Scouting Firefly Algorithm and its Performance on Global Optimization Problems

Jolitte A. Villaruz[1], Bobby D. Gerardo[2], Ariel O. Gamao[3], Ruji P. Medina[4]
Graduate Programs, Technological Institute of the Philippines, Quezon City, Philippines[1, 2, 4]
College of Information and Computing Studies, Northern Iloilo State University, Iloilo City, Philippines[2]
Institute of Computing, Davao del Norte State College, Panabo City, Philippines[3]

***Abstract*—For effective optimization, metaheuristics should maintain the proper balance between exploration and exploitation. However, the standard firefly algorithm (FA) posted some limitations in its exploration process that can eventually lead to premature convergence, affecting its performance and adding uncertainty to the optimization results. To address these constraints, this study introduces an additional novel search mechanism for the standard FA inspired by the behavior of the scout bee in the artificial bee colony (ABC) algorithm, termed the "Scouting FA". Specifically, fireflies stuck in the local optima will take directed extra random walks to escape toward the region of the optimum solution, thus improving convergence accuracy. Empirical findings on the five standard benchmark functions have validated the effects of this modification and revealed that Scouting FA is superior to its original version.**

Keywords—Metaheuristics; firefly algorithm; modified firefly algorithm; global optimization; scout bee; exploitation and exploration

I. Introduction

The firefly algorithm (FA) is a nature-inspired metaheuristics mimicking how fireflies behave. It was introduced by Yang [1] in 2009 to optimize multimodal problems. Over competing algorithms, FA proved advantageous owing to its simplicity, flexibility, ease of implementation, and few parameters to tune. As a result, FA quickly gains popularity in the scientific community. Furthermore, it has been empirically proven to handle NP-hard problems effectively [2]. Since its inception almost 15 years ago, FA and its modified variants have demonstrated significant success in various fields of application. For example, in multilevel image segmentation [3], as a way to reduce the number of dimensions [4], optimizing convolutional neural networks [5], solving course timetabling problems [6], and dealing with complex engineering tasks [7], [8], among other things. Knowing that FA has a universal application makes it a fascinating subject to pursue. In fact, this metaheuristic approach can be investigated further to provide solutions to real-world problems, such as IoT-based applications [9], time-series forecasting [10], [11], and machine vision-based tasks [12].

Metaheuristic optimization algorithms yielded approximations. Even though it does not guarantee the best solution, it gives the best result possible. The two main concepts of metaheuristics are exploration and exploitation [13]. Exploration searches space globally to locate the region with the optimal solution. On the other hand, exploitation seeks the optimal location of convergence by doing a local search within the area discovered by exploration. Because exploration and exploitation are inherently contrasting processes, it is critical to establish a robust exploration mechanism before starting the exploitation process. Likewise, it is also vital to maintain the proper balance between the two techniques during the search. That is why, if exploration is inefficient, the solution may converge too soon because it will become stuck in suboptimal domains before finding the optimal region [14], [15].

Previous studies reveal that FA is relatively robust at exploitation, although its exploratory ability can be improved [16]. Weaknesses in fireflies' exploration ability can eventually impact their convergence accuracy, adding uncertainty to the optimization results [17], [18]. Consequently, this limitation has narrowed the scope of the FA's applications. A recent approach to counter this drawback applied a damped vibration distribution factor to enhance the attractiveness and randomization formula [2]. Another notable solution utilized a quasi-reflection-based learning approach in the initiation stage of implementation to diversify the FA's population [19]. In addition, [20] enhanced the FA's exploration by adding genetic operators and using a dynamically modified step size. A hybrid method [21] used the group search approach established from the social network search (SNS) algorithm [22]. In [23], a novel method for updating the firefly's new location and the ABC [24] algorithm's scout bee search technique was also introduced to supplement the FA's search operation.

After numerous successful deployments of updated and hybridized FA variants, there are still opportunities for improvement. Initiatives to hybridize metaheuristic algorithms, such as those described in [25]–[27] have emerged. Similarly, the exploration mechanism of the FA can be combined with processes from various metaheuristics. Thus, the strengths of each algorithm could be merged to develop new algorithms that are more robust and accurate.

This work is yet another attempt to improve the FA's exploration process. An additional novel search mechanism inspired by the behavior of the scout bee in the ABC algorithm complements the original implementation. Specifically, fireflies stuck in the local optima will take directed extra random walks to escape toward the region of the optimum solution, thus improving convergence accuracy. This method is known as the scouting firefly algorithm (Scouting FA). This study's most significant contribution is providing a better version of the widely used FA that fixes the established shortcomings of the original version.

