

Improvement of Firefly Algorithm using Particle Swarm Optimization and Gravitational Search Algorithm

Mahdi Tourani*

Technical faculty of Ferdows, University of Birjand, Iran
tourani.mahdi@birjand.ac.ir

Received: 11/Oct/2020

Revised: 27/Mar/2021

Accepted: 11/Apr/2021

Abstract

Evolutionary algorithms are among the most powerful algorithms for optimization, Firefly algorithm (FA) is one of them that inspired by nature. It is an easily implementable, robust, simple and flexible technique. On the other hand, Integration of this algorithm with other algorithms, can be improved the performance of FA. Particle Swarm Optimization (PSO) and Gravitational Search Algorithm (GSA) are suitable and effective for integration with FA. Some method and operation in GSA and PSO can help to FA for fast and smart searching. In one version of the Gravitational Search Algorithm (GSA), selecting the K-best particles with bigger mass, and examining its effect on other masses has a great help for achieving the faster and more accurate in optimal answer. As well as, in Particle Swarm Optimization (PSO), the candidate answers for solving optimization problem, are guided by local best position and global best position to achieving optimal answer. These operators and their combination with the firefly algorithm (FA) can improve the performance of the search algorithm. This paper intends to provide models for improvement firefly algorithm using GSA and PSO operation. For this purpose, 5 scenarios are defined and then, their models are simulated using MATLAB software. Finally, by reviewing the results, It is shown that the performance of introduced models are better than the standard firefly algorithm.

Keywords:K-best Attractive Firefly; Global and Local Best Position; Gravitational Search Algorithm (GSA); Improved Firefly Algorithm (IFA); Movement in Algorithm; Particle Swarm Optimization.

1- Introduction

Since most real-life problems can be modeled as optimization tasks, many methods and techniques that could tackle such problems were developed. Thus, the optimization became one of the most applicable fields in mathematics and computer science. The difficulty of an optimization problem depends on the mathematical relationships between the objective function, potential constraints, and decision variables. Hard optimization problems can be combinatorial (discrete) or continuous (global optimization), while continuous problems can be further be classified as constrained or unconstrained (bound constrained) [1].

Today, different algorithms have been proposed to solve optimization problems, one of them is Evolutionary Algorithms (EA). These algorithms are inspired by nature and offer different solutions to search in the problem space. EAs are stochastic optimization methods based on the evolution theory. They handle a population of candidate solutions (offspring) that evolves according to the principles of natural selection; that is, using selection, recombination, and mutation processes. During the

evolution, individuals compete, and the fittest among them mate for creating the offspring population.

EAs are able to locate the global optimum and are widely used in engineering optimization problems because they may accommodate any ready-to-use evaluation software. However, EAs call for a great number of fitness function evaluations before reaching the global optimum [2].

One of the most popular of EA is the Firefly Algorithm (FA). Fireflies are winged beetles or insects that produce light and blinking at night. The light has no infrared or an ultraviolet frequency which is chemically produced from the lower abdomen is called bioluminescence. They use the flash light especially to attract mates or prey. The flash light also used as a protective warning mechanism to remind the fireflies about the potential predators. Firefly algorithm formulated by Yang [3] is a metaheuristic algorithm that is inspired by the flashing behavior of fireflies and the phenomenon of bioluminescent communication. Reference [3] formulated the firefly algorithm with the following assumptions:

- 1) A firefly will be attracted to each other regardless of their sex because they are unisexual.
- 2) Attractiveness is proportional to their brightness whereas the less bright firefly will be attracted to the

* Corresponding Author

brighter firefly. However, the attractiveness decreased when the distance of the two fireflies increased.

3) If the brightness of both fireflies is the same, the fireflies will move randomly [4].

After [3], authors in [5-11] and some other improved Firefly optimization and contributed to the development of this algorithm. Improvement of Firefly Algorithm (FA) in [5] with Asexual Reproduction Optimization Algorithm (ARO), in [6] with Artificial Bee Colony (ABC) and in reference [7] using Pattern Search (PS) are done. There are also many research on the application of the firefly algorithm to various issues [12-15]. Such as these, Engineering optimization problems can be addressed.

The aims of this paper is improvement the firefly algorithm (FA) performance using other algorithms such as Particle Swarm Optimization (PSO) and Gravitational Search Algorithm (GSA).

In the firefly algorithm, the search is based on current information, but in the PSO, algorithm preserves past data such as local best answer and global best answer to used in new candidate answer. Using this PSO feature and combine it with the firefly algorithm, the behavior of improved FA (Introduced in this paper) will be a function of current and past information.

