

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/347276267>

The improved mayfly optimization algorithm

Article in *Journal of Physics Conference Series* · November 2020

DOI: 10.1088/1742-6596/1684/1/012077

CITATIONS

7

READS

697

4 authors, including:



Zheng-Ming Gao

Jingchu university of technology

59 PUBLICATIONS 235 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



intelligent information [View project](#)

PAPER • OPEN ACCESS

The improved mayfly optimization algorithm

To cite this article: Zheng-Ming Gao *et al* 2020 *J. Phys.: Conf. Ser.* **1684** 012077

View the [article online](#) for updates and enhancements.



IOP | ebooks™

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

The improved mayfly optimization algorithm

Zheng-Ming GAO^{1,a}, Juan ZHAO^{2,b}, Su-Ruo LI^{3,c}, Yu-Rong HU^{4,d}

¹School of computer engineering, Jingchu University of technology Jingmen, China, gaozming@jcut.edu.cn

²School of electronics and information engineering, Jingchu University of technology Jingmen, China, ajuan323@jcut.edu.cn

³School of computer engineering, Jingchu University of technology Jingmen, China, 2410500736@qq.com

⁴Department of science and technology, Jingchu University of technology Jingmen, China, yrhu@jcut.edu.cn

Abstract— The mayfly optimization (MO) algorithm was proposed with a better hybridization of the particle swarm optimization (PSO) and the differential evolution (DE) algorithms. The velocity would be relevant to the Cartesian distance among the relevant individuals. In this paper, a reasonable revision for the velocity updating equations was proposed based on the idea of moving towards each other as capable as they can. Simulation results proved that the improved MO algorithm would perform better than the original one.

1. INTRODUCTION

Lots of optimization algorithms have been proposed to solve the problems we met during our exploring, exploiting, and conquering nature. However, most of the algorithms still lack the capability to solve all of the problems, and consequently, efforts still remained for scientists and engineers to find more capable algorithms.

Regarding of the structure of the algorithms, only the averages were involved in the past, such as the ant colony optimization (ACO) algorithm^[1], the individuals in swarms would update their positions according to their current positions and the average of swarms, similar operation was taken for the bat algorithm^[2], individuals in the bat algorithm would fly with more complicated forms. In the particle swarm optimization (PSO) algorithm^[3], however, the best candidate together with the best historical trajectories were all introduced in updating equations. Considering the individuals' intelligence and the social hierarchy of swarms, the top three best candidates were all included in the update equation in the grey wolf optimization (GWO) algorithm^[4], furthermore, top four best candidates including their average were involved in the updating equation in the equilibrium optimization (EO) algorithm,^[5] although efforts have been made and results showed that the best candidate only would perform better in updating^[6].

Recently, the mayfly optimization (MO) algorithm was proposed^[7]. In this algorithm, the individuals in swarms would be specifically identified as male and female mayflies. And both of them perform different updating behavior. However, in the original version of the MO algorithm, if the current positions were far away from the best candidate or the historical best trajectories, the individuals would run towards the best position with a lower speed. On the contrary, if the current positions is near the global best candidate or the best historical trajectories, then the individuals would perform with faster speeds. Such



operations would directly slow down the convergent rate. Therefore, in this paper, we would reconstruct the updating equations for individuals and improve the capability of MO algorithms.

The following parts of this paper would be arranged as follows: in section II, we would briefly describe the MO algorithm, while in section III, the proposed improvement would be raised and simulation experiments would be carried out in section IV. Discussions would be made and conclusions would be drawn in section V.

2. THE MO ALGORITHM

The mayflies in swarms for the MO algorithm would be separated into male and female individuals. And the male mayflies would always strong and consequently, they would perform better in optimization.

Similar to the individuals in swarms of the PSO algorithm, the individuals in the MO algorithm would update the positions according to their current positions $p_i(t)$ and velocity $v_i(t)$ at the current iteration:

$$p_i(t+1) = p_i(t) + v_i(t+1) \quad (1)$$

All of the male mayflies and female mayflies would update their positions with equation (1). However, their velocity would be updated in different ways.

2.1. Movements of male mayflies

Male mayflies in swarms would carry on exploration or exploitation procedure during iterations. The velocity would be updated according to their current fitness values $f(x_i)$ and the historical best fitness values in trajectories $f(x_{h_i})$.