The remainder of the paper is organized as follows: Section II highlights the FA and ABC algorithms, and related works. Section III suggests a novel way to improve the FA. Section IV describes the experimental setup. The results and discussions are elucidated in Section V. Section VI gives a summary of the conclusions and future works.

II. BACKGROUND AND RELATED WORKS

A. Standard Firefly Algorithm (FA)

The fundamental concept of FA relates to the insects' bioluminescence, which is used to communicate with other fireflies. The following assumptions were made to represent the algorithm mathematically:

1) All fireflies are unisex and attract each other without regard to their sex;

2) Attractiveness is linked to brightness; thus, the firefly with lower intensity will approach the brighter one. Unless there is a brighter firefly, it will move randomly; and

3) The firefly's brightness corresponds to the optimization problem's objective function.

In the FA, convergence accuracy and speed largely depend on two key factors: the variation of light intensity and the formulation of attractiveness [28]. The attractiveness β is relative, depending on how a particular firefly perceives other fireflies. The relativity of β is computed relative to the distance r_{ij} between the firefly i and j. Accordingly, the farther apart the fireflies are, the less light they can see from one another, as governed by the inverse square law. Besides, the fact that light is absorbed by the atmosphere is also a significant factor to consider.

The Cartesian distance r_{ij} between any two fireflies i and j located at x_i and x_j is given by:

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{d}\left(x_{i,k} - x_{j,k}\right)^2} \quad (1)$$

where d is the dimension of the optimization problem.

Correspondingly, the light intensity $I(r)$ varies with the distance r monotonically and exponentially, as depicted in the formula:

$$I = I_0 e^{-\gamma r} \quad (2)$$

where I_0 is the initial light intensity and γ is the light absorption coefficient that controls the light intensity.

As a firefly's attractiveness is proportional to the light intensity perceived by adjacent fireflies, the attractiveness β of a firefly can be derived by:

$$\beta = \beta_0 e^{-\gamma r^2} \quad (3)$$

where β_0 is the attractiveness at $r = 0$.

The formula to determine the movement of a less-bright firefly i that is attracted to brighter firefly j is given by:

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \alpha\varepsilon \quad (4)$$

where the $\beta_0 e^{-\gamma r_{ij}^2}\left(x_j - x_i\right)$ calculates the attraction, while the α is randomization with the vector of random variables ε_I taken from a Gaussian distribution or uniform distribution in range [0, 1].

B. Artificial Bee Colony (ABC) Algorithm

The ABC algorithm is another well-known nature-inspired metaheuristic [24]. It was formally established by Karaboga in 2005 and modeled on how honeybee swarms search for food. Artificial bee colonies are classified as employed, onlookers, and scouts. Employed bees identify food sources within the search space and relay information about the food sources to onlookers via dance moves. The onlookers choose one of the food sources based on its attributes, while the scouts search for new food sources randomly. Initially, all the bees in the colony were scouts.

When the scout finds a new food source using Eq. 5 and starts to consume it, that scout is turned into an employed bee. The abandonment of the food sources will commence once the "limit" given by Eq. (6) is met. Conversely, the employed bee will become a scout once the "trial" exceeds the "limit." The "trial" counter will increase if the scout cannot discover a new food source; otherwise, it will be reset to zero.

$$x_i^j = x_{min}^j + rand(0,1)\left(x_{max}^j - x_{min}^j\right) \quad (5)$$

Assume that the abandoned source is x_i and $j \in \{1, 2, ..., D\}$, then the scout discovers a new food source to be replaced with x_i.

The limit parameter l is computed using:

$$l = (CS * D)/2 \quad (6)$$

where CS is the colony size and D is the dimension of the problem.

C. Related Works

According to the no-free-lunch theorem [29], no single approach can solve all optimization problems. Thus, the original implementations of metaheuristics are modified to perform better. Two outstanding FA versions incorporating the ABC algorithm are discussed below.

A hybrid of firefly and multi-strategy ABC for optimizing single-objective problems is presented in [27]. The FA performs the global search, while the novel multi-strategy ABC does the local search. Nevertheless, this approach displayed computational complexity because two independent search techniques coexisted throughout the search process. Furthermore, switching the search from FA to the multi-strategy ABC employs a diversity measure that raises computational costs.

The work of [23] is another impactful study that enhanced the exploration of FA using the scout bee search mechanism of the ABC algorithm. In their approach, non-improving fireflies will be replaced by new fireflies at random locations within the specified lower (lb_j) and upper bounds (ub_j) based on the formula:

$$x_{i,j} = lb_j + rand * (ub_j - lb_j) \quad (7)$$

where $x_{i,j}$ signifies the *i*th firefly and the corresponding *j*th element; and *rand* is a uniform random number.