On the other hand, one of the positive feature of the GSA is the use of more effective information in generating new candidate answer. The main purpose of this, is to reduce the running time of the algorithm, but this method can be avoided confusion in choosing the optimal answer due to the high volume of information too. Therefore, using this GSA feature and combine it with the firefly algorithm, the improved FA (that Introduced in this paper), will use only useful information in problem search processing instead of all the information - effective or ineffective -.

These reasons and other advantages, make the PSO and GSA a good candidate to improve the FA. The high speed and the accuracy of the Improved FA, can be of great help in solving the engineering optimization problems such as Smart Grids, Electric Vehicles and etc.

2- Standard Firefly Algorithm

There are three basic rules in the firefly algorithm that are mentioned in the introductory section. It should be noted that the FA is based on these rules. Initially, a random position of fireflies is produced. The brightness of the fireflies should be associated with the objective function of the related problem.

2-1- The Attractiveness of the Firefly

The attractiveness of firefly i on the firefly j is based on the degree of the brightness of the firefly i and the distance r_{ij} between the firefly i and the firefly j [16] as in Eq. (1).

$$I(r) = \frac{I_s}{r^2} \quad (1)$$

It is supposed that there are n fireflies; and x_i corresponds to the solution for firefly i . The brightness of the firefly i , is associated with the objective function $f(x_i)$. The brightness I of a firefly is chosen to reveal its recent position of its fitness value or objective function $f(x)$ as in Eq. (2).

$$I_i = f(x_i) \quad (2)$$

The less bright (attractive) firefly is attracted and moved to the brighter one. Each firefly has a certain attractiveness value β . However, the attractiveness value β is relative based on the distance between fireflies. The attractiveness function of the firefly is established by Eq. (3).

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

Where β_0 is the firefly attractiveness value at $r=0$ and γ is the media light absorption coefficient [4].

2-2- The Movement Towards Attractive Firefly

It is worth pointing out that the exponent γ_r can be replaced by other functions such as γ_{rm} when $m>0$. The distance between any two fireflies i and j at x_i and x_j can be Cartesian Distance in Eq. (4).

$$r_{ij} = \|x_i - x_j\|_2 = \sqrt{\sum_{k=1}^d (x_{ik} - x_{jk})^2} \quad (4)$$

The firefly i movement is attracted to another more attractive (brighter) firefly j is determined by Eq. (5):

$$x_i^n = x_i^{n-1} + \beta_0 e^{-\gamma r_{ij}^2} (x_j^{n-1} - x_i^{n-1}) + \alpha \epsilon_i^{n-1} \quad (5)$$

Where x_i^n is position of firefly i in iteration n . In this equation, the second term is due to the attraction, while the third term is randomization with the vector of random variable ϵ_i being drawn from a Gaussian Distribution and ($\alpha \in [0,1]$) [5].

The flowchart of the Standard Firefly Algorithm is shown in Fig. 1.

3- Proposed Algorithms

In the standard version of the firefly algorithm, the main role in problem space search is played by the Movement Operator. As this operator is better, the FA is faster and more accurate. In this paper, Gravitational Search Algorithm (GSA) and Particle Swarm Optimization (PSO) are used to improve the performance of Standard FA. These algorithms help to improve Movement Operator in problem space search.

3-1- Improvement of Firefly Algorithm using Gravitational Search Algorithm

The Gravitational Search Algorithm (GSA) is a kind of population based stochastic search algorithm which was first proposed by Rashedi et al. [17], [18].

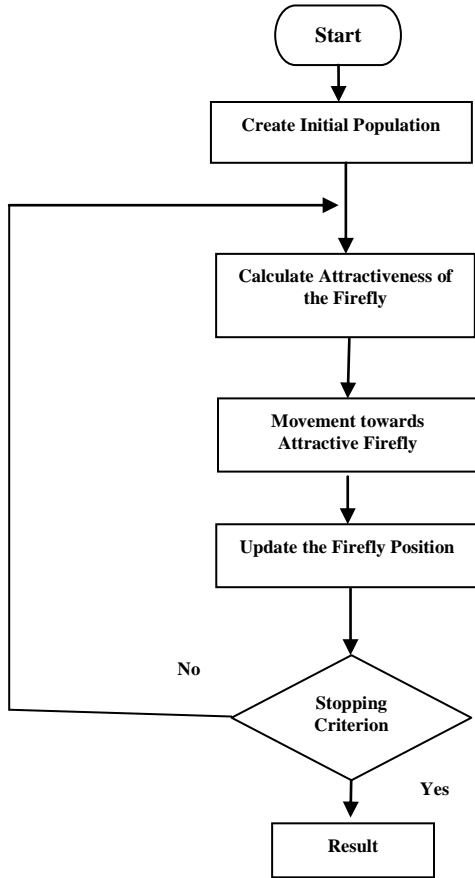


Figure 1. Flowchart of the Standard Firefly Algorithm

The method is based on Newton's theory. Newton's law states that every particle (mass) attracts another particle by means of some gravitational force. Technically, in GSA, each particle has associated with four attributes: particle position, its inertial mass, active gravitational mass, and passive gravitational mass. The particle's position gives the solution of a problem while fitness function is used to calculate the gravitational and inertial masses [19].