If $f(x_i) > f(x_{h_i})$, then, the male mayflies would update their velocities according to their current velocities, together with the distance between them and the global best position, the historical best trajectories:

$$v_i(t+1) = g \cdot v_i(t) + a_1 e^{-\beta r_p^2} [x_{h_i} - x_i(t)] + a_2 e^{-\beta r_g^2} [x_g - x_i(t)] \quad (2)$$

Where, g is a variable declined linearly from the maximum value to a smaller one. a_1 , a_2 , and β are two constants to balance the values. r_p and r_g are two variables used to tell the Cartesian distance between the individuals and its historical best position, the global best position in swarms. The Cartesian distance would be the second norm for the distance array:

$$\|x_i - x_j\| = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2} \quad (3)$$

On the other hand, if $f(x_i) < f(x_{h_i})$, the male mayflies would update their velocities from the current one with a random dance coefficient d :

$$v_i(t+1) = g \cdot v_i(t) + d \cdot r_1 \quad (4)$$

Where, r_1 is the random number in uniform distribution and selected from the domain $[-1, 1]$.

2.2. Movements of female mayflies

The female mayflies would update their velocities with a different style. Biologically speaking, the female mayflies with wings only survive in one day to seven days at most, so the female mayflies would be in rush to find the male mayflies to mate and reproduce themselves. Therefore, they would update their velocities based on the male mayflies they want to mate.

In the MO algorithm, the top best female and male mayflies would be treated as the first mate, and the second best female, male mayflies would be treated as the second mates, and so on. So for the i -th female mayfly, if $f(y_i) < f(x_i)$:

$$v_i(t+1) = g \cdot v_i(t) + a_3 e^{-\beta r_m^2} [x_i(t) - y_i(t)] \quad (5)$$

Where, a_3 is another constant and also used to balance the velocities. r_m represents the Cartesian distance between them.

On the contrary, if $(y_i) < f(x_i)$, the female mayflies would update their velocities from the current one with another random dance fl :

$$v_i(t) = g \cdot v_i(t) + fl \cdot r_2 \quad (6)$$

Where, r_2 is also a random number in uniform distribution in domain $[-1, 1]$.

2.3. Mating of mayflies

All of the top half female and male mayflies would be mated and given pair of children for every one of them. Their offspring would be randomly evolved from their parents:

$$offspring1 = L * male + (1 - L) * female \quad (7)$$

$$offspring2 = L * female + (1 - L) * male \quad (8)$$

Where, L are random numbers in Gauss distribution.

3. THE IMPROVED MO ALGORITHM

Equations (4) and (6) would show us that in some circumstances, the individuals in swarms would update their velocities with randomness. However, the velocities would be updated with complicated methods in the other situations. According to equations (2) and (5), the velocities of individuals were updated from the weighted current velocities with some other weighted distance between them and the historical best trajectories, global best candidate, or their mates. In details, we would find that either parts of the weighted distance would appear in the following way:

$$v_p = a_i e^{-\beta r_j^2} (p_j - p_i) \quad (9)$$

Apparently, r_j would be larger if the distance between the j -th individual and the i -th individual increased (the i -th individual might be referred to the global best position, the historical best position for the j -th individual, or its mate). However, because of the declination of the negative exponential function, the weights for the distance would be smaller instead. These means that if the distance between p_j and p_i is increased, the weights would be decreased, the composited velocity v_p would be then decreased. On the other hand, if the distance between p_j and p_i is decreased, the weights would be increased instead.

Consequently, when p_j is far away from p_i , it would update its velocity in a lower amplitude, while when p_j is near p_i , it would update its velocity in a larger amplitude. Which also means that when they are far away from each other, they would approach to each other in a lower rate, on the contrary, if they meet each other face to face, they would dance away with a larger rate. These situations would be totally un-acceptable.

Literally speaking, when the individuals are far away from each other, they should update their velocities with larger rates and when they are nearby, the velocities should be updated with smaller rates. Therefore, equation (9) must be revised to satisfy such situation with an example as follows:

$$v_p = a_i e^{-\frac{\beta}{r_j}} (p_j - p_i) \quad (10)$$

4. SIMULATION EXPERIMENTS

In this section, we would introduce some classical benchmark functions to verify whether the improvement with equation (10) was reasonably true or not.

