat expanded the exploration horizon of the SO. However, as stated in Wang et al., [20], due to limited executions and dynamically narrowing boundaries over iterations, the development mode, which relies on information exchange between the male and female snake populations, might still result in the optimizer becoming trapped in local optima. Wang et al. suggested incorporating Lévy flights into the ESO for mutating optimal individuals. As a distinctive heavy-tailed distribution, Lévy flights, with their long and short step lengths, help the algorithm escape local optima. In our IESO, we employed a dynamic boundary to the opposite learning mechanism to enhance performance further, which has better results escaping the local optima than ESO. However, in some cases, it may not be fully utilized.

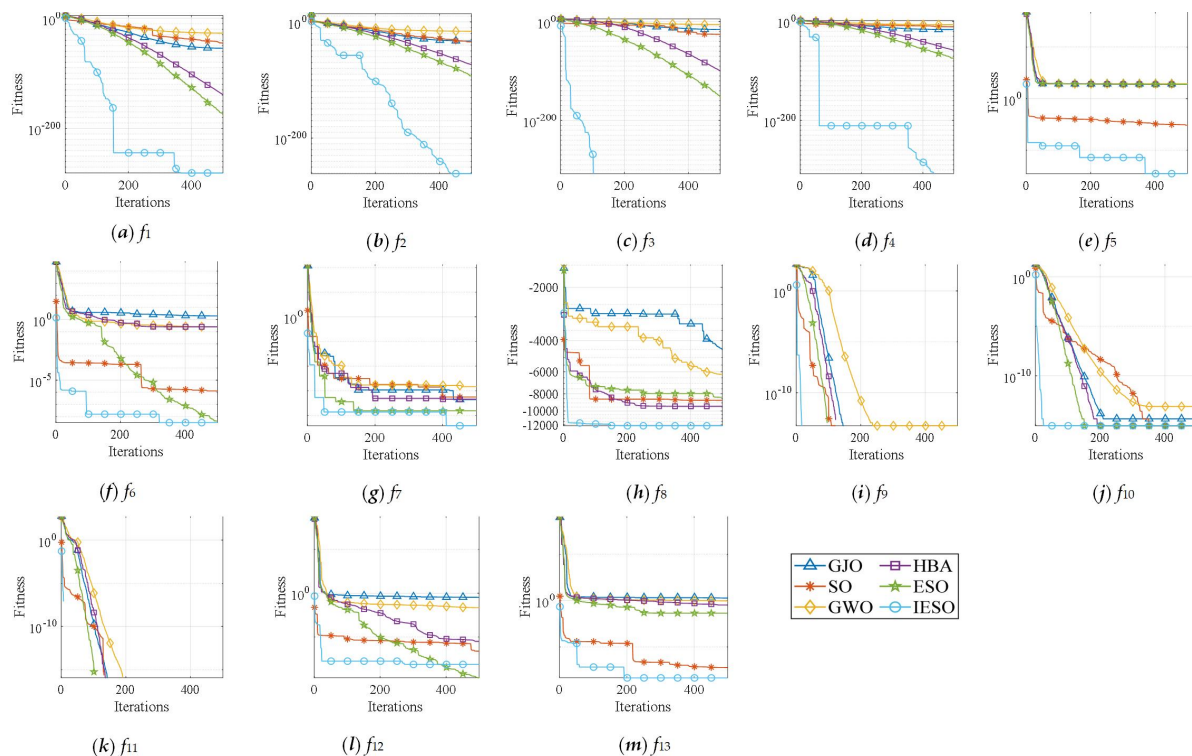


Figure 5. Iteration fitness curves for different test functions.

A simulated digital elevation model of rugged mountainous terrain was obtained, with the map dimensions measuring 100 km by 150 km by 3 km. The maximum elevation difference in this area exceeds 2 km, featuring a terrain with significant undulations and numerous ravines. Randomly generated "no-fly zones" within this terrain are added and are illustrated in Figure 6 (purple areas). The launch point is set at coordinates (10, 90, 1.115), and the destination is at (130, 10, 1.373), with all units in kilometres (km). The fundamental challenge of path planning lies in identifying suitable waypoints during the flight to achieve an optimal flight route. To this end, this study sets 15 flight waypoints within the three-dimensional map and then employs interpolation to obtain detailed flight paths between adjacent waypoints. The parameters to be optimized are the spatial coordinates of these 15 waypoints.