In gravitational search algorithm, the movement plays an important role in searching the problem space. The force exerted on the particles by other masses causes this movement. This force is obtained by Eq. (6-10)

$$F_i^n = \sum_{j=1}^m F_{ij}^n \times rand_j \quad (6)$$

$$F_{ij}^n = G^n \frac{M_i^n \times M_j^n}{(r_{ij}^n)^p + \varepsilon} (x_j^n - x_i^n) \quad (7)$$

$$G^n = G_0 e^{\frac{-cn}{Max(n)}} \quad (8)$$

$$M_i^n = \frac{q_i^n}{\sum_{i=1}^m q_i} \quad (9)$$

$$q_i^n = \frac{fit_i^n - worst^n}{best^n - worst^n} \quad (10)$$

where, F_i^n is the force on the i th particle in n th iteration, $rand_j$, random value, F_{ij}^n , the force on the i th particle from j th particle in n th iteration, G_0 , primary gravitational constant, $Max(n)$, maximum of iterations, r_{ij} , the distance between i th and j th particles, fit_i^n , fitness of i th particle, $best^n$ and $worst^n$ are best and worst fitness of n iteration.

By creating force on masses, the particles are accelerated and eventually moved.

$$a_i^n = \frac{F_i^n}{M_i^n} \quad (11)$$

$$V_i^{n+1} = rand_i \times V_i^n + a_i^n \quad (12)$$

$$x_i^{n+1} = x_i^n + V_i^{n+1} \quad (13)$$

where, a_i^n is acceleration of the i th particle in n th iteration, V_i^{n+1} , i th particle velocity in $(n+1)$ th iteration and x_i^n is position of i th particle in n th iteration.

In Eq. (6), calculating the effect of all masses on each particle is take a long time and it also can reduce the accuracy of the optimal response, therefore, in one version of the GSA, K -best particles with bigger mass are used to calculate the force exerted on the particles, rather than calculating the effect of all.

$$F_i^n = \sum_{j=1}^{kbest} F_{ij}^n \times rand_j \quad (14)$$

where, $kbest$ is k particle with bigger fitness in n th iteration.

This can improve the performance of the algorithm. So that, for proposed FA in this paper, it is suggested to be used $Kbest$ for Movement Operator. That way, each firefly is attracted to K -best more attractive instead of all more attractive fireflies. therefore, The Eq. (5) is modified as the Eq. (15).

$$x_i^n = x_i^{n-1} + \beta_0 e^{-\gamma_{ij \in Kbest}^{n2}} (x_{j \in Kbest}^{n-1} - x_i^{n-1}) + ae_i^{n-1} \quad (15)$$

where, $j \in Kbest$ Specify just K -best of more attractive fireflies can be effective in Eq. (15).

It is expected to improve the speed and accuracy of the firefly algorithm with this change in movement towards attractive firefly.

3-2- Improvement of Firefly Algorithm using Particle Swarm Optimization

The Particle Swarm Optimization (PSO) algorithm is a member of the wide category of swarm intelligence methods for solving global optimization problems. It was originally proposed by Kennedy as a simulation of social behavior, and it was initially introduced in 1995 as an optimization method [20,21]. PSO is a population based optimization tool, where the system is initialized with a population of random particles and the algorithm searches for optimal by updating generations [22].

As the GSA, in Particle Swarm Optimization (PSO), the movement generates a new candidate response for problem space. For this movement, the particle velocity is calculated by Eq. (16).

$$V_i^{n+1} = c_1 (x_i^{localbest} - x_i^n) + c_2 (x^{globalbest} - x_i^n) + wV_i^n \quad (16)$$

where, $x_i^{localbest}$ best historical position, $x^{globalbest}$ best position of the entire population, V_i^n i th particle velocity in n th iteration and c_1 , c_2 and w are impact factors for moving to local or global best answer and old particle velocity.

