



Firefly Algorithm with Opposition-Based Learning

Yanping Qiao^{1,2(✉)}, Feng Li^{3,4}, Cong Zhang⁵, Xiaofeng Li⁴, Zhigang Zhou⁴,
Tao Zhang^{3,4}, and Quanhua Zhu⁴

¹ School of Power and Energy, Northwestern Polytechnical University,
Xi'an 710072, China

qiaoypnwpu@qq.com

² Science and Technology on Altitude Simulation Laboratory,
Mianyang 621700, China

³ Southern Marine Science and Engineering Guangdong Laboratory,
Guangzhou 511458, China

⁴ China Ship Scientific Research Center, Wuxi 214082, China

⁵ AECC Sichuan Gas Turbine Establishment, Mianyang 621700, China

Abstract. The firefly algorithm (FA) is a swarm intelligence optimization algorithm based on the firefly's glow and attractive behavior. It possesses a simple design, is easy to implement, and has been applied in many engineering fields. Although FA well solves complex optimization problems, it has flaws regarding premature convergence and overall performance. Its learning style is relatively simple, as it can update individual positions merely by the fluorescence intensity between individuals. This paper proposes the firefly algorithm with opposition-based learning (FA-OL), which enriches learning to improve its capability to jump out of local optima. The opposition-based learning (OL) mechanism can promote individuals to search from the current and opposite points in the search space, and can quickly broaden the solution space. The optimal random parameter settings of FA-OL are obtained through experiments. Comprehensive experiments on nine commonly used test functions compare the proposed algorithm with the latest improved FA and seven other algorithms. The experimental results show that the introduction of OL greatly improved the search capability of FA, and FA-OL achieved better results than other algorithms.

Keywords: Firefly algorithm · Opposition-based learning · Global optimization

This work is supported by a project commissioned by AECC Sichuan Gas Turbine Establishment and by Key Special Project for Introduced Talents Team of Southern Marine Science and Engineering Guangdong Laboratory (Guangzhou, No. GML2019ZD0502).

© Springer Nature Singapore Pte Ltd. 2022

L. Pan et al. (Eds.): BIC-TA 2021, CCIS 1565, pp. 342–352, 2022.

https://doi.org/10.1007/978-981-19-1256-6_26

1 Introduction

Metaheuristic algorithms have been widely used to solve complex and highly nonlinear optimization problems. Among them, swarm intelligence optimization, such as particle swarm optimization, ant colony optimization, and the firefly algorithm (FA), simulates group behavior in nature, and has performed well at solving complex optimization problems. Proposed in 2008 by Dr. Xinshe Yang of the University of Cambridge, FA simulates the group behavior of fireflies. Each firefly positions itself by its own fluorescence, through whose intensity it can attract other fireflies, catch prey, or send out warning signals. It can move individually to complete a search process in a solution space. The algorithm has seen many improvements. An improved FA to learn between mates was proposed, expanding upon the original FA, which has a single-sex firefly, and improving its optimization performance [14]. Wang et al. [23] proposed Ying-Yang FA based on a dimensional Cauchy mutation. A randomized attraction model promoted convergence. To better use FA to solve global continuous optimization problems, Wu et al. designed a logarithmic spiral search method to improve the individual's exploration capability, and a dynamic search method to improve individual exploration capability [25]. Nand et al. [11] proposed an improved FA called FA-Step, and designed a new step search strategy and a step strategy mixed with CMAES. Liu et al. proposed dynamic adaptive FA, which improved the convergence speed and solution accuracy, and stopped the algorithm from suffering from premature convergence [9].

