The improved mayfly optimization algorithm with opposition based learning rules

Zheng-Ming GAO¹, Juan ZHAO², Su-Ruo LI¹, Yu-Rong HU³

¹School of computer engineering, Jingchu University of technology, Jingmen, 448000, China
²School of electronics and information engineering, Jingchu University of technology, Jingmen, 448000, China
³Department of science and technology, Jingchu University of technology, Jingmen, 448000, China
gaozming@jcut.edu.cn

Abstract. Literal researches proved that not only the best candidates or the best historical trajectories would perform well in guiding the individuals in swarms to find the best solution, the worst and the worst historical trajectories would also work well in doing so. Such situations could be directly treated as pairs of oppositions, and satisfied the ancient Chinese Yin-Yang philosophy, where the opposition based learning (OBL) rule was directly derived from. In this paper, the improved mayfly optimization (MO) algorithm with OBL rules were proposed, simulation experiments were carried out and results showed that the improved MO algorithm with OBL rules would perform better than usual.

1. Introduction
Literally speaking, almost all of the algorithms were involving the positive information, for example, the averages in the ant colony optimization (ACO) algorithm[1] or the bat algorithm[2], or the historical best trajectories, the global best candidates in the particle swarm optimization (PSO) algorithm[3], the equilibrium optimization (EO) algorithm[4], the grey wolf optimization (GWO) algorithm[5], the Harris hawk optimization (HHO) algorithm[6]. However, sometimes the worst or historical worst candidates could also be introduced to guide the individuals in swarms in optimization, simulation experiments showed that the negative PSO algorithm[7] would also perform well in optimization, especially the multimodal benchmark functions.

The ancient Chinese Yin-Yang philosophy shows us that there would be a strong relationship between the Yin and Yang sides, or right and wrong, negative and positive, fire and water, dry and wet[8]. So, the opposition based learning (OBL) rule was proposed[9]. Literal researches proved that most of the algorithms could be improved with OBL rules and the improved algorithms would perform better than usual.

In this paper, we would introduce the OBL rule to the newly proposed mayfly optimization (MO) algorithm (MOA). The following sections would be arranged as follows: in Section 2, we would describe the MO algorithm briefly. In section 3 the improved MO algorithm with OBL rules (OBL-MOA) would be proposed. Simulation experiments would be carried out in Section 4. Discussions would be made and conclusions would be drawn in Section 5.
2. The MO algorithms
In this section, the OBL-MO algorithm would be proposed as an improvement of the MO algorithm.

For the MO algorithm, they would be split into two types, the male mayfly $x_i(t)$ for $i$-th individual in the current iteration $t$, and the female mayfly $y_i(t)$. They all updated their positions with their velocities $v_i(t)$ in the current iteration:

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

The male and female mayflies would update their velocity with different ways:

For the female mayflies:

$$v_i(t) = g \cdot v_i(t) + a_1 e^{-\mu r_{mf}} [x_i(t) - y_i(t)] f[y_i(t)] > f[x_i(t)]$$

Where $f(x)$ is the fitness function. $g$ and $f_l$ are weights which would be declined from their maximum to minimum value. $a_1$ and $\beta$ are constants. $r_{mf}$ represents the Cartesian distance between the female and male couple, it would be calculated as follows:

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

For the male mayflies:

$$v_i(t) = \begin{cases} 
    g \cdot v_i(t) + a_2 e^{-\mu r_{m}} [x_{h_i} - x_i(t)] + a_2 e^{-\mu r_{b}} [x_g - x_i(t)] & f[x_i(t)] > f[x_{h_i}(t)] \\
    g \cdot v_i(t) + d \cdot r_2 & f[x_i(t)] \leq f[x_{h_i}(t)]
\end{cases}$$

Where, $a_2$ and $a_3$ are two other constants. $x_{h_i}$ represents the $i$-th historical best trajectory. $r_p$ and $r_g$ represent the Cartesian distance between $x_i(t)$ and $x_{h_i}$, the global best candidate $x_g$ respectively. $d$ represents the dance ratio around the current position and $r_2$ is another random number in uniform distribution with interval of -1 and 1.

When the positions of mayflies have finished the updating, they would again reselect themselves. Half of the top groups were elected as male mayflies and the rest of them would be treated as the female mayflies. And then, the top pair would be nominated as pairs of mates, the best male mates the best female mayflies, the second best male mates the second female mayflies, and so on. They would then give birth to their offspring with the following equations:

$$\text{offspring1} = L \ast \text{male} + (1 - L) \ast \text{female}$$

$$\text{offspring2} = L \ast \text{female} + (1 - L) \ast \text{male}$$

The offspring would also be mutated as well, and when the current iteration come to the end, the offspring would grow up, they would be sorted according to their fitness values and then be nominated as male or female again for the next iteration.

3. The OBL-MO algorithm
Considering the Yin-Yang philosophy, we could find that if the current position $p_i(t)$ was far away from the best, then, the opposite position $op_i(t)$ would near the best in a given definitional domain. Therefore, we can calculate the opposite to the current position:

$$op_i(t) = a + b - p_i(t)$$

And then, if $f[\text{op}_i(t)] < f[p_i(t)]$, we can switch the opposite directly and consequently, the individuals would be more nearer to the global optimum and thus, the exploration procedure would be increased in speed.

This is the basic idea of the OBL rule, simulation experiments have verified its capability in improvements.
4. Simulation experiments

In this section, we would carry out some experiments to verify whether the improved OBL-MO algorithm would be more capable in optimizing or not. The Monte Carlo simulation experiments were also involved to reduce the influence of randomness, which would cause fluctuation of results.