4.1. Simulation experiments on unimodal benchmark functions

Sphere function:

$$f(x) = \sum_{i=1}^d x_i^2 \quad (11)$$

Sphere function is unimodal and simple, most of the optimization algorithm would easily find the best solutions. Similar jobs would be done for the MO algorithm, as shown in Figure 1.

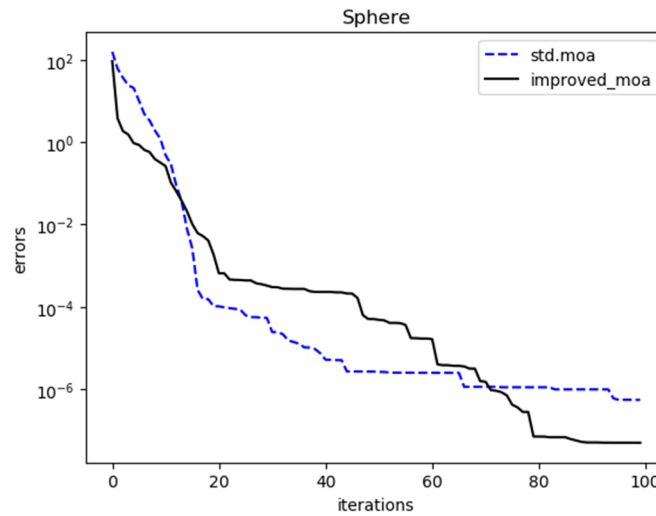


Figure 1 Simulation experiments results versus iterations

Note that the overall results were the averages for 100 simulation experiments. The Monte Carlo method was introduced here to reduce the influence of randomness.

4.2. Simulation experiments on multimodal benchmark functions

Styblinski-Tang function:

$$f(x) = \frac{1}{2} \sum_{i=1}^d (x_i^4 - 16x_i^2 + 5x_i) + 78.332 \tag{12}$$

Styblinski-Tang function is multimodal with basins in its profiles, as shown in Figure 2, which is the three-dimensional profile of Styblinski-Tang function. The results were more promising this time, as shown in Figure 3.

4.3. Simulation experiments on non-symmetric benchmark functions

Most of the algorithms would fail to optimize the non-symmetric benchmark functions^[8]. The non-symmetry refers to the characteristics that the profiles of the benchmark functions are not axial symmetric or mirror symmetric through the whole domain. Due to the fact that most of the algorithms would fail to optimize the non-symmetric benchmark functions, symmetry might be another characteristic to describe the benchmark functions as modality, separability, scalability and differentiability^[9].

In this experiment, Egg Holder function was introduced:

$$f(x) = 959.64 - \sum_{i=1}^{d-1} \left[(x_{i+1} + 47) \sin \sqrt{|x_{i+1} + \frac{x_i}{2} + 47|} + x_i \sin \sqrt{|x_i - (x_{i+1} + 47)|} \right] \tag{13}$$