FA has been successfully used in many engineering applications. A hybrid cooperative FA was used to solve the restricted vehicle routing problem, achieving a good balance between intensification and diversity [1]. Cheng et al. proposed a hybrid algorithm based on FA and GA, whose group attraction operator reduced the time complexity. Three combined mutation strategies were introduced, and the algorithm was used for constrained optimization problems. Tao et al. [18] proposed an adaptive strategy in the FA attraction model to design random models and penalty functions. It was used to solve constrained engineering optimization problems, where it achieved rapid convergence and high accuracy. Krishna et al. designed an improved random attraction FA for fuzzy image clustering problems [6]. Tian et al. [20] improved and optimized individual fireflies by combining mutation operations, used them to guide the direction of the group's operation, and applied the algorithm to the design of the SMC-PHD filter. Xie et al. [26] proposed two improved methods based on FA to overcome the local convergence problem in k-means clustering. Ireneus applied FA to the problem of entity selection in machine learning [4]. Wang et al. [24] designed an improved chaotic FA for wireless sensor resource allocation in the IoT environment. Dimitra et al. [22] designed an individual encoding and decoding strategy to use FA for the prize-collecting vehicle routing problem. Cheng et al. [2] improved the FA attraction model to promote convergence at a lower cost, designed three staged adaptive firefly functions to determine its parameters, and used the designed SAFA for the UAV charging planning problem in wireless sensor networks. Dash et al. designed an improved FA-based optimal design of special signal blocking

IIR filters [5]. Rohulla et al. designed a restricted multi-objective optimization model for the privacy protection problem in social networks, and combined FA and fuzzy clustering to solve it [8]. Tian et al. [19] designed an improved FA for particle filtering and used it for multi-target tracking, achieving better accuracy.

This paper addresses the problem of FA easily falling into local optima. Our improved FA based on opposition-based learning (OL), FA-OL, broadens the search space of the algorithm and improves its ability to cover the feasible solution area. The remainder of this article is organized as follows. Section 2 introduces the principle of the standard FA and FA based on the offline decreasing random factor. Section 3 introduces the principle of OL and its application to FA. Section 4 relates the results of experiments comparing our algorithm to seven others, and discusses the randomized parameter-setting method that affects our algorithm. Section 5 discusses our conclusions.

2 Introduction to FA

2.1 FA Concept

The two most basic behaviors of fireflies are attracting fireflies of the opposite sex in the same population and attracting prey, both of which depend on the fluorescence they emit, which is the fundamental principle of FA design. We know from optics that the intensity of light is attenuated because it is absorbed by the air, and the degree of attenuation is directly related to the air absorption and distance of propagation. The intensity of light will also decrease as the distance between the light source and the object increases.

In FA, each feasible solution is expressed as the position of each firefly, and the fluorescence intensities of the fireflies is used as the target function value. In standard FA, the fireflies are all of the same sex, so the attractive relationship between them does not need to be considered. In addition, FA believes that the degree of attraction between fireflies is proportional to their own fluorescence intensity. Individuals with lower fluorescence intensity are attracted to fireflies with higher fluorescence intensity, and will fly to them. The attraction behavior between fireflies is also the convergence behavior of the algorithm, i.e., the fireflies are searching for areas with better target function values.

The individual position update equation of FA has three parts, namely the individual flight inertia; the movement of the i -th firefly attracted by the j -th firefly is equivalent to the convergence behavior of the firefly in the evolutionary process; the randomized search term, which is completely unrelated to fireflies i and j . Through this, random disturbances can be generated to make the fireflies reach a new search position, as follows:

$$x_i^k(t+1) = x_i^k(t) + \beta_{ij}(t) \cdot (x_j^k(t) - x_i^k(t)) + \alpha(t) \cdot (rand_{ij}^k(t) - 0.5) \quad (1)$$

$$\beta_{ij}(t) = \beta_0 \cdot e^{-\gamma r_{ij}^2(t)} \quad (2)$$

$$r_{ij}(t) = \|x_j(t) - x_i(t)\| = \sqrt{\sum_{k=1}^D (x_j^k(t) - x_i^k(t))^2} \quad (3)$$

where $x_i^k(t)$ and $x_j^k(t)$ respectively refer to the positions of fireflies i and j in the k -th dimension of the t -th iteration; $r_{ij}(t)$ is the distance between fireflies i and j in the t -th iteration; D is the problem scale; $rand$ is a uniformly distributed random number in $[0, 1]$; α is a random factor; γ is the absorption coefficient; β represents the attraction between two fireflies; and β_0 is the attraction when the distance between two fireflies is zero, i.e., the maximum attraction between them. Regarding the parameters of FA, Yang [27, 28] proposed that, when implementing and applying FA, $\beta_0 = 1$, $\gamma \in [0.01, 100]$ is a constant generally set to 1, and $\alpha \in [0, 1]$ is a constant generally set to 0.5.