The above method provides an entirely random position for fireflies that exhibits no improvements above a specific threshold limit. This can cause fireflies already nearing convergence to spread farther away, necessitating additional exploitation and exploration. This implementation method may impact both the convergence accuracy and the convergence rate.

III. SCOUTING FIREFLY ALGORITHM

The Scouting FA aims to improve the standard FA's exploration ability by allowing the fireflies stuck in the local optimum over a specific threshold limit ("limit") to take directed extra random walks to scout unfamiliar regions in the search space further. The formula for "limit" is identical to that stated in Eq. 6. Afterward, a greedy selection will be utilized. If the new solution has a higher fitness value than the previous one, its position will be updated by Eq. 8; otherwise, no movement will occur.

The directed extra random walk formula is defined as:

$$x_{i+1} = x_i + (\alpha_i - 0.5) * (ub - lb) \quad (8)$$

where x_{i+1} refers to the new position of the firefly after taking a random walk, x_i is the current position, α_i denotes a random number drawn from uniform distribution $U(0, 1)$, while *ub* stands for upper bound and *lb* for the lower bound. $(\alpha_i - 0.5)$ will ensure that a directed extra random walk is provided, such that, if the generated α_i is < 0.50, the firefly will move backward, and if it is > 0.50, a forward movement will be made.

The novelty of Scouting FA over [23] is that instead of providing an entirely random position for fireflies that exhibits no improvements, directed extra random walks with greedy selection is applied. This is to avoid spreading away fireflies that were already nearing convergence. Furthermore, unlike [27], as an alternative for having two independent search techniques that coexisted throughout the search process and the required diversity measure for switching between searches, this study complemented the standard FA as the extra random walks will only be executed when it improves the fitness value even further. The pseudo code of the Scouting FA is described in Algorithm 1.

Algorithm 1: Pseudocode of the Scouting Firefly Algorithm

```
Objective function f(x), x = (x1, ..., xd)^T.
Generate an initial population of n fireflies xi (i = 1, 2, ..., n).
Light intensity Ii at xi is determined by f(xi).
Define FA parameter: α, β, γ.
Set value for the threshold limit value, limit, using Eq. 6
while (t < maxGeneration),
  for i = 1:n (all n fireflies)
    for j = 1:n (all n fireflies) (inner loop)
      if (Ii < Ij)
        Move firefly i towards j using Eq. 4
        Reset triali = 0
      else // solution has no improvement
        triali++
      end if
      Vary attractiveness with distance r via exp[−γr2].
      Evaluate new solutions and update light intensity.
    end for j
  end for i
  for i = 1:n (all n fireflies)
    if (triali >= limit) // no improvement for limit times
      Scout for a new position, P, via random walk using Eq. 8
      if (IP > Ii) // greedy selection
        Move firefly i to position P
      end if
    end if
  end for
  Rank the fireflies and find the current global best g∗.
end while
Postprocess results and visualization.
```

IV. EXPERIMENTAL SET-UP

Five well-known benchmark functions were used to validate the performance of the Scouting FA compared to the original implementation. Table I lists these functions, their formula, variable limits, and global optimum.

TABLE I. BENCHMARK FUNCTIONS USED IN THE EXPERIMENTS

Function	Formula	Limits	Optimum
Sphere	$f_1(x) = \sum_{i=1}^{d} x_i^2$	[-5.12, 5.12]	f(x)=0 at x=(0,0)
Booth	$f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	[-10, 10]	f(x)=0 at x=(1,3)
Easom	$f_3(x) = -cos(x_i)\ cos(x_i)\ exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$	[-100, 100]	f(x)=-1 at x=(π,π)
Rosenbrock	$f(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_1^2)^2 + (x_i - 1)^2]$	[-5, 10]	f(x)=0 at x=(1,1)
Ackley	$f_2(x) = -20 \exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d} cos(cx_i)\right) + a + \exp(1)$	[-5, 5]	f(x)=0 at x=(0,0)

The control parameter values used in the simulations are given in Table II.

TABLE II. CONTROL PARAMETERS OF STANDARD FA AND SCOUTING FA AS IMPLEMENTED IN THE STUDY

Parameter	Value
Maximum generation (*maxGeneration*)	300
Population size (*n*)	30
Randomization parameter (α)	1.0
Attractiveness (β) at $r = 0$	1.0
Light absorption coefficient (γ)	0.97

All experiments were carried out in Python 3.10.5 on an Intel(R) Core (TM) i7-11370H processor running at 3.30 GHz and with 40 GB of random-access memory.