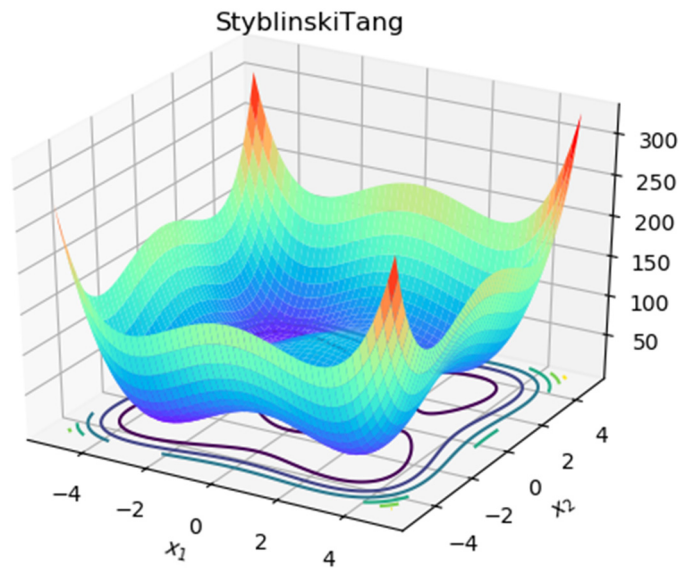


Figure 2 Three dimensional profile for Styblinski-Tang function

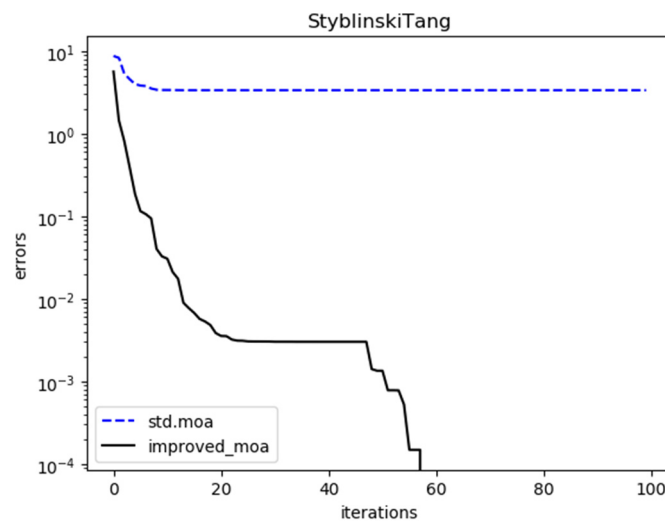


Figure 3 Simulation experiments results versus iterations

The global optimum for Egg Holder function is located at point $x^* = (512,404.2319)$, and $f(x^*) = 0$. Egg Holder function was not only non-symmetric in profiles, it was also multimodal, the three dimensional profile would be very complicated, as shown in Figure 4.

However, simulation experiments demonstrated neither of the results would be satisfactory. While the improved MO algorithm would perform better than the original version, as shown in Figure 5.

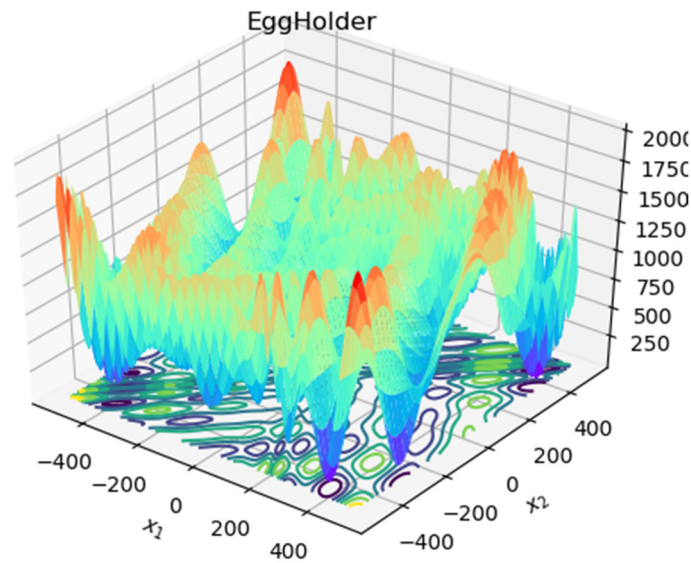


Figure 4 Three dimensional profile for Egg Holder function

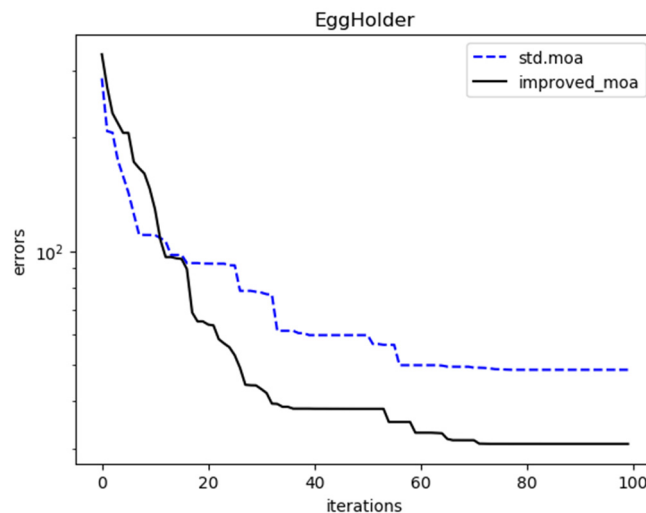


Figure 5 Simulation experiments results versus iterations

5. DISCUSSIONS AND CONCLUSIONS

The MO algorithm was just proposed recently. Simulation experiments in literature demonstrated that it would perform better in optimization both the benchmark functions and the engineer problems in real world. However, after a detailed study on the updating equation for individuals in swarms, we would find that the structure might embrace a little departure from our normal recognition.