2.2 Improved FA Based on a Proportionally Decreasing Random Factor

From the position update equation of FA, we know that when the absorption coefficient is zero, the update equation is

$$x_i^k(t+1) = x_j^k(t) + \alpha(t) \cdot (rand_{ij}^k(t) - 0.5) \quad (4)$$

and when the absorption coefficient tends to infinity, the update equation is

$$x_i^k(t+1) = x_i^k(t) + \alpha(t) \cdot (rand_{ij}^k(t) - 0.5) \quad (5)$$

It can be seen from formulas (4) and (5) that whether the fluorescence is absorbed or not, the random search term is an important part of the firefly position update, and the random search ability will be affected by the random factor. We carried out research on random factors, and propose control methods based on linear decline, proportional decline, parabolic decline, and fixed constants. Through the experimental and statistical analysis results, it can be known that the improved FA adopting the proportional decreasing method has the best comprehensive optimization performance among the compared algorithms. The proportional decreasing method of random factors has the formula

$$\alpha(t+1) = k \cdot \alpha(t), \alpha(0) = 0.5 \quad (6)$$

where k is the common ratio, which is a constant less than 1. Optimizing the standard test function with the value of k in the interval $[0.97, 0.998]$ can obtain a more reasonable result. Through experiments, it was concluded that the best optimization performance occurs when k is 0.9902 [15]. FA based on a proportionally decreasing random factor is referred to as FA-Prop in this article.

3 Firefly Algorithm with Opposition-Based Learning

3.1 OL Principle

Generally speaking, heuristic algorithms are iterative. Starting from an initial solution, after continuous iterative operations, a heuristic algorithm tries to improve candidate solutions. When the algorithm stop condition is met, the search process is terminated. When the prior information of the solution was lacking, Tizhoosh et al. [21] proposed an OL mechanism, which used a random guess to generate candidate solutions, and whose calculation time was related to the distance between the initial and optimal solutions. Optimization algorithms with superior performance often have a high probability of jumping out of local optima and finding the global optimum, which is closely related to the ability of the particles in the algorithm to explore and open up unknown areas. OL can effectively broaden the search space and cover the feasible solution area.

Definition 1 Opposite number. If $x \in [a, b]$ and $x \in R$ then the opposite number x is $x^* = a + b - x$.

Definition 2 Opposite point. If $P = (x_1, x_2, \dots, x_D)$ is a point in a D -dimensional space, $x_i \in R$, and $x_i \in [a_i, b_i]$ ($i = 1, 2, \dots, D$), then P is the corresponding opposite point, where $P^* = (x_1^*, x_2^*, \dots, x_D^*)$ and $x^* = a + b - x$.

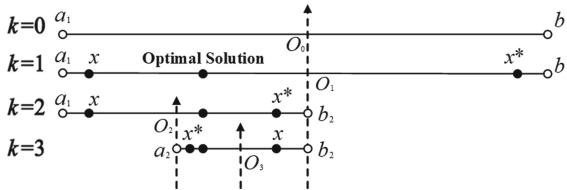


Fig. 1. Opposition-based learning in one dimension [16]

Opposition-based optimization thinking: Assume $P = (x_1, x_2, \dots, x_D)$ is a point in a D -dimensional space (it can be viewed as a candidate solution), with opposite point $P^* = (x_1^*, x_2^*, \dots, x_D^*)$. Suppose $f(\cdot)$ is an evaluation function to calculate the fitness of the candidate solution. Calculate $f(P)$ and $f(P^*)$ in each iteration. If $f(P^*) \leq f(P)$, then replace P with P^* , and otherwise still iterate with P . The one-dimensional space OL process is shown in Fig. 1, where O_k is the k -th OL reference point. Its value is the midpoint of the sum of a_i and b_i . It is also called the symmetric OL mechanism. If you take a random position O_k , it is random OL.