V. RESULTS AND DISCUSSION

Due to the stochastic nature of the metaheuristic algorithms, each iteration is seeded with a random number to ensure that each solution is unique. Experimental results are shown in Table III. The best, worst, and mean optimal fitness value were noted and compared. The results of the mean optimal fitness value were generated in 100 independent runs to eliminate the effect of the stochastic simulation discrepancy [28]. If an algorithm gets the best results for the performance metric, the results are shown in bold and in a slightly bigger font size.

TABLE III. COMPARATIVE ANALYSIS WITH STANDARD FA AND SCOUTING FA IMPLEMENTATIONS FOR THE FIVE BENCHMARKS

Functions	Optimization Method	Best	Worst	Mean Optimal Fitness Value
Sphere	Standard FA	1.84E-05	1.83E+00	6.44E-02
	Scouting FA	**2.27E-06**	**1.25E-02**	**1.96E-03**
Booth	Standard FA	2.65E-04	4.80E+00	4.79E-01
	Scouting FA	**1.65E-06**	**1.58E-01**	**6.23E-03**
Easom	Standard FA	-1.00E+00	-8.02E-05	-7.11E-01
	Scouting FA	**-1.00E+00**	**-9.70E-01**	**-9.98E-01**
Rosenbrock	Standard FA	8.66E-03	5.30E+01	2.48E+00
	Scouting FA	**4.51E-04**	**1.12E+00**	**2.63E-01**
Ackley	Standard FA	1.18E-02	4.42E+00	2.11E+00
	Scouting FA	**2.67E-03**	**4.04E-01**	**2.63E-01**

As shown, the Scouting FA outperformed the standard FA in all test functions. Even if the worst values are considered, the Scouting FA is closer to the global optimum than the standard FA.

Fig. 1 to 5 compare convergence plots in the five benchmark functions between the standard FA and the Scouting FA.

Fig. 1 demonstrates that when optimizing the Sphere function, the solution in the standard FA is trapped in the local optimum close to 0.20 of the fitness value in the early iterations. On the other hand, the Scouting FA performs better because it avoided premature convergence and gradually moved closer to the global optimum, *f(x)=0 at x=(0,0)*, in just more than 50 iterations.

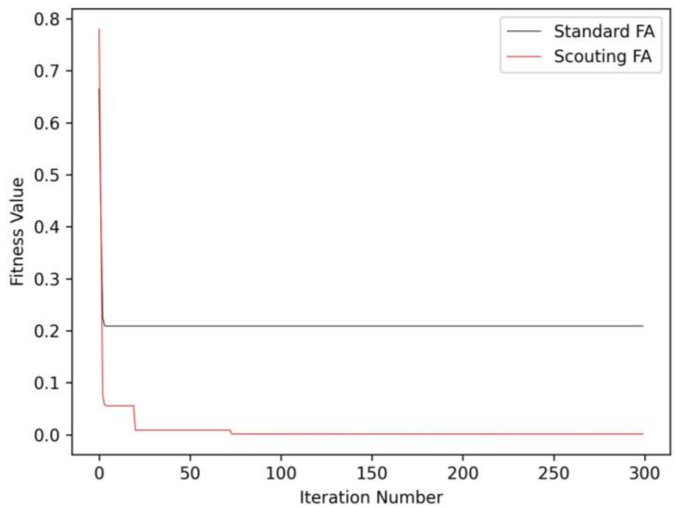


Fig. 1. Convergence plot for the Sphere function.

As shown in Fig. 2, early runs of the standard FA reveal an early convergence nearing 0.20 of the fitness value. When tested using the Booth function, the application of the Scouting FA is more effective. The graph demonstrates the solution progresses towards convergence at the global optimum, *f(x)=0 at x=(1,3)*, in more than 150 iterations.

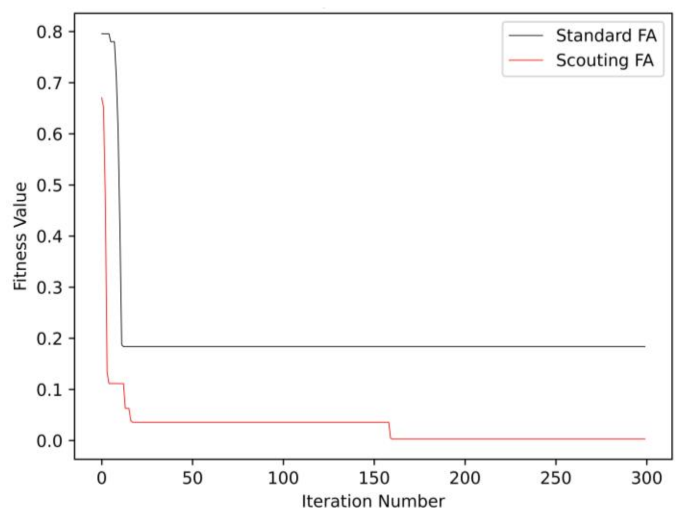


Fig. 2. Convergence plot for the booth function.