The six algorithms—GJO, HBA, GWO, SO, ESO, and our IESO—were selected to plan the optimized routing on this simulated terrain. As shown in Figure 7, IESO exhibits a significantly faster

convergence time than the other algorithms, at least 20% faster than GJO and over 35% faster than ESO. Additionally, Figures 8 and 9 demonstrate that the route proposed by IESO is the shortest and successfully avoids all no-fly zones, as seen from both 2D and 3D perspectives. It is at least 10% shorter than the competitors. Therefore, we can conclusively state that IESO significantly improves flight planning in complex environments, such as rugged mountain terrains with substantial undulations and numerous ravines, while effectively avoiding arbitrary no-fly zones.

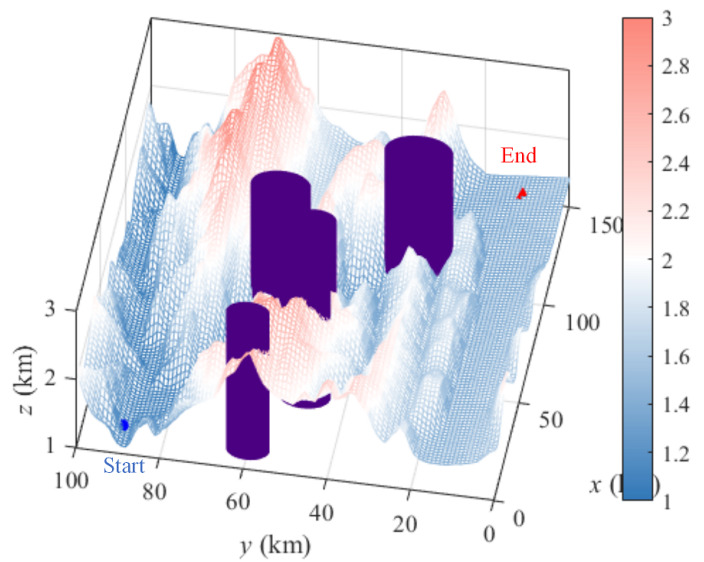


Figure 6. The simulated rugged mountain terrain with "no-fly" zones.