After calculating the velocity, the new position of the particles is obtained by Eq. (17)

$$x_i^{n+1} = x_i^n + V_i^{n+1} \quad (17)$$

In PSO, candidate answers for each iteration calculate according to the distance of its current position from both its own best historical position and the best position of the entire population or its neighborhood. This operation can be defined as movement towards local and global optimum point. it is the key to achieving best response in PSO and can be used to improve the performance of the Firefly algorithm.

$$x_i^n = x_i^{n-1} + c_1\beta_1(x_i^{localbest} - x_i^{n-1}) + c_2\beta_2(x^{globalbest} - x_i^{n-1}) \quad (18)$$

Where $x_i^{localbest}$ best historical position, $x^{globalbest}$ best position of the entire population and c_1 and c_2 are impact factors for moving to local or global best answer.

4- Formulation of Proposed Firefly Algorithm

According to the concepts mentioned in Sections 3-1 and 3-2, the movement operator in Standard FA, Eq. (5), is change to Eq. (19).

$$x_i^n = x_i^{n-1} + c_0\beta_0 e^{-\gamma r_{ij}^{best}} (x_{j \in Kbest}^{n-1} - x_i^{n-1}) + c_1\beta_1 (x_i^{localbest} - x_i^{n-1}) + c_2\beta_2 (x^{globalbest} - x_i^{n-1}) + ae_i^{n-1} \quad (19)$$

Where $x_i^{localbest}$ best historical position, $x^{globalbest}$ best position of the entire population, c_0 impact factors for moving to K-best of more attractive fireflies and c_1 and c_2

are impact factors for moving to local or global best answer.

Therefore, Eq. (19) provides the new definition of movement in Improved Firefly Algorithm. The flowchart of the Proposed Firefly Algorithm is shown in Fig. 2.

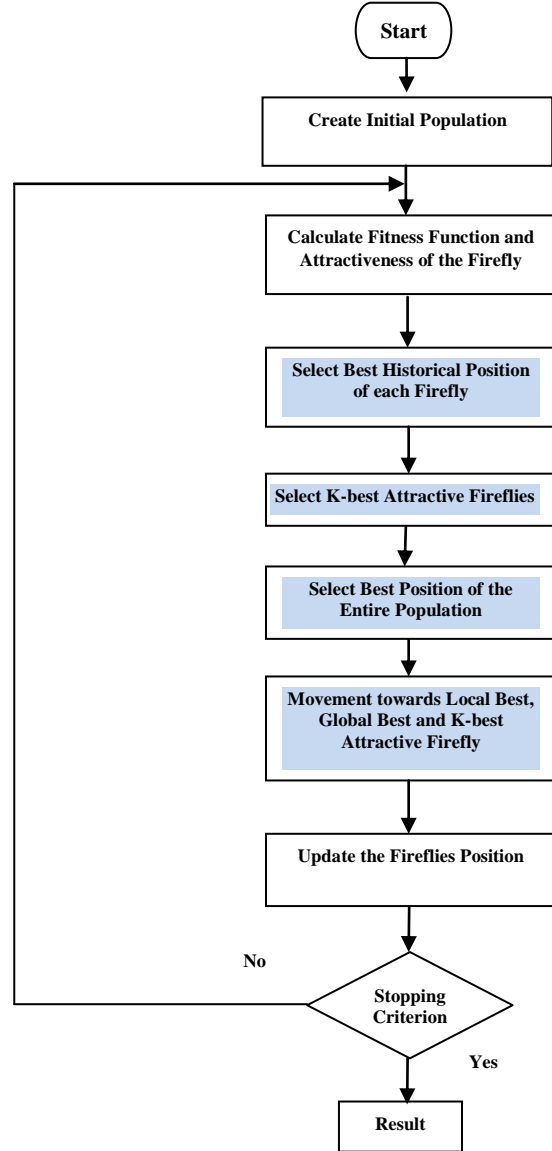


Figure 2. Flowchart of the Proposed Firefly Algorithm

5- Validation and Computational Experiment

To show the effectiveness and the power of Improved firefly algorithm, proposed in this paper, it is evaluated on minimizing 7 well-known benchmark functions [23].

These benchmark functions are presented in table 1. The number of variables in these objective functions is $n=30$.