Reconstructing the updating equations for individuals in mayfly swarms, we proposed an improvement for the MO algorithm. Simulation experiments were carried out and all of the results proved that the improved MO algorithm would perform better than the original one.

We have discovered that some of the non-symmetric benchmark functions could be optimized with satisfactory solutions. However, simulation experiments on the non-symmetric benchmark functions in this paper also proved that not all of the non-symmetric benchmark functions could be optimized by the MO algorithm or its improvements.

We can draw the conclusions that the improved algorithm could indeed increase the performance of the MO algorithm, however, efforts remain in needed because this newly raised algorithm could not perform all well for all of the benchmark functions.

ACKNOWLEDGMENTS

The authors would like to thank the supports of the following projects: The second batch of scientific research team of Jingchu University of technology with grant number TD202001. The general Excellent Students Work Funding Project of Hubei Provincial Colleges with grant number 2019XGJPB3013. The key research and development project of Jingmen with grant number 2019YFZD009 and 2020YFYB033. Hubei Provincial Natural Science Foundation with grant number 2019CFB661. The research project of Hubei Provincial Department of Education with grant number B2019213. The cultivatable science foundations of Jingchu University of technology with grant number PY201903 and PY202003.

REFERENCES

- [1] Tülin İnkaya, Sinan Kayalığıl, Nur Evin Özdemirel. Ant Colony Optimization based clustering methodology[J]. *Applied Soft Computing*, 2015, 28 (0): 301-311.
<http://dx.doi.org/10.1016/j.asoc.2014.11.060>
- [2] Juan Zhao, Zheng Ming Gao. The bat algorithm and its parameters: The 4th International Conference on Electronic, Communications and Networks (CECNet2014) Boca Raton : CRC Press, [2014], 2015[C]. Boca Raton : CRC Press, [2014], IV: 1323-1326. 10.1201/b18592-237
- [3] Yacine Labbi, Djillani Ben Attous. A hybrid particle swarm optimization and pattern search method to solve the economic load dispatch problem[J]. *International Journal of System Assurance Engineering and Management*, 2014, 5 (3): 435-443. 10.1007/s13198-013-0186-1
- [4] Seyedali Mirjalili, Seyed Mohammad Mirjalili, Andrew Lewis. Grey Wolf Optimizer[J]. *Advances in Engineering Software*, 2014, 69 (0): 46-61.
<http://dx.doi.org/10.1016/j.advengsoft.2013.12.007>
- [5] Afshin Faramarzi, Mohammad Heidarinejad, Brent Stephens, et al. Equilibrium optimizer: A novel optimization algorithm[J]. *Knowledge-Based Systems*, 2019: 105190.
<https://doi.org/10.1016/j.knosys.2019.105190>
- [6] Juan Zhao, Zhengming Gao. The Improved Equilibrium Optimization Algorithm with Best Candidates[J]. *Journal of Physics: Conference Series*, 2020, 1575: 012089. 10.1088/1742-6596/1575/1/012089
- [7] Konstantinos Zervoudakis, Stelios Tsafarakis. A mayfly optimization algorithm[J]. *Computers & Industrial Engineering*, 2020, 145: 106559. <https://doi.org/10.1016/j.cie.2020.106559>
- [8] Peifeng Niu, Songpeng Niu, Nan liu, et al. The defect of the Grey Wolf optimization algorithm and its verification method[J]. *Knowledge-Based Systems*, 2019, 171: 37-43.
<https://doi.org/10.1016/j.knosys.2019.01.018>
- [9] Momin Jamil, Xin-She Yang. A literature survey of benchmark functions for global optimization problems[J]. *Int. Journal of Mathematical Modelling and Numerical Optimisation*, 2013, 4 (2): 150-194.