Zhou et al. proposed normalized OL [29]. Let $\varphi(x) = \Delta - x$ and $x^* = \Delta - x$, where Δ is a calculated value and the corresponding opposite solution is $x \in [\Delta - b, \Delta - a]$. Compared with the original OL, the difference is the symmetric position of the current and opposite points. The current and opposite points of the original OL are with regard to $(a + b)/2$, while the normal OL is symmetric

with regard to $(2\Delta - a - b)/2$. Let $\Delta = \eta(a + b)$, where η is a real number. Then normal OL is

$$x^* = \eta(a + b) - x \tag{7}$$

According to the different values of η , there are four types of OL. When $\eta = 0$, the current and opposition-based solutions are symmetric with regard to the origin, which is called OL based on solution symmetry; when $\eta = 1/2$, the search interval is symmetric about the origin, which is called OL based on interval symmetry; $\eta = 1$ is the special case of the original OL; and when η is a random number, it is called random OL.

In the swarm intelligence algorithm, the OL mechanism is introduced, and the individual can search at the current and opposite points at the same time. If the opposite point has a better fitness value than the current point, then the search area near the opposite point has a better chance to find the optimal value. The individual can jump directly to the opposite point to continue searching, which can quickly expand the solution space and cover the feasible solution area on a larger scale, ultimately increasing the probability of finding the global optimal solution.

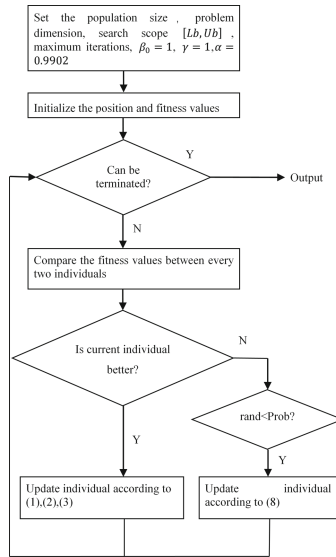


Fig. 2. Flowchart of FA-OL

3.2 Opposition Learning FA

In the standard FA and FA-Prop, when comparing the fluorescence intensity of two individual fireflies, the individual with weaker fluorescence intensity needs to learn from the individual with high fluorescence intensity, i.e., according to formulas (1)–(3), complete the location update of the weaker individuals. In FA-OL,

Table 1. Experimental results of FA-OL on different random parameters

	F1	F2	F3	F4	F5	F6	F7	F8	F9
FA_Prop	2.38E-13 3.97E-14	1.70E-03 8.17E-03	8.83E+01 1.40E+02	-7.97E+03 7.16E+02	6.67E+01 2.09E+01	1.12E-07 9.20E-09	2.37E-03 3.99E-03	6.68E-16 1.22E-16	1.32E-03 3.64E-03
FA-OL (Prob = 0.1)	0 0	0 0	2.72E+01 1.33E-01	-2.16E+02 6.22E+01	0 0	-8.88E-16 0	4.75E+01 3.92E-01	1.81E-15 5.22E-16	4.39E-04 2.20E-03
FA-OL (Prob = 0.5)	0 0	0 0	2.72E+01 1.32E-01	-1.62E+02 2.57E+01	0 0	-8.88E-16 0	5.93E+01 1.69E-01	1.82E-15 4.16E-16	4.39E-04 2.20E-03
FA-OL (Prob = 0.01)	0 0	0 0	2.73E+01 1.68E-01	-6.96E+03 7.91E+02	0 0	-8.88E-16 0	2.96E-04 1.48E-03	1.11E-15 3.12E-16	1.32E-03 3.64E-03
FA-OL (Prob = 0.02)	0 0	0 0	2.73E+01 1.47E-01	-4.48E+03 7.84E+02	0 0	-8.88E-16 0	2.96E-04 1.48E-03	1.28E-15 2.94E-16	1.32E-03 3.64E-03
FA-OL (Prob = 0.03)	0 0	0 0	2.72E+01 1.50E-01	-3.32E+03 5.81E+02	0 0	-8.88E-16 0	5.92E-04 2.05E-03	1.41E-15 3.27E-16	8.79E-04 3.04E-03
FA-OL (Prob = 0.04)	0 0	0 0	2.73E+01 1.33E-01	-2.66E+03 3.71E+02	0 0	-8.88E-16 0	4.20E-02 4.95E-02	1.75E-15 3.79E-16	4.39E-04 2.20E-03
FA-OL (Prob = 0.05)	0 0	0 0	2.73E+01 1.54E-01	-1.87E+03 4.42E+02	0 0	-8.88E-16 0	1.57E+00 1.30E-01	1.64E-15 4.67E-16	1.76E-03 4.11E-03