Table 1. The 7 Benchmark Functions used in experimental study, n=30, [18]

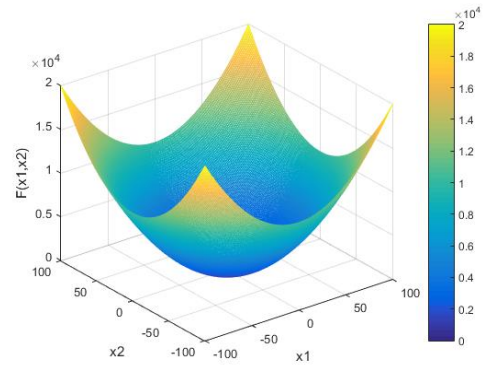
Name Function	Function	Range
Sphere Model	$F_1(x) = \sum_{i=1}^n x_i^2$	[-100,100]
Generalized Rastrigin's Function	$F_2(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	[-5.12,5.12]
Generalized Griewank Function	$F_3(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[-600,600]
Generalized Penalized Functions	$F_4(x) = \frac{\pi}{n} \times \left\{ 10 \sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 (1 + 10 \sin^2(\pi y_{i+1}) + (y_n - 1)) + \sum_{i=1}^n u(x_i, 10, 100, 4) \right.$ $y_i = 1 + \frac{1}{4}(x_i + 1)$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i \text{ f } a \\ 0 & -a \leq x_i \leq a \\ k(-x_i - a)^m & x_i \text{ p } -a \end{cases}$	[-50,50]
Ackley's Function	$f_6(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	[-10,10]
Schwefel's Problem	$f_7(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$	[-32,32]

For the computational test purpose, the simulation executed on PC with Intel Core i5-2410M CPU @ 2.30 GHz processor and 6 GB RAM.

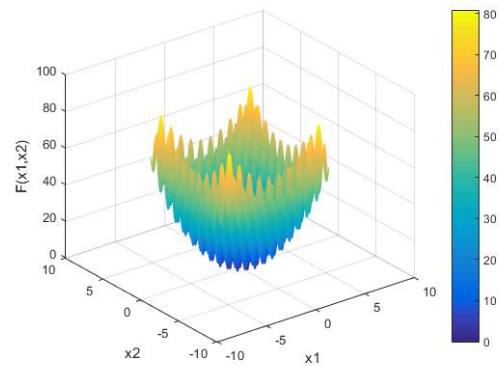
In the functions given on the first column of the table 1, the goal of the algorithm is to minimize the value of the function. On the other hand, this minimization must happen in the shortest possible time. The second column of the table shows the dimensions of the problem and the third column shows its limitations.

The constraints of the problem show the upper and the lower limit of the decision variables. Therefore, the algorithm must find the best optimal answer with these limitations.

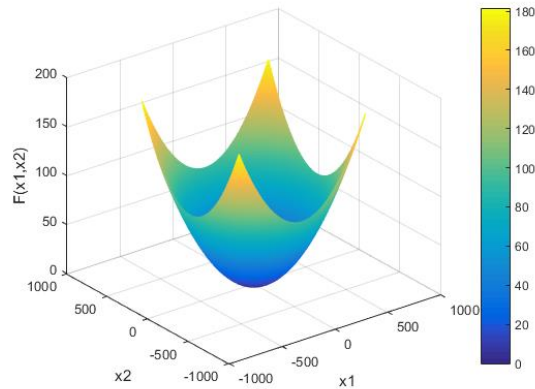
Graphs of three example functions (F1, F2 and F3) in two dimensions are also illustrated in Fig. 3.



A. Graph of F1



B. Graph of F2



C. Graph of F3

Figure 3. Graphs of three example functions (F1, F2 and F3) for n=2

In this section, five scenarios are defined to validate the performance of the Improved Firefly Algorithm obviously.

Scenario I. Standard Firefly Algorithm Simulation (SFA)

Standard FA is simulated in scenario I, initially. With this simulation, the initial state of the optimization is obtained.

In this scenario, Eq. (5) is used to obtain the new position of the firefly.

Scenario II. Applying K-best Attractive Fireflies in the Firefly Algorithm (K-FA)

In this scenario, for movement of firefly, each of them is attracted to K-best more attractive instead of all more attractive fireflies. Eq. (15) gives the new position of the firefly. Total running time of this method is expected to be reduced.

Scenario III. Applying Local Best Position of Each Firefly in the Firefly Algorithm (L-FA)

As mention in section 3-2, local best position of each firefly in movement FA, can be help to optimization process. In this scenario, movement FA is affected by two factors, their local best position and attractive firefly.

Scenario IV. Applying General Best position of Fireflies in the Firefly Algorithm (G-FA)

Similar to the scenario III, but with a difference, the global best position being used instead of local best position.

Scenario V. Applying K-best, local and General Best Position in the Standard Algorithm (IFA)

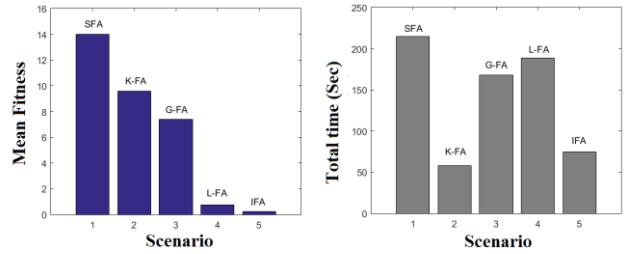
Finally, in scenario V, using results of scenarios II-IV, all parameters (K-best, local and global best position) are contributed to the optimization. This is proposed algorithm and it is expected to improve Firefly algorithm. In this scenario, Eq. (19) is used to obtain the new position of the firefly.