As depicted in Fig. 3, in terms of the Easom function, the premature convergence occurs very quickly at nearly -0.80 in the standard FA. Contrary to that, Scouting FA yielded a better result, as escaping from the local optimum is evident. In this problem, the global minimum, *f(x)*=-1 at *x*=(π,π), is nearly achieved at below 50 iterations.

When the Scouting FA optimized the difficult Rosenbrock function, as shown in Fig. 4, there was a significant improvement in achieving a near-optimal result. As illustrated, the Scouting FA's extra random walks slowly improve the solution until it converges close to the global optimum at *f(x)=0 at x=(1,1)* after the 200th iteration.

For a complex optimization problem like the Ackley function, the graph depicted in Fig. 5 revealed that the Scouting FA is considerably better than its original implementation. While standard FA prematurely converges above the 1.5 fitness value, the extra random walks the Scouting FA provides have progressively improved the fitness value until it achieves close to *f(x)=0 at x=(0,0)* before it reaches the 300th iteration.

Fig. 6 shows how 30 fireflies in the five test functions converged after 300 iterations when the standard FA was used. In Fig. 7, the Scouting FA was utilized. As depicted, indeed, with Scouting FA, fireflies can escape being trapped in the sub-optimal solution by taking directed extra random walks.

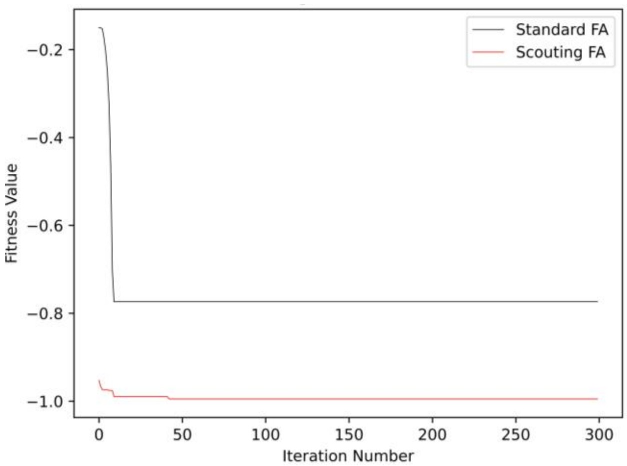


Fig. 3. Convergence plot for the Easom function.

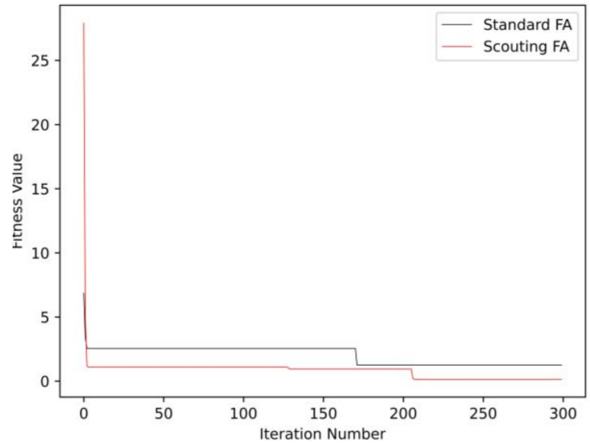


Fig. 4. Convergence plot for the rosenbrock function.