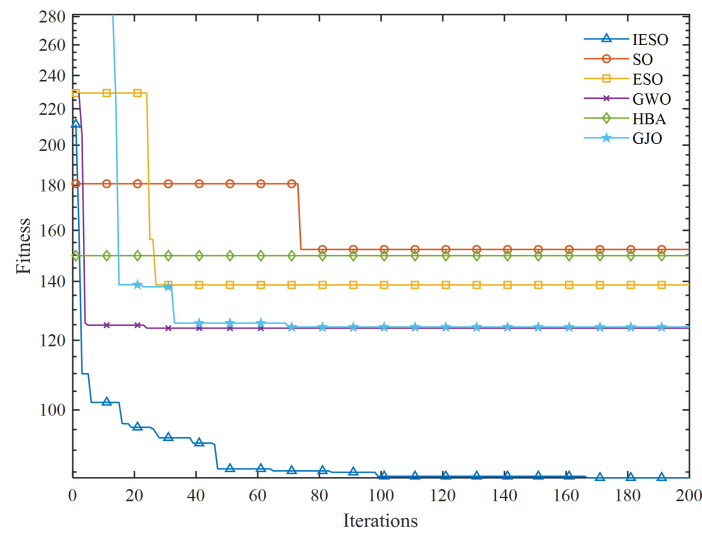


Figure 7. Iteration fitness curves for the six algorithms

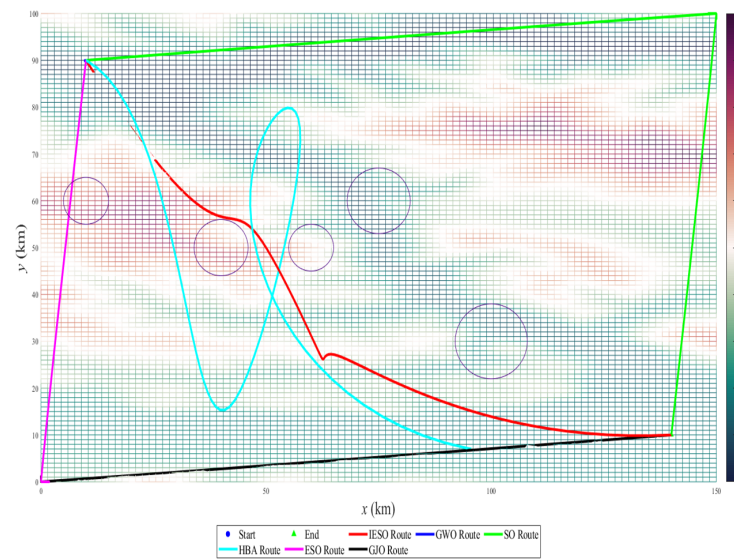


Figure 8. The planned routes from the six algorithms

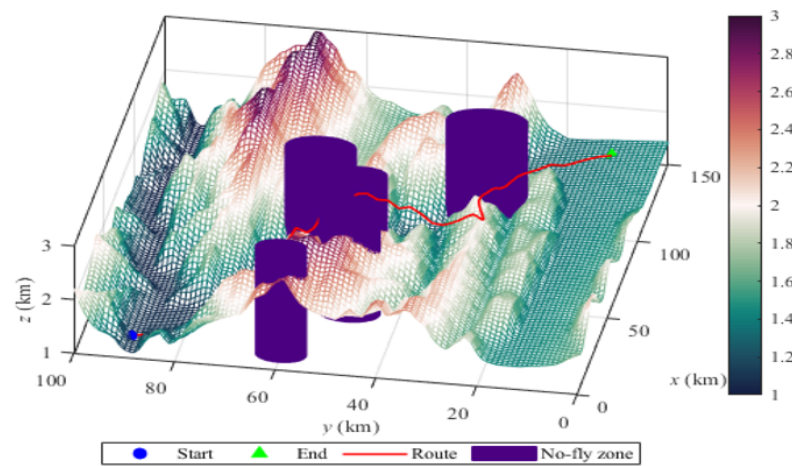


Figure 9. The IESO planned route on the simulated terrain

4. Conclusions

In this study, we propose an Improved Enhanced Snake Optimizer (IESO) to optimize 3D UAV path planning in complex conditions such as rugged mountainous terrains. The IESO refines the Enhanced Snake Optimizer (ESO) by incorporating Chebyshev chaotic mapping, a non-monotonically decreasing temperature factor, and dynamic boundary-based opposition learning (DBOL).

We validated the results in simulated experiments using thirteen different test functions under three criteria against five other innovative optimization algorithms. The IESO consistently outperformed these algorithms, demonstrating superior convergence accuracy and speed, which is over 30% shorter in distance and 10% faster in computing speed.

Author Contributions: Conceptualization, W.L. and K.Z.; methodology, W.L.; software, W.L. and Q.X.; validation, W.L., and Q.X.; formal analysis, W.L. and K.Z.; investigation, W.L.; resources, W.L. and Q.X.; data curation, X.X.; writing—original draft preparation, W.L. and X.C.; writing—review and editing, K.Z.; visualization, W.L. and K.Z.; supervision, Q.X.; project administration, Q.X.; funding acquisition, Q.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the Natural Science Foundation of Hunan Province under Grant NO. 2023JJ50328, and partly by the Scientific Research Project of HUAS under Grant No. 22ZD03.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available on request from the corresponding author, [Wuke Li], upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DBOL	Dynamic Boundary-based Opposition Learning
DOL	Dynamic Opposition Learning
ESO	Enhanced Snake Optimizer
GJO	Golden Jackal Optimization
GWO	Grey Wolf Optimizer
HBA	Honey Badger Algorithm
MA	Meta-heuristic Algorithm
SO	Snake Optimizer
IESO	Improved Enhanced Snake Optimizer
OBL	Opposition-Based Learning
UAV	Unmanned Aerial Vehicle