In the Following, scenario 1-5 are implemented on the test functions described in table I. In entire scenarios, population size is set to 40 and all of the fireflies are located in search space randomly. Maximum iteration of algorithm is 200. the result, that shown in tale 2, are achieved by the mean of 30 independent run for each scenario. *Kbest* in scenario II and V (K_FA and IFA) set to 50% of firefly population. Simulation results are shown in table 2 and figure 4. they are compared by Standard Firefly (scenario I, SFA).

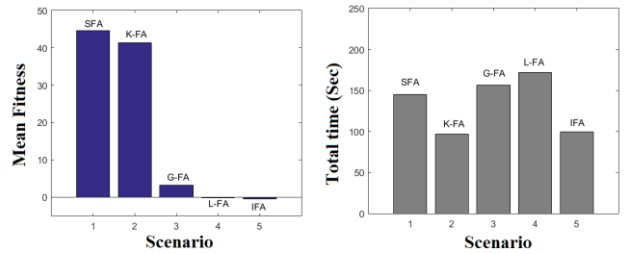
Table 2. The average of final best fitness and total time for 30 runs of minimizing benchmark functions 1, 2,3,4,5,6 and 7, number of iterations=200, n=30

Scenario		SFA	K_FA	G_FA	L-FA	IFA
Min F ₁ (x)	Mean Fitness	3.56	3.03	0.19	0.09	0.04
	Improvement		23.9%	94.6%	97.5%	98.9%
	Total Time	118.78	84.82	132.17	134.89	93.45
	Improvement		28.6%	-11.3%	-13.6%	21.3%
Min F ₂ (x)	Mean Fitness	136.87	149.20	119.8	49.32	47.06
	Improvement		-9%	12.5%	64%	65.6%
	Total Time	127.65	40.24	88.96	135.49	62.27
	Improvement		68.5%	30.3%	-6.1%	51.2%
Min F ₃ (x)	Mean Fitness	0.52	0.52	0.1	0.08	0.05
	Improvement		0%	80.8%	84.6%	90.4%
	Total Time	104.99	60.34	100.60	129.38	84.34
	Improvement		60.3%	4.2%	-23.2%	20%
Min F ₄ (x)	Mean Fitness	14.02	9.85	7.38	0.73	0.22
	Improvement		29.7%	47.4%	94.8%	98.4%
	Total Time	214.54	58.15	167.65	188.37	75.00
	Improvement		72.9%	21.9%	12.2%	65%
Min F ₅ (x)	Mean Fitness	44.56	41.34	3.14	-0.23	-0.5

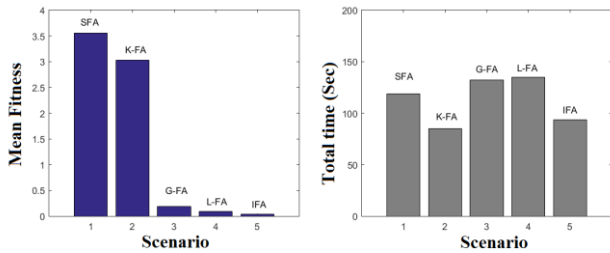
Min $F_6(x)$	Improvement	7.2%	93%	100%	101%	
	Total Time	144.81	96.65	156.73	172.21	99.44
	Improvement	33.3%	-8.2%	-18.9%	31.3%	
	Mean Fitness	35.47	28.26	27.52	4.61	4.81
	Improvement	20.3%	22.4%	87%	86.4%	
Min $F_7(x)$	Mean Fitness	6.96	6.12	1.96	0.80	0.92
	Improvement	12.1%	71.8%	92%	90.8%	
	Total Time	151.86	103.69	100.24	157.55	106.70
	Improvement	31.7%	34%	-3.7%	29.7%	



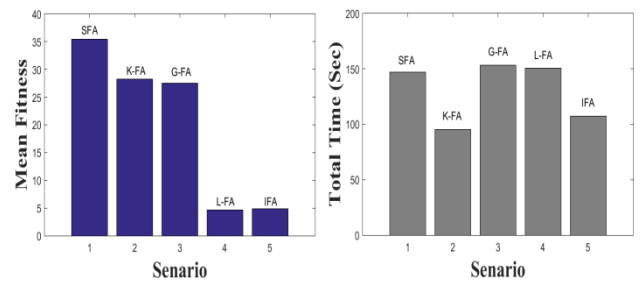
D. Mean Fitness Function and Total Running Time of F4