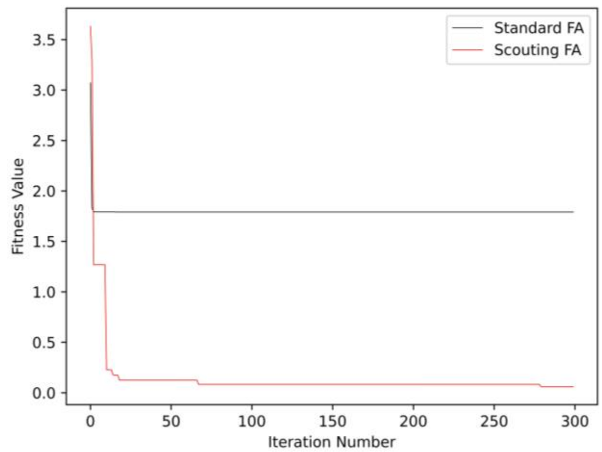
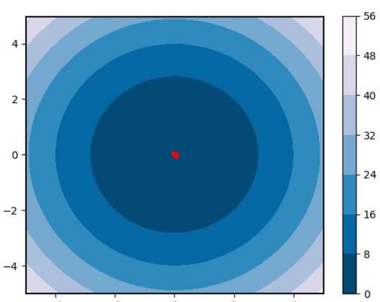
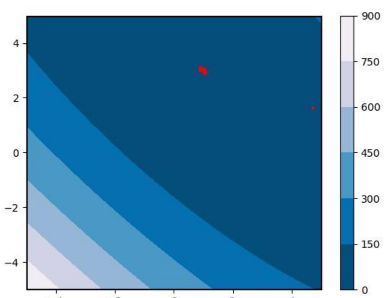


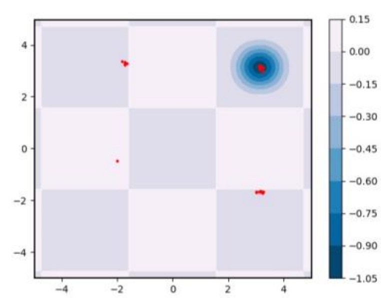
Fig. 5. Convergence plot for the ackley function.