Table 2. Ranking results of FA-OL on different random parameters (ANOVA)

	F1	F2	F3	F4	F5	F6	F7	F8	F9	Total
FA_Prop	8	1	8	1	8	8	4	1	1	40
FA-OL (Prob = 0.1)	1	1	1	7	1	1	7	7	1	27
FA-OL (Prob = 0.5)	1	1	1	8	1	1	8	8	1	30
FA-OL (Prob = 0.01)	1	1	1	2	1	1	1	2	1	11
FA-OL (Prob = 0.02)	1	1	1	3	1	1	1	3	1	13
FA-OL (Prob = 0.03)	1	1	1	4	1	1	1	4	1	15
FA-OL (Prob = 0.04)	1	1	1	5	1	1	5	6	1	22
FA-OL (Prob = 0.05)	1	1	1	6	1	1	6	5	1	23

to improve the ability of the algorithm to jump out of local optima, individuals with higher fluorescence intensity expand the search space with probability *Prob* through the OL mechanism, i.e., the position update is completed as

$$x_i^k(t + 1) = Ub + Lb - x_i^k(t) + rand(1, D) .* x_i^k(t) \tag{8}$$

where *Ub* and *Lb* are the upper and lower bounds, respectively, of the search space, and *D* is the problem dimension. It can be seen from formula (8) that after an individual performs an opposition search, it can quickly search for the best point in a new search area, which improves the global search capability of the algorithm. The flowchart of FA-OL is shown as Fig. 2.

4 Experiment

4.1 Experiment Setup

We used nine standard test functions [15] to analyze the performance of FA-OL. We also compared it with the latest improved FA, FACL [14]; improved

ant colony optimization algorithm NABC [12]; and improved cuckoo algorithm MSSCS [13]; and with classic improved algorithms PSO with inertia weight (PSO-In) [17], PSO with constriction factor (PSO-Co) [3], Gaussian Bare Bone PSO (GBBPSO) [7], and PSO with lbest (PSO-Lb) [10]. The dimension of the test function was $D = 30$, the population size of each algorithm was 30, and the maximum number of evaluations of the target function was 60,000. Other parameters of each algorithm adopted the recommended values in the corresponding literature. Each algorithm independently and randomly optimized each function 25 times, and the average and standard deviation optimal results were recorded.

Table 3. Comparison results of FA-OL with other 7 algorithms

	F1	F2	F3	F4	F5	F6	F7	F8	F9
FA-OL (Prob = 0.01)	0 0	0 0	2.73E+01 1.68E-01	-6.96E+03 7.91E+02	0 0	-8.88E-16 0.00E+00	2.96E-04 1.48E-03	1.11E-15 3.12E-16	1.32E-03 3.64E-03
FA-CL	1.23E+00 2.88E-01	3.06E+01 1.33E+01	3.07E+02 3.45E+02	-4.38E+03 3.19E+02	1.50E+02 1.60E+01	5.49E+00 1.60E+00	8.91E-02 2.34E-02	3.15E+00 1.48E+00	3.80E+01 2.48E+01
MSSCS	1.53E-37 7.23E-37	1.66E-04 4.18E-04	1.84E+01 1.21E+01	-1.26E+04 1.24E-05	1.95E-04 4.75E-04	7.50E-15 2.49E-15	1.39E-04 6.97E-04	1.15E-12 8.67E-13	8.79E-04 3.04E-03
NABC	1.29E-23 1.19E-23	1.38E+03 5.30E+02	2.05E+00 2.94E+00	-1.26E+04 1.44E-05	1.56E-14 4.40E-14	1.02E-11 3.88E-12	2.72E-12 1.31E-11	3.31E-24 8.45E-24	1.25E-22 1.48E-22
PSO_In	1.94E-09 3.97E-09	1.69E+03 1.41E+03	1.54E+02 1.88E+02	-3.20E+04 4.43E+03	4.05E+01 1.21E+01	1.90E-05 1.81E-05	1.05E-02 1.53E-02	5.39E-02 6.77E-02	3.52E-03 5.23E-03
PSO_CO	3.52E-03 5.23E-03	5.19E-01 8.71E-01	4.88E+01 4.15E+01	-2.11E+04 3.03E+03	6.42E+01 1.92E+01	2.19E+00 1.24E+00	2.40E-02 2.65E-02	1.29E-01 1.67E-01	2.84E-02 1.24E-01
GBPSO	2.99E-31 6.20E-31	1.49E+04 6.11E+03	1.48E+04 3.35E+04	-8.92E+03 6.38E+02	8.45E+01 2.64E+01	1.83E+00 5.37E+00	2.51E+00 1.25E+01	4.15E-02 6.69E-02	3.52E-03 5.23E-03
PSO_LB	1.41E-05 1.55E-05	3.63E+03 1.16E+03	8.79E+01 5.26E+01	-3.17E+04 3.85E+03	5.05E+01 9.77E+00	1.79E-01 5.22E-01	3.84E-03 7.90E-03	7.97E-02 1.05E-01	4.70E-02 7.25E-02