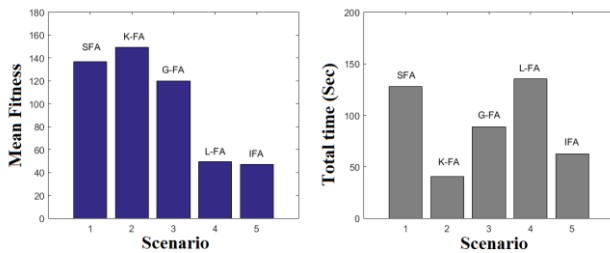
E. Mean Fitness Function and Total Running Time of F5



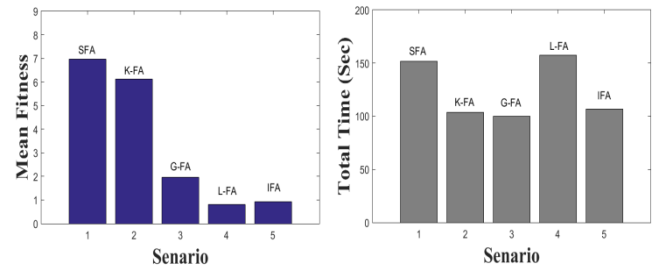
A. Mean Fitness Function and Total Running Time of F1



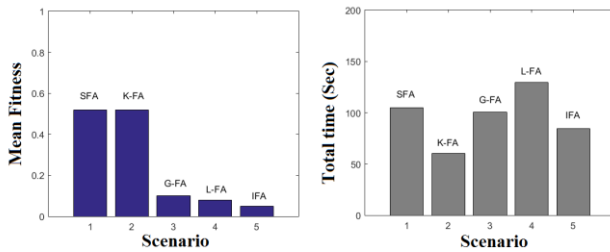
F. Mean Fitness Function and Total Running Time of F6



B. Mean Fitness Function and Total Running Time of F2



G. Mean Fitness Function and Total Running Time of F7



C. Mean Fitness Function and Total Running Time of F3

Figure 4, Mean Fitness Function and Total Running Time of Benchmark Functions

As is clear from the simulation results, performance of all the algorithms introduced in Scenario 2-5, (K_FA, G_FA, L_FA and IFA), are better than the Standard Firefly Algorithm (SFA). In K_FA, The total time of the 30 independent run is significantly reduced compared to the SFA. Of course,

response accuracy is improved in most functions, too. On the other hand, By running the simulation, the response accuracy in G_FA and L_FA have increased and these scenarios have better performance than the SFA. As a result, in scenario 5, which is obtained from the integration of scenarios 2-4, both of the total time and the response accuracy has improved simultaneously compared to SFA. This improvement (IFA) is better than the other scenarios (K_FA, G_FA and L_FA).

6- Conclusion

This paper proposed models for improvement of Firefly algorithm. These models are developed using the Gravitational Search Algorithm (GSA) and Particle Swarm Optimization (PSO). K-best attractive fireflies, local best position of each firefly and general best position of population, are innovation that apply on Standard Firefly Algorithm. These parameters affect to the movement of fireflies and they cause the search process to be smart and fast. After analyzing and simulating the models, it is shown that they perform better than the standard version of the Firefly Algorithm. It is also found that among these models, performance of IFA is better than the other models and Standard Firefly Algorithm in accuracy and response speed.