Three kinds of simulation experiments on the unimodal, multimodal benchmark functions\[^{[10]}\] and those who have basins in their profiles would be carried out. And another experiment on the non-symmetric benchmark function would be also involved to make sure the OBL-MO algorithm would not have defects\[^{[11]}\]

4.1. Simulation experiments on unimodal benchmark functions

Ackley 2 function would be a representative for the unimodal benchmark functions:

\[
f(x) = 200 - 200e^{-0.02\sqrt{x_1^2 + x_2^2}} + e^{-\frac{1}{3}(\sin(3\pi x_1) + \sin(\pi x_2))}
\]  

(8)

Ackley 2 function is unimodal and its global optimum is located at the Origin, as shown in Figure 1. The Monte Carlo simulation results were shown in Figure 2. Apparently, results proved that the improved MO algorithm with OBL rules labelled ‘OBL_MOA’ would perform better than the original MO algorithm labelled ‘MOA’ in Figure 2. A simple result would demonstrate the capability of OBL-MOA for unimodal benchmark functions.

4.2. Simulation experiments on multimodal benchmark functions

In this experiment, Davis’ function would be introduced as the representative:

\[
f(x) = (x_1^2 + x_2^2)^{0.25}\sin^2[50(3x_1^2 + x_2^2)^{0.1}] + 1
\]  

(9)

Davis’ function is highly multimodal, as shown in Figure 3. The Monte Carlo simulation results were shown in Figure 4. The results also verified that the OBL-MO algorithm would also work well in optimizing the multimodal benchmark functions.

4.3. Simulation experiments on benchmark functions with basins

Sometimes we would find that some of the algorithms would fail to find the best solutions even the functions are unimodal. The basins or valleys would increase the difficulty in optimization. Therefore, we would carry on this specific experiment on the benchmark functions who have basins or valleys in their profiles, a representative as Hyper-Ellipsoid function:

\[
f(x) = \sum_{i=1}^{d} i^2x_i^2
\]  

(10)

Hyper-Ellipsoid function is unimodal, but there is a basin around the global optimum, as shown in...
4.4. Simulation experiments on non-symmetric benchmark functions

Literal researches proved that most of the swarm based algorithm have difficulties to find the best solution for non-symmetric algorithms. In this experiment, the Mishra 3 function would be chosen as the representative:

\[
f(x) = 1.17777 - \sqrt{\cos(\sqrt{x_1^2 + x_2^2})} + 0.01(x_1 + x_2)
\]  

Mishra 3 function is highly multimodal is highly multimodal and it is also non-symmetric, as shown in Figure 7. The global optimum for Mishra 3 function locates at point \( x^* = (-8.46613775046579, -9.998213089999999) \) and \( f(x^*) = 0 \). The Monte Carlo simulation experiments’ results were shown in Figure 8. Results showed that the OBL-MO algorithm would also perform better than the original version of the MO algorithm.
5. Discussions and Conclusions
In this paper, we proposed the OBL rules to the MO algorithm, and carried out four kinds of simulation experiments. Results showed that the OBL-MO algorithm would all perform better in optimization than the original MO algorithm. The residual errors would be decreased dramatically along with iterations, which meant that the performance of the MO algorithm would be increased extraordinarily with the OBL rule.

The individuals in almost all of the algorithms would update their positions or concentrations or velocities according to the specific information of averages, best candidates or historical best trajectories. Random walk\[12\] or Levy flight\[13\] might be involved in for the best candidates to take a further step in iterations, but the overall performance would not be increased so much as done by the OBL rule.

Considering the OBL rule is easy to understand and code when needed, we could draw a conclusion that the OBL rule should be added to increase the capability of algorithms in applications as capable as we can.

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

References
[1] Tülin İnkaya, Sinan Kayaloglu, Nur Evin Özdemirol. 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] J. Kennedy, R. Eberhart. Particle swarm optimization: Proceedings of ICNN'95 - International Conference on Neural Networks, 27 Nov.-1 Dec. 1995, 1995[C]. 4: 1942-1948 vol.4. 10.1109/ICNN.1995.488968

[4] Juan ZHAO, Zheng-Ming GAO. The Improved Equilibrium Optimization Algorithm with Levy Flight: 2020 5th International Conference on Computer and Communication Systems (ICCCS), 15-18 May 2020, 2020[C]. 352-355. 10.1109/ICCCS49078.2020.9118502

[5] 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

[6] Ali Asghar Heidari, Seyedali Mirjalili, Hosssam Faris, et al. Harris hawks optimization: Algorithm and applications[J]. Future Generation Computer Systems, 2019. HTTPS://DOI.ORG/10.1016/J.FUTURE.2019.02.028

[7] Yang Chunming, D. Simon. A new particle swarm optimization technique: 18th International Conference on Systems Engineering (ICSEng'05), 16-18 Aug. 2005, 2005[C]. 164-169. 10.1109/ICSENG.2005.9

[8] Sedigheh Mahdavi, Shahryar Rahnamayan, Kalyanmoy Deb. Opposition based learning: A literature review[J]. Swarm and Evolutionary Computation, 2018, 39: 1-23. HTTPS://DOI.ORG/10.1016/J.SWEVO.2017.09.010

[9] H. R. Tizhoosh. 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), 28-30 Nov. 2005, 2005[C]. 1: 695-701. 10.1109/CIMCA.2005.1631345

[10] 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.

[11] 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

[12] Juan Zhao, Zheng-Ming Gao. A Random Walk Equilibrium Optimization Algorithm: 2020 3rd International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE), 24-26 April 2020, 2020[C]. 22-25. 10.1109/AEMCSE50948.2020.00012

[13] Juan Zhao, Zheng-Ming Gao, Wu Sun. The improved slime mould algorithm with Levy flight[J]. Journal of Physics: Conference Series, 2020, 1617: 012033. 10.1088/1742-6596/1617/1/012033