4.2 Effect Analysis of Random Parameters

In the proposed FA-OL, the execution of OL depends on the random parameter Prob. To clearly understand the impact of its setting on the optimization performance of the algorithm, we set its value to 0.5, 0.1, 0.05, 0.04, 0.03, 0.02, and 0.01. The nine test functions were solved randomly and independently 25 times, with average and standard deviation as displayed in Table 1. The first row of data corresponding to each algorithm is the average value of the optimal solution, and the second row is the standard deviation. It can be seen from Table 1 that the introduction of OL significantly improved the optimization performance of FA on most test functions, especially on the functions of F1, F2, F5, and F6, on which higher solution accuracy was attained.

We used analysis of variance (ANOVA) to test for significant differences between the mean value of the algorithm for each test function at the level of 0.05. The score of each algorithm on each function is shown in Table 2. At the same time, the total score of each algorithm was calculated. The smaller the score the better the comprehensive optimization performance. It can be seen from Table 2 that the comprehensive optimization performance of FA-OL is best when the random parameter value is 0.01.

Table 4. Ranking results of 8 algorithms (ANOVA)

	F1	F2	F3	F4	F5	F6	F7	F8	F9	Total
FA-OL (Prob = 0.01)	1	1	3	7	1	1	1	2	1	18
FA-CL	8	4	7	8	8	8	7	8	8	66
MSSCS	1	1	2	4	3	2	1	3	1	18
NABC	4	5	1	4	1	3	1	1	1	21
PSO_In	5	5	6	1	4	4	4	5	4	38
PSO_CO	7	3	4	3	6	4	6	7	4	44
GBPSO	3	8	8	6	7	4	7	4	4	51
PSO_LB	6	7	5	1	5	4	4	6	4	42

4.3 Comparison with Other Algorithms

Table 3 shows the experimental results of FA-OL and the other seven algorithms, and Table 4 shows their scores on the nine test functions. It can be seen from Table 3 that the proposed algorithm achieves the best optimization results on six functions (F1, F2, F5, F6, F7, and F9), especially in F1, F2, and F5 where the position of the optimal solution was found, showing a strong global search capability. From the statistical scoring results in Table 4, we can see that FA-OL and MSSCS achieve the best statistical results, followed by NABC.

5 Conclusion

We proposed FA-OL, whose OL mechanism can effectively expand the current search area, help individuals jump out of local optima, enhance individuals' search capability, and improve the algorithm's global search capability. The OL mechanism and the attraction mechanism of FA act on two individuals participating in comparative fitness, so that the learning capability of the group is enhanced and the optimization capability is improved. The optimization effect of the proposed algorithm was verified based on nine standard test functions, and compared with seven improved algorithms. The optimization results were analyzed by ANOVA. The experimental results show that FA-OL achieves the best optimization performance and robustness.