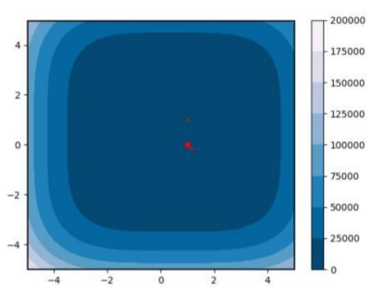
(a) Sphere



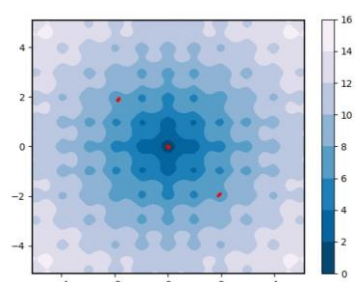
(b) Booth



(c) Easom

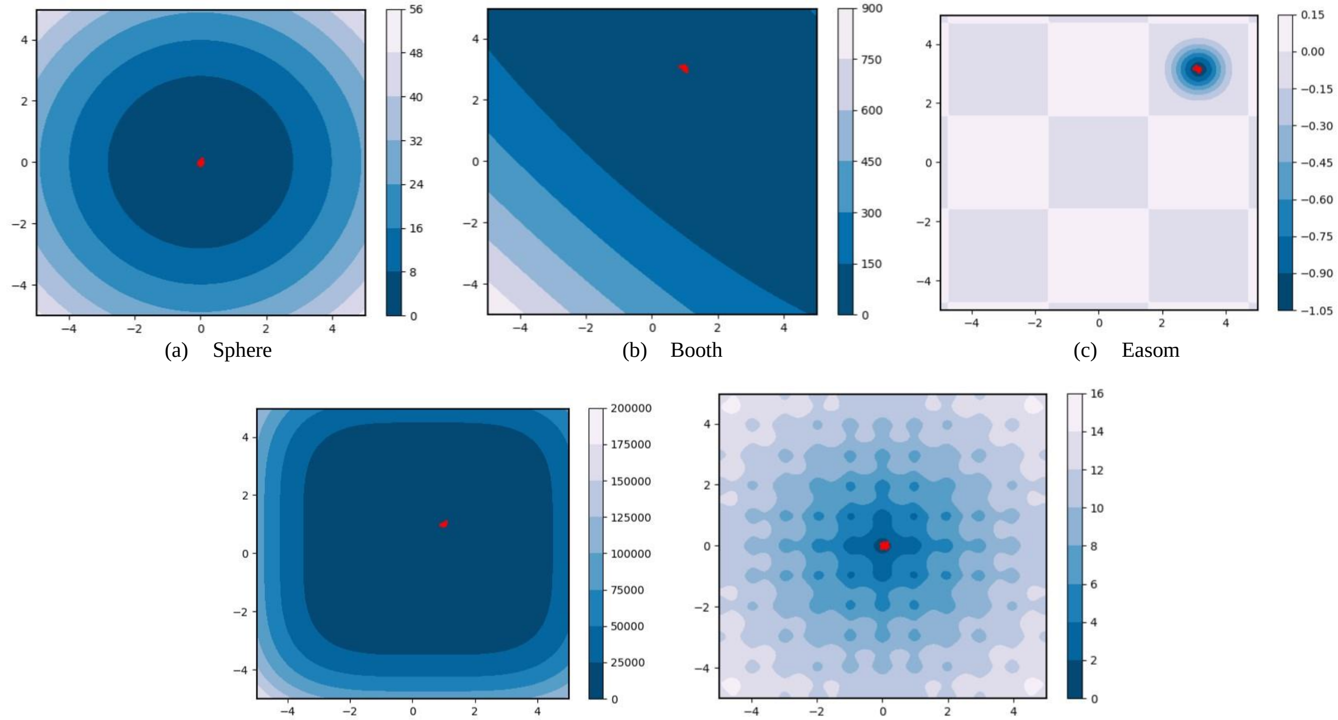


(d) Rosenbrock



(e) Ackley

Fig. 6. The convergence of 30 fireflies in the five test functions after 300 iterations by implementing the standard FA.



(a) Sphere (b) Booth (c) Easom

(d) Rosenbrock (e) Ackley

Fig. 7. The convergence of 30 fireflies in the five test functions after 300 iterations by implementing the Scouting FA.

VI. Conclusions and Future Works

This study improved the exploration of the standard FA by adding the behavior of the scout bee from the ABC algorithm. When the novel search method was used to solve global optimization problems, it made the search more precise, significantly improving convergence accuracy. This result implies that the Scouting FA is more powerful than its original implementation.

In the future, researchers will investigate how well the Scouting FA works when used to tune the hyperparameters of machine learning-based methods for solving real-world optimization problems.

Acknowledgment

The authors acknowledge Aklan State University for the scholarship grant that made this work possible. The recommendations and expertise shared by Mr. Kirk Hilario are also greatly appreciated.

References

[1] X.-S. Yang, “Firefly algorithms for multimodal optimization,” in *International symposium on stochastic algorithms*, 2009, pp. 169–178.

[2] J. Liu, J. Shi, F. Hao, M. Dai, and X. Zhang, “A novel enhanced exploration firefly algorithm for global continuous optimization problems,” *Eng Comput*, vol. 38, pp. 4479–4500, Dec. 2022, doi: 10.1007/s00366-021-01477-6.

[3] A. Sharma, R. Chaturvedi, and A. Bhargava, “A novel opposition based improved firefly algorithm for multilevel image segmentation,” *Multimed Tools Appl*, vol. 81, no. 11, pp. 15521–15544, 2022.

[4] A. O. Gamao and B. D. Gerardo, “Prediction-based model for student dropouts using modified mutated firefly algorithm,” *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 6, pp. 3461–3469, Nov. 2019, doi: 10.30534/ijatcse/2019/122862019.

[5] J. Wang, Y. Liu, and H. Feng, “IFACNN: efficient DDoS attack detection based on improved firefly algorithm to optimize convolutional neural networks,” *Mathematical Biosciences and Engineering*, vol. 19, no. 2, pp. 1280–1303, 2022.

[6] T. Thepphakorn and P. Pongcharoen, “Modified and hybridised bi-objective firefly algorithms for university course scheduling,” *Soft comput*, pp. 1–38, 2023.

[7] S. Bazi, R. Benzid, Y. Bazi, and M. M. al Rahhal, “A fast firefly algorithm for function optimization: Application to the control of bldc motor,” *Sensors*, vol. 21, no. 16, Aug. 2021, doi: 10.3390/s21165267.

[8] H. Peng *et al.*, “Multi-strategy firefly algorithm with selective ensemble for complex engineering optimization problems,” *Appl Soft Comput*, vol. 120, p. 108634, 2022.

[9] D. B. Balmadrid, J. N. Mallorca, B. D. Gerardo, and R. P. Medina, “IoT-based LED Lighting System with Variable Pulsing Frequency and Dark Periods for Sunflower Microgreens,” *International Journal of Emerging Technology and Advanced Engineering*, vol. 12, no. 8, pp. 136–143, Aug. 2022, doi: 10.46338/ijetae0822_17.

[10] M. J. D. Viñas, B. D. Gerardo, and R. P. Medina, “Forecasting PM2. 5 and PM10 Air Quality Index using Artificial Neural Network,” *Journal of Positive School Psychology*, pp. 6863–6871, 2022.

[11] S. G. Aribe, B. D. Gerardo, and R. P. Medina, “Time Series Forecasting of HIV/AIDS in the Philippines Using Deep Learning: Does COVID-19 Epidemic Matter?,” *International Journal of Emerging Technology and Advanced Engineering*, vol. 12, no. 9, pp. 144–157, Sep. 2022, doi: 10.46338/ijetae0922_15.