Reference

- [1]. N. Bacanin, M. Tuba, "Firefly Algorithm for Cardinality Constrained Mean-Variance Portfolio Optimization Problem with Entropy Diversity Constraint", *Scientific World Journal*, Vol. 2014, P.P. 1-16, May 2014.
- [2]. W.W. Hwu, "GPU Computing Gems Jade Edition", Morgan Kaufmann press, 2011.
- [3]. X.S. Yang, "Nature-Inspired Metaheuristic Algorithms", Luniver Press. 2008.
- [4]. N. F. Johari, A. M. Zain, N.H. Mustaffa, A. Udin, "Firefly Algorithm for Optimization Problem", *Applied Mechanics and Materials*, Vol. 421, P.P. 512-517, 2013.
- [5]. A.H. Damia, M. Esnaashari, "Automated Test Data Generation Using a Combination of Firefly Algorithm and Asexual Reproduction Optimization Algorithm", *International Journal of Web Research*, Vol. 3, No. 1, P.P. 19-28, Spring-Summer, 2020.
- [6]. I. Brajević, P.S. Stanimirović, S. Li, X. Cao, "A Hybrid Firefly and Multi-Strategy Artificial Bee Colony Algorithm", *International Journal of Computational Intelligence Systems*, Vol. 13, P.P. 810-821, 2020.
- [7]. F. Wahid1, M. Sultan zia, R.N. Bin Rais, M. AAmir, U. Muneer butt, M. Ali, A. Ahmed, I. Ali khan, O. KHalid, "An Enhanced Firefly Algorithm Using Pattern Search for Solving Optimization Problems", *IEEE Access*, Vol. 8, P.P. 148264-148288, 2020.
- [8]. W. Alomoush, K. Omar, A. Alrosan, Y. M. Alomari, D. Albashish, A. Almomani, "Firefly photinus search algorithm", *Journal of King Saud University – Computer and Information Sciences*, Vol. 32, P.P. 599-607, 2020.
- [9]. SH. Mashhadi Farahani, A.A. Abshouri, B. Nasiri, M.R. Meybodi, "Some Hybrid Models to Improve Firefly Algorithm Performance", *International Journal of Artificial Intelligence*, Vol. 8, P.P. 1-20, 2012.
- [10]. G.G. Wang, L. Guo, H. Duan, H. Wang, "A New Improved Firefly Algorithm for Global Numerical Optimization", *Journal of Computational and Theoretical Nanoscience*, Vol. 11, P.P. 477-485, 2014.
- [11]. F. Wahid, R. Ghazali, L.H. Ismail, "Improved Firefly Algorithm Based on Genetic Algorithm Operators for Energy Efficiency in Smart Buildings", *Arabian Journal for Science and Engineering*, Vol. 44, P.P. 4027-4047, 2019.
- [12]. J. Nayak, B. Naik, P. Dinesh, K. Vakula, P.B. Dash, "Firefly Algorithm in Biomedical and Health Care: Advances, Issues and Challenges", *SN Computer Science*. Vol. 1, P.P. 1-36, 2020.
- [13]. H. Zhang, J Yang, J. Zhang, P. Song, X. Xu, "A Firefly Algorithm Optimization-Based Equivalent Consumption Minimization Strategy for Fuel Cell Hybrid Light Rail Vehicle", *Energies*, Vol. 12, P.P. 1-18, 2019.
- [14]. M. Zile, "Routine Test Analysis in Power Transformers by Using Firefly Algorithm and Computer Program", *IEEE Access*, Vol. 8, P.P. 132033-132040, 2019.
- [15]. S.P. Mishra, P.K. Dash, "Short-term prediction of wind power using a hybrid pseudo-inverse Legendre neural network and adaptive firefly algorithm", *Neural Computing and Applications*, Vol. 31, P.P. 2243-2268, 2019.
- [16]. X.S. Yang, "Firefly Algorithm for Multimodal Optimization", *SAGA 2009, Lecture Notes in Computer Science*, Vol. 5792, P.P. 169-178, 2009.
- [17]. Rashedi, "Gravitational search algorithm," M.Sc. thesis, Dept. Elect. Eng., Shahid Bahonar University of Kerman, Kerman, Iran, 2007.
- [18]. E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, "GSA: A gravitational search algorithm," *Information Science*, Vol. 179, P.P. 2232-2248, June 2009.
- [19]. P. Tharawetcharak, T. Karot, C. Pomsing, "An Improved Gravitational Coefficient Function for Enhancing Gravitational Search Algorithm's Performance", *International Journal of Machine Learning and Computing*, Vol. 9, P.P. 261-266, June 2019.
- [20]. R.C. Eberhart, J. Kennedy, "A new optimizer using particle swarm theory", *Proceedings of the Sixth International Symposium on Micromachine and Human Science*, Nagoya, Japan, P.P. 39-43, 1995.
- [21]. J. Kennedy, R.C. Eberhart, "Particle swarm optimization", in *Proceedings of IEEE International Conference on Neural Networks*, Piscataway, NJ, P.P. 1942-1948, 1995.
- [22]. Y. Jiang, T. Hu, C.C. Huang, X. Wu, "An improved particle swarm optimization algorithm", *Applied Mathematics and Computation*, Vol. 193, P.P. 231-239, 2007.
- [23]. X. Yao, Y. Liu, G. Lin, "Evolutionary Programming Made Faster", *IEEE Transactions on Evolutionary Computation*, Vol. 3, No. 2, P.P. 82-102, 1999.

Mahdi Tourani is an Assistant Professor with the Department of Electrical Engineering, Technical faculty of Ferdows, University of Birjand, Birjand, Iran. He received his B.Sc. in Electrical