References

1. Altabeeb, A.M., Mohsen, A.M., Abualigah, L., Ghallab, A.: Solving capacitated vehicle routing problem using cooperative firefly algorithm. *Appl. Soft Comput.* **108**, 107403 (2021). <https://doi.org/10.1016/j.asoc.2021.107403>. <https://www.sciencedirect.com/science/article/pii/S1568494621003264>
2. Cheng, L., Zhong, L., Zhang, X., Xing, J.: A staged adaptive firefly algorithm for UAV charging planning in wireless sensor networks. *Comput. Commun.* **161**, 132–141 (2020). <https://doi.org/10.1016/j.comcom.2020.07.019>. <https://www.sciencedirect.com/science/article/pii/S0140366420307763>

3. Clerc, M., Kennedy, J.: The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evol. Comput.* **6**(1), 58–73 (2002). <https://doi.org/10.1109/4235.985692>
4. Czarnowski, I.: Firefly algorithm for instance selection. *Procedia Comput. Sci.* **192**, 2269–2278 (2021). <https://doi.org/10.1016/j.procs.2021.08.240>. <https://www.sciencedirect.com/science/article/pii/S1877050921017373>
5. Dash, J., Dam, B., Swain, R.: Improved firefly algorithm based optimal design of special signal blocking IIR filters. *Measurement* **149**, 106986 (2020). <https://doi.org/10.1016/j.measurement.2019.106986>. <https://www.sciencedirect.com/science/article/pii/S0263224119308528>
6. Dhal, K.G., Das, A., Ray, S., Gálvez, J.: Randomly attracted rough firefly algorithm for histogram based fuzzy image clustering. *Knowl.-Based Syst.* **216**, 106814 (2021). <https://doi.org/10.1016/j.knosys.2021.106814>. <https://www.sciencedirect.com/science/article/pii/S0950705121000770>
7. Kennedy, J.: Probability and dynamics in the particle swarm. In: *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*, vol. 1, pp. 340–347. <https://doi.org/10.1109/CEC.2004.1330877>
8. Langari, R.K., Sardar, S., Amin Mousavi, S.A., Radfar, R.: Combined fuzzy clustering and firefly algorithm for privacy preserving in social networks. *Expert Syst. Appl.* **141**, 112968 (2020). <https://doi.org/10.1016/j.eswa.2019.112968>. <https://www.sciencedirect.com/science/article/pii/S0957417419306864>
9. Liu, J., Mao, Y., Liu, X., Li, Y.: A dynamic adaptive firefly algorithm with globally orientation. *Math. Comput. Simul.* **174**, 76–101 (2020). <https://doi.org/10.1016/j.matcom.2020.02.020>. <https://www.sciencedirect.com/science/article/pii/S0378475420300598>
10. Mendes, R., Kennedy, J., Neves, J.: The fully informed particle swarm: simpler, maybe better. *IEEE Trans. Evol. Comput.* **8**(3), 204–210 (2004). <https://doi.org/10.1109/TEVC.2004.826074>
11. Nand, R., Sharma, B.N., Chaudhary, K.: Stepping ahead firefly algorithm and hybridization with evolution strategy for global optimization problems. *Appl. Soft Comput.* **109**, 107517 (2021). <https://doi.org/10.1016/j.asoc.2021.107517>. <https://www.sciencedirect.com/science/article/pii/S1568494621004403>
12. Peng, H., Deng, C., Wu, Z.: Best neighbor-guided artificial bee colony algorithm for continuous optimization problems. *Soft Comput.* **23**(18), 8723–8740 (2019). <https://doi.org/10.1007/s00500-018-3473-6>. <https://doi.org/10.1007/s00500-018-3473-6>
13. Peng, H., Zeng, Z., Deng, C., Wu, Z.: Multi-strategy serial cuckoo search algorithm for global optimization. *Knowl.-Based Syst.* **214**, 106729 (2021). <https://doi.org/10.1016/j.knosys.2020.106729>. <https://www.sciencedirect.com/science/article/pii/S0950705120308583>
14. Peng, H., Zhu, W., Deng, C., Wu, Z.: Enhancing firefly algorithm with courtship learning. *Inf. Sci.* **543**, 18–42 (2021). <https://doi.org/10.1016/j.ins.2020.05.111>. <https://www.sciencedirect.com/science/article/pii/S0020025520305363>
15. Qiao, Y., Li, F., Zhang, C., Li, X., Zhou, Z.: Study on the random factor of firefly algorithm. In: Tan, Y., Shi, Y. (eds.) *ICSI 2021*. LNCS, vol. 12689, pp. 58–71. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-78743-1_6
16. Shao, P., Wu, Z., Zhou, X., Deng, C.: Improved particle swarm optimization algorithm based on opposite learning of refraction. *Acta Electron. Sin.* **43**(11), 2137–2144 (2015)

17. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360), pp. 69–73 (1998). <https://doi.org/10.1109/ICEC.1998.699146>
18. rl Tao, R., Meng, Z., Zhou, H.: A self-adaptive strategy based firefly algorithm for constrained engineering design problems. *Appl. Soft Comput.* **107**, 107417 (2021). <https://doi.org/10.1016/j.asoc.2021.107417>. <https://www.sciencedirect.com/science/article/pii/S1568494621003409>
19. Tian, M., Bo, Y., Chen, Z., Wu, P., Yue, C.: Multi-target tracking method based on improved firefly algorithm optimized particle filter. *Neurocomputing* **359**, 438–448 (2019). <https://doi.org/10.1016/j.neucom.2019.06.003>. <https://www.sciencedirect.com/science/article/pii/S0925231219308240>
20. Tian, M., Bo, Y., Chen, Z., Wu, P., Yue, C.: A new improved firefly clustering algorithm for SMC-PHD filter. *Appl. Soft Comput.* **85**, 105840 (2019). <https://doi.org/10.1016/j.asoc.2019.105840>. <https://www.sciencedirect.com/science/article/pii/S1568494619306210>
21. Tizhoosh, H.R.: Opposition-based learning: a new scheme for machine intelligence. In: International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC 2006), vol. 1, pp. 695–701 (2006). <https://doi.org/10.1109/CIMCA.2005.1631345>
22. Trachanatzi, D., Rigakis, M., Marinaki, M., Marinakis, Y.: A firefly algorithm for the environmental prize-collecting vehicle routing problem. *Swarm Evol. Comput.* **57**, 100712 (2020). <https://doi.org/10.1016/j.swevo.2020.100712>. <https://www.sciencedirect.com/science/article/pii/S2210650220303655>
23. Wang, W.c., Xu, L., Chau, K.w., Xu, D.m.: Yin-yang firefly algorithm based on dimensionally Cauchy mutation. *Expert Syst. Appl.* **150**, 113216 (2020). <https://doi.org/10.1016/j.eswa.2020.113216>. <https://www.sciencedirect.com/science/article/pii/S0957417420300427>
24. Wang, Z., Liu, D., Jolfaei, A.: Resource allocation solution for sensor networks using improved chaotic firefly algorithm in IoT environment. *Comput. Commun.* **156**, 91–100 (2020). <https://doi.org/10.1016/j.comcom.2020.03.039>. <https://www.sciencedirect.com/science/article/pii/S0140366420302164>
25. Wu, J., Wang, Y.G., Burrage, K., Tian, Y.C., Lawson, B., Ding, Z.: An improved firefly algorithm for global continuous optimization problems. *Expert Syst. Appl.* **149**, 113340 (2020). <https://doi.org/10.1016/j.eswa.2020.113340>. <https://www.sciencedirect.com/science/article/pii/S0957417420301652>
26. Xie, H., Zhang, L., Lim, C.P., Yu, Y., Liu, C., Liu, H., Walters, J.: Improving k-means clustering with enhanced firefly algorithms. *Appl. Soft Comput.* **84**, 105763 (2019). <https://doi.org/10.1016/j.asoc.2019.105763>. <https://www.sciencedirect.com/science/article/pii/S1568494619305447>
27. Yang, X.-S.: Firefly algorithms for multimodal optimization. In: Watanabe, O., Zeugmann, T. (eds.) SAGA 2009. LNCS, vol. 5792, pp. 169–178. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04944-6_14
28. Yang, X.S.: Firefly algorithm, Levy flights and global optimization. In: Bramer, M., Ellis, R., Petridis, M. (eds.) Research and Development in Intelligent Systems XXVI, pp. 209–218. Springer, London (2010). https://doi.org/10.1007/978-1-84882-983-1_15
29. Zhou, X., Wu, Z., Wang, H.: A differential evolution algorithm using elite opposition-based learning. *J. Chin. Comput. Syst.* **34**(09), 2129–2134 (2013)