[12] D. M. Barrios, R. G. Lumauag, and J. A. Villaruz, “Machine vision-based dried danggit sorter,” in *2019 IEEE 4th International Conference on Computer and Communication Systems, ICCCS 2019*, 2019. doi: 10.1109/CCOMS.2019.8821634.

[13] M. Sababha, M. Zohdy, and M. Kafafy, “The enhanced firefly algorithm based on modified exploitation and exploration mechanism,” *Electronics (Switzerland)*, vol. 7, no. 8, Aug. 2018, doi: 10.3390/electronics7080132.

[14] M. J. Goldanloo and F. S. Gharehchopogh, "A hybrid OBL-based firefly algorithm with symbiotic organisms search algorithm for solving continuous optimization problems," *J Supercomput*, vol. 78, no. 3, pp. 3998–4031, 2022.

[15] [15] K. Rezaei and H. Rezaei, "An improved firefly algorithm for numerical optimization problems and it's application in constrained optimization," *Eng Comput*, vol. 38, no. 4, pp. 3793–3813, 2022.

[16] X.-S. Yang, *Nature-inspired Algorithms and Applied Optimization*, vol. 744. 2018. [Online]. Available: http://link.springer.com/10.1007/978-3-319-67669-2

[17] M.-R. Chen, L.-Q. Yang, G.-Q. Zeng, K.-D. Lu, and Y.-Y. Huang, "IFA-EO: An improved firefly algorithm hybridized with extremal optimization for continuous unconstrained optimization problems," *Soft comput*, pp. 1–22, 2022.

[18] A. Abbaszadeh Shahri, M. Khorsand Zak, and H. Abbaszadeh Shahri, "A modified firefly algorithm applying on multi-objective radial-based function for blasting," *Neural Comput Appl*, pp. 1–17, 2022.

[19] T. Bezdan, D. Cvetnic, L. Gajic, M. Zivkovic, I. Strumberger, and N. Bacanin, "Feature selection by firefly algorithm with improved initialization strategy," in *7th conference on the engineering of computer based systems*, 2021, pp. 1–8.

[20] N. Bacanin, M. Zivkovic, T. Bezdan, K. Venkatachalam, and M. Abouhawwash, "Modified firefly algorithm for workflow scheduling in cloud-edge environment," *Neural Comput Appl*, vol. 34, no. 11, pp. 9043–9068, 2022.

[21] D. Jovanovic, M. Antonijevic, M. Stankovic, M. Zivkovic, M. Tanaskovic, and N. Bacanin, "Tuning Machine Learning Models Using a Group Search Firefly Algorithm for Credit Card Fraud Detection," *Mathematics*, vol. 10, no. 13, Jul. 2022, doi: 10.3390/math10132272.

[22] S. Talatahari, H. Bayzidi, and M. Saraee, "Social network search for global optimization," *IEEE Access*, vol. 9, pp. 92815–92863, 2021.

[23] N. Bacanin, T. Bezdan, K. Venkatachalam, and F. Al-Turjman, "Optimized convolutional neural network by firefly algorithm for magnetic resonance image classification of glioma brain tumor grade," in *Journal of Real-Time Image Processing*, Aug. 2021, vol. 18, no. 4, pp. 1085–1098. doi: 10.1007/s11554-021-01106-x.

[24] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Technical report-tr06, Erciyes university, engineering faculty, computer …, 2005.

[25] L. D. Austero, A. M. Sison, J. B. Matias, and R. P. Medina, "Solving course timetabling problem using Whale Optimization Algorithm," in *2022 8th International Conference on Information Technology Trends (ITT)*, 2022, pp. 160–166.

[26] J. C. M. Bustillo, R. P. Medina, A. M. Sison, and M. Y. Orong, "Predictive Hybridization Model integrating Modified Genetic Algorithm (MGA) and C4.5," in *2022 6th International Conference on Electronics, Communication and Aerospace Technology*, 2022, pp. 1500–1507. doi: 10.1109/ICECA55336.2022.10009532.

[27] I. Brajević, P. S. Stanimirović, S. Li, and X. Cao, "A hybrid firefly and multi-strategy artificial bee colony algorithm," *International Journal of Computational Intelligence Systems*, vol. 13, no. 1, pp. 810–821, 2020, doi: 10.2991/ijcis.d.200612.001.

[28] X.-S. Yang, "Firefly algorithm, stochastic test functions and design optimisation," *arXiv preprint arXiv:1003.1409*, 2010.

[29] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE transactions on evolutionary computation*, vol. 1, no. 1, pp. 67–82, 1997.