References

1. Jones, M.; Djahel, S.; Welsh, K. Path-Planning for Unmanned Aerial Vehicles with Environment Complexity Considerations: A Survey. *ACM Computing Surveys* **2023**, *55*, 1–39 <https://doi.org/10.1145/3570723>.
2. Aggarwal, S.; Kumar, N. Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges. *Computer Communications* **2020**, *149*, 270–299. <https://doi.org/10.1016/j.comcom.2019.10.014>.
3. Yahia, H. S.; Mohammed, A. S. Path Planning Optimization in Unmanned Aerial Vehicles Using Meta-heuristic Algorithms: A systematic review. *Environmental Monitoring and Assessment* **2023**, *10*, 10 <https://doi.org/10.1007/s10661-022-10590-y>.
4. Li, D.; Yin, W.; Wong, W.E.; Jian, M.; Chau, M. Quality-oriented Hybrid Path Planning Based on A* and Q-learning for Unmanned Aerial Vehicle. *IEEE Access* **2021**, *10*, 7664–7674 [10.1109/ACCESS.2021.3139534](https://doi.org/10.1109/ACCESS.2021.3139534).
5. Pan, Z.; Zhang, C.; Xia, Y.; Xiong H.; Shao, X. An Improved Artificial Potential Field Method for Path Planning and Formation Control of the Multi-UAV Systems. *IEEE Transactions on Circuits and Systems II: Express Briefs* **2022**, *69*, 1129–1133 [10.1109/TCSII.2021.3112787](https://doi.org/10.1109/TCSII.2021.3112787).
6. Wang, J.; Li, Y.; Li, R.; Chen, H.; Chu, K. Trajectory Planning for UAV Navigation in Dynamic Environments with Matrix Alignment Dijkstra. *Application of soft computing* **2022**, *26*, 12599–12610 <https://doi.org/10.1007/s00500-022-07224-3>.
7. Prasad, N.L.; Ramkumar, B. 3-D Deployment and Trajectory Planning for Relay Based UAV Assisted Cooperative Communication for Emergency Scenarios Using Dijkstra's Algorithm. *IEEE Transactions on Vehicular Technology* **2022**, *72*, 5049–5063 [10.1109/TVT.2022.3224304](https://doi.org/10.1109/TVT.2022.3224304).
8. Flores-Caballero, G.; Rodríguez-Molina, A.; Aldape-Pérez, M.; Villarreal-Cervantes, M. G. Optimized Path-Planning in Continuous Spaces for Unmanned Aerial Vehicles Using Meta-Heuristics. *IEEE Access* **2020**, *8*, 176774–176788 [10.1109/ACCESS.2020.3026666](https://doi.org/10.1109/ACCESS.2020.3026666).
9. Shao, S.; He, C.; Zhao, Y.; Wu, X. Efficient Trajectory Planning for UAVs Using Hierarchical Optimization. *IEEE Access* **2021**, *9*, 60668–60681 [10.1109/ACCESS.2021.3073420](https://doi.org/10.1109/ACCESS.2021.3073420).
10. Chen, J. C.; Ling, F. Y.; Zhang, Y.; You, T.; Liu, Y. F.; Du, X. Y. Coverage Path Planning of Heterogeneous Unmanned Aerial Vehicles Based on Ant Colony System. *Swarm and Evolutionary Computation* **2022**, *69*, 101005 <https://doi.org/10.1016/j.swevo.2021.101005>.
11. Yao, L.; Yuan, P.; Tsai, C.Y.; Zhang, T.; Lu, Y.; Ding, S. ESO: An Enhanced Snake Optimizer for Real-world Engineering Problems. *Expert Systems with Applications* **2023**, *230*, 120594 <https://doi.org/10.1016/j.eswa.2023.120594>.

12. Hashim, F.A., Hussien, A.G. Snake Optimizer: A Novel Meta-heuristic Optimization Algorithm. *Expert Systems with Applications* **2022**, 242, 108320 <https://doi.org/10.1016/j.knsys.2022.108320>.
13. Wolpert, D.H., Macready, W.G. No Free Lunch Theorems for Optimization. *IEEE Transactions on Evolutionary Computation* **1997**, 1, 67–82 [10.1109/4235.585893](https://doi.org/10.1109/4235.585893).
14. Chopra, N., Ansari, M.M. Golden Jackal Optimization: A Novel Nature-inspired Optimizer for Engineering Applications. *Expert Systems with Applications* **2022**, 198, 198 <https://doi.org/10.1016/j.eswa.2022.116924>.
15. Hashim, F.A., Houssein, E.H., Hussain, K., Mabrouk, M.S., Al-Atabany, W. Honey Badger Algorithm: New Metaheuristic Algorithm for Solving Optimization Problems. *Mathematics and Computers in Simulation* **2022**, 192, 84–110 <https://doi.org/10.1016/j.matcom.2021.08.013>.
16. Nadimi-Shahraki M., Taghian S., Mirjalili S. An Improved Grey Wolf Optimizer for Solving Engineering Problems. *Expert Systems with Applications* **2021**, 166, 113917 <https://doi.org/10.1016/j.eswa.2020.113917>.
17. Xu, Y., Li, X., Yang, J., Zhang, D. Integrate the original face image and its mirror image for face recognition. *Neurocomputing* **2014**, 131, 191–199 <https://doi.org/10.1016/j.neucom.2013.10.025>.
18. Tizhoosh, H. R. Opposition-based learning: a new scheme for machine intelligence. *International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06)* **2005**, 131, 695–701 <https://doi.org/10.1109/cimca.2005.1631345>.
19. Wu, T.Y., Li, H., Chu, S.C. CPPE: An Improved Phasmatodea Population Evolution Algorithm with Chaotic Maps. *Mathematics* **2023**, 11, 1977 <https://doi.org/10.3390/math11091977>.
20. 16. Wang, C., Jiao, S., Li, Y. and Zhang, Q. Capacity Optimization of a Hybrid Energy Storage System Considering Wind-Solar Reliability Evaluation Based on A Novel Multi-strategy Snake Optimization Algorithm *Expert Systems with Applications* **2023**, 231, 120602 <https://doi.org/10.1016/j.eswa.2023.120602>.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.