G. Grigoras and P. Lorenz (Eds.)

doi:10.3233/FAIA230917

An Improved Coati Optimization Algorithm Based on Multi-Strategy

Xingwang RU[1]
School of Information Engineering, Anhui business and technology college
Hefei, China

Abstract. To solve the problems of the original Coati Optimization Algorithm with low exploration ability, insufficient exploitation ability and easy to fall into local optimum, an improved coati optimization algorithm based multi-strategy is proposed. Firstly, the initial solution traversal is improved through Circle chaotic mapping to lay the foundation for global search; secondly, the local optimum is jumped out through Lévy flight to improve the global search ability of the algorithm; finally, the performance of the proposed algorithm is evaluated using numerical analysis and convergence analysis in comparison with four algorithms such as PSO and WOA. The experimental results show that the algorithm in this paper has better search accuracy and convergence speed and has better performance in solving high-dimensional problems.

Keywords. Coati Optimization Algorithm, Circle chaotic mapping, Lévy flight, global optimisation search, metaheuristic algorithm

1. Introduction

In recent years, more and more swarm intelligence optimization algorithms[1] have been proposed to solve combinatorial optimization problems, including particle swarm optimization (PSO) [2], grey wolf optimization (GWO) [3], whale optimization algorithm (WOA) [4], etc. As a popular optimization algorithm in recent years, the metaheuristic algorithm [5] is widely used because of its good stochasticity and strong global search ability and can solve many engineering problems of finding optimal solutions. However, due to the high complexity and difficulty of most engineering problems, it is difficult for a separate optimization algorithm to solve all practical optimization problems. The Coati Optimization Algorithm (COA) is a meta-heuristic algorithm based on swarm intelligence proposed by Dehghani et al [6] in 2023, which emulates the two social behaviours of coati in nature. The algorithm has better search capability compared to other swarm intelligence algorithms, but the COA algorithm still has some disadvantages, such as the problem that the algorithm falls into local optima and the convergence performance decreases as the search increases. Therefore, researchers have made some improvements to the COA algorithm. For example, Sammen et al. [7] proposed a binary COA optimisation algorithm combined with an extreme learning machine model to predict reservoir water level. Rama et al. [8] proposed a new approach to predict dengue disease in patients using COATI-optimized support vector machines. Due to the short time for the COA algorithm to be proposed,

[1] Xingwang RU, Anhui business and technology college; e-mail: ruxw@ahbvc.edu.cn

so far, there is not much new progress in the improvement and application of the COA algorithm, and the original algorithm is only quoted for comparison with the newly proposed algorithm, for example, the Kepler optimization algorithm newly proposed by Abdel-Basset et al. [9] in 2023 was compared with the performance of the COA algorithm, and the results showed that both algorithms have good competitiveness. A leader–advocate–believer-based optimization algorithm proposed by Reddy et al. [10] was also compared with the COA algorithm, and the results showed that the new algorithm performed better in optimization results.

Due to the simple optimisation algorithm on the actual problem, there are still disadvantages such as poor optimisation search effect. These optimization algorithms need to be further optimized and improved in order to better apply them to practical application scenarios. To solve the problems of the original COA algorithm such as easy to fall into local optimum and low convergence accuracy. In this paper, we propose an improved coati optimization algorithm (ICOA) based on multi-strategy, which firstly enhances the traversal of the initial solution through Circle chaotic mapping to lay the foundation for global search; and secondly enhances the global search capability of the algorithm by jumping out of the local optimum through Lévy flight.

2. Original Coati Optimization Algorithm

The COA is a new meta-heuristic based algorithm proposed by Dehghani et al. in 2023, which mimics two social behaviors of coatis in nature: (i) the behavior when attacking and hunting iguanas (exploration phase), and (ii) the behavior when fleeing from predators (the exploitation phase).

2.1. Initializing the Population

In the COA algorithm, coatis are considered as members of the population and are randomly initialized using Equation (1) in the initialization phase of the algorithm.

$$X_i: x_{i,j} = lb_j + r \cdot \left(ub_j - lb_j\right) \tag{1}$$

where X_i represents the location of coati in generation i, $x_{i,j}$ is the value of the decision variable of the population in generation j, r is the value of the random variable taking values between [0,1], and lb_j and ub_j are the lower and upper boundary values in generation j, respectively.

2.2. Exploration Phase

In the COA algorithm, the coati at the optimal position represents the position of the iguana, where half of the coatis are in the trees to catch the prey and half of them are on the ground. Therefore, equation (2) was used to simulate the hunting behavior.

$$X_{i,j}^{P1}: x_{i,j}^{P1} = x_{i,j} + r \cdot \left(Iguana_j - I \cdot x_{i,j}\right), for\ i = 1,2,\dots,\left\lfloor \frac{N}{2} \right\rfloor\ and\ j = 1,2,\dots,m \tag{2}$$

The iguana will fall from the tree to a random location on the ground. The coati on the ground will move according to the iguana's position, using equation (3)(4) to simulate.

$$Iguana^G: Iguana_j^G = lb_j + r \cdot (ub_j - lb_j), j = 1,2,\dots,m \tag{3}$$

$$X_{i,j}^{P1}: x_{i,j}^{P1} = \begin{cases} x_{i,j} + r \cdot (Iguana_j^G - I \cdot x_{i,j}), F_{Iguana^G} < F_i \\ x_{i,j} + r \cdot (x_{i,j} - Iguana_j^G), else \end{cases} \tag{4}$$

2.3. Exploitation Phase

When a coati is attacked, it will flee its current location. We generate a random location near the current location of each coati to represent the fleeing location of the coati and use equation (5)(6) to simulate this behavior.

$$lb_j^{local} = \frac{lb_j}{t}, ub_j^{local} = \frac{ub_j}{t}, where\ t = 1,2,\dots,T \tag{5}$$

$$X_{i,j}^{P2}: x_{i,j}^{P2} = x_{i,j} + (1 - 2r) \cdot \left(lb_j^{local} + r \cdot (ub_j^{local} - lb_j^{local})\right), i = 1,2,\dots,N \tag{6}$$

If the new position has a higher fitness value than the original position, then we receive the current new position, and this condition is simulated using equation (7).

$$X_i = \begin{cases} X_i^{P2}, & F_i^{P2} < F_i \\ X_i & else \end{cases} \tag{7}$$

3. Multi-strategy Improvements

3.1. Circle Chaos Mapping

The location of the initial solution plays a crucial role in the algorithm, which directly affects the optimisation effect and convergence of the algorithm. The initialisation of the algorithm uses pseudo-random numbers to generate the individual positions. Although this allows the whole population to be distributed in the solution space, it will inevitably cause uneven distribution of the population and cause some regions to be too dense. Circle chaos mapping is used to initialise the population, as shown in equation (8)

$$x_{i+1} = mod\left(x_i + 0.2 - \left(\frac{0.5}{2\pi}\right) sin(2\pi x_i), 1\right) \tag{8}$$

Compared with the randomly distributed population of the original algorithm, using Circle mapping to initialize the population can make the initial location of the population more uniformly distributed and increase the diversity of population locations. This can improve to a certain extent the defect that the algorithm is easy to fall into local extremes, thus improving the ability of the algorithm to jump out of local optimum. Figure 1 shows the 2D population distribution, the figure on the left is the 2D initialized population after

Circle Chaos mapping, the figure on the right is the original algorithm randomized initialized population.

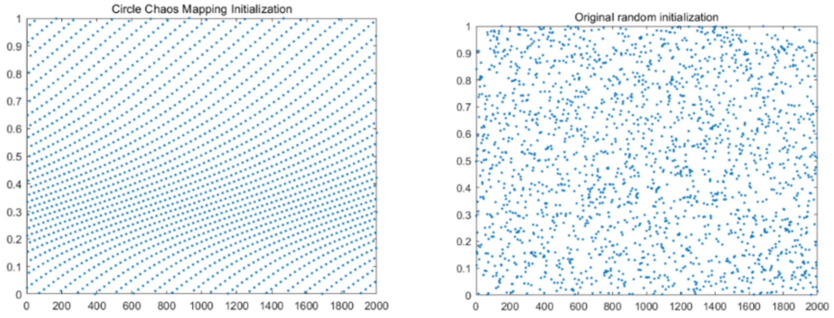

Figure 1. 2D initial population distribution map.

3.2. Lévy Flight

In the last decade, Lévy flights have been widely used in intelligent optimisation algorithms to enhance the search capability, avoid local optima, speed up convergence and increase population diversity. The Lévy flight pattern is characterised by many small and occasionally large step sizes, which allows the algorithm to efficiently search for distant regions of space in global optimisation problems and helps the algorithm to avoid getting trapped in local optima. This paper therefore uses the Lévy flight strategy for the first and second stages of the algorithm, with the position update Equation (9). Figure 2 shows the step and trajectory diagram of Lévy flight.

$$\begin{cases} X_{i,j}^{P1}: x_{i,j}^{P1} = x_{i,j} + r \cdot \left(Iguana_j - I \cdot x_{i,j}\right) \otimes Levy(D) \\ X_{i,j}^{P2}: x_{i,j}^{P2} = x_{i,j} + (1-2r) \cdot \left(lb_j^{local} + r \cdot \left(ub_j^{local} - lb_j^{local}\right)\right) \otimes Levy(D) \end{cases} \tag{9}$$

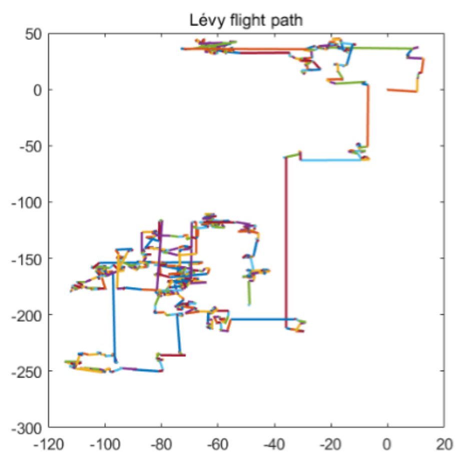

Figure 2. Lévy flight path.

4. Simulation Experiments and Analysis

To test the performance of the improved ICOA algorithm, comparisons were made with four algorithms, including COA, PSO, GWO and WOA. The population size of each algorithm was set to 200 and the maximum number of iterations was 500, and each was tested on 8 different types of benchmark test functions. The runs were repeated 50 times independently to obtain the mean value of the optimal solution with standard deviation. Table 1 describes the function names, types, search ranges, and global polarization points of the tested functions. Table 2 describes the results of each algorithm tested. The hardware used in the experimental environment is Windows 10 operating system, processor Intel(R) Core(TM) i9-12900H CPU @ 3.50GHz and 64.0 GB RAM. The programming software was MATLAB R2022b.

Table 1. The description of benchmark function.

Function Type	Func	Function Name	Range	Theoretical Optimization Value
Unimodal test functions	F1	Sphere	[-100,100]	0
	F2	Schwefe 2.22	[-10,10]	0
	F3	Schwefe 1.2	[-100,100]	0
Multimodal test functions	F8	Schwefe 2.26	[−500, 500]	−418.9829 × Dimensions
	F9	Rastrigin	[−5.12, 5.12]	0
	F10	Ackley's	[-32， 32]	0

Table 2. Benchmark function of experimental results.

Func	Type	ICOA	COA	PSO	WOA	GWO
F1	Mean	9.83E-115	8.83E-32	2.37E+03	4.47E-06	1.13E-07
	Std	6.72E-114	5.45E-31	6.09E+02	3.93E-06	6.52E-08
	Time/s	14.166992	18.569670	30.902941	27.308306	19.452467
F2	Mean	2.15E-43	1.48E+02	4.68E+01	4.53E-05	2.75E-05
	Std	1.51E-42	2.29E+02	8.99E+00	3.33E-05	7.26E-06
	Time/s	44.479212	19.300527	25.258943	30.947919	20.782940
F3	Mean	3.75E-65	2.00E-12	1.78E+05	1.49E+05	2.06E+04
	Std	2.63E-64	6.07E-12	3.84E+04	3.75E+04	1.15E+04
	Time/s	93.277829	66.472205	449.635650	114.946045	69.122119
F8	Mean	-2.22E+04	-3.46E+04	-3.65E+04	-1.92E+04	-2.83E+04
	Std	2.67E+03	3.25E+03	4.43E+03	1.35E+03	3.17E+03
	Time/s	52.520541	23.864145	54.726333	21.509454	24.635168
F9	Mean	0.00E+00	4.74E-11	6.18E+02	2.18E+03	2.21E+01
	Std	0.00E+00	3.35E-10	7.55E+01	2.95E+02	1.26E+01
	Time/s	25.981522	29.342761	52.590775	22.551872	23.025226
F10	Mean	4.44E-16	1.15E-15	6.93E+00	3.57E-04	2.23E-05
	Std	0.00E+00	1.44E-15	5.64E-01	5.21E-04	5.63E-06
	Time/s	20.226922	22.760933	38.834268	20.906987	22.138801

From the analysis of convergence accuracy, a smaller mean indicates better average performance of the algorithm, it can be seen from Table 2 that the mean values obtained by the ICOA algorithm are all better than the other algorithms, and for functions for which the global optimal solution cannot be found, the ICOA algorithm obtains the solution with the highest accuracy.From the convergence speed analysis, a smaller variance indicates better stability of the algorithm. It can be seen from Table 2 that the number of iterations of the ICOA algorithm is the lowest for functions that can find the extreme value point, such as F9. And the running time of the functions are also all less. It can also be seen from Figure 3 that the convergence speed of the ICOA algorithm is better than the other algorithms. Combined with the above analysis, the algorithm in this paper has better performance in terms of better finding accuracy and convergence speed for solving high-dimensional problems.

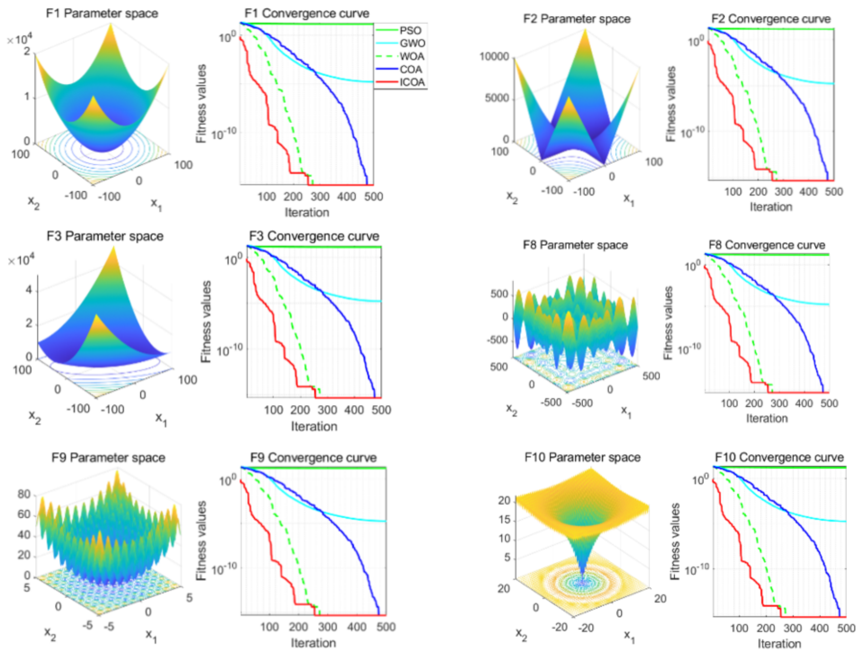

Figure 3. Benchmark function of convergence curve.

5. Conclusion

In this paper, to solve the problems of the original COA algorithm, an improved coati optimization algorithm based on multi-strategy is proposed. Firstly, the traversal of the initial solution is enhanced through Circle chaotic mapping to lay the foundation for global search. Secondly, the local optimum is jumped out through Lévy flight to enhance the global search capability of the algorithm. The results show that the algorithm in this paper has the advantages of fast speed, high accuracy, and robustness in solving high-dimensional complex function problems. The next research direction is to apply the ICOA algorithm to practical engineering optimization problems to test the performance of the algorithm.

Acknowledgement

This work has been supported by the Anhui Provincial Quality Project (2021xjjyZD02, 2021cjrh006, 2021jxtd032, 2022xnfzjd003, 2022jyxm157, 2022zygzsj007), Educational Teaching Research Planning Project of Anhui Professional and Adult Education Society (Azcj2022005), Major Project of Natural Science Research in Higher Education of Anhui Province (KJ2019A1167, 2022AH052794).

References

[1] Wang, Xiaonan, et al. "On the mathematical models and applications of swarm intelligent optimization algorithms." Archives of Computational Methods in Engineering 29.6 (2022): 3815-3842.
[2] Li, Taiyong, et al. "Pyramid particle swarm optimization with novel strategies of competition and cooperation." Applied Soft Computing 121 (2022): 108731.
[3] Meidani, Kazem, et al. "Adaptive grey wolf optimizer." Neural Computing and Applications 34.10 (2022): 7711-7731.
[4] Yang, Wenbiao, et al. "A multi-strategy Whale optimization algorithm and its application." Engineering Applications of Artificial Intelligence 108 (2022): 104558.
[5] Pan, Jeng-Shyang, et al. "A survey on binary metaheuristic algorithms and their engineering applications." Artificial Intelligence Review (2022): 1-67.
[6] Dehghani, Mohammad, et al. "Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems." *Knowledge-Based Systems* 259 (2023): 110011.
[7] Sammen, Saad Sh, et al. "Binary Coati Optimization Algorithm-Multi-Kernel Least Square Support Vector Machine-Extreme Learning Machine Model (BCOA-MKLSSVM-ELM): A New Hybrid Machine Learning Model for Predicting Reservoir Water Level." Water 15.8 (2023): 1593.
[8] Rama Krishna, K., K. G. Mohan, et al. " A Novel Approach to Predict Dengue Diseases in Patients Using Coati Optimization-Based Support Vector Machine." 2023, 12(Regular Issue 4), 3823-3830.
[9] Abdel-Basset, Mohamed, et al. "Kepler optimization algorithm: A new metaheuristic algorithm inspired by Kepler's laws of planetary motion." Knowledge-Based Systems 268 (2023): 110454.
[10] Reddy, Ruturaj, et al. "LAB: a leader–advocate–believer-based optimization algorithm." Soft Computing (2023): 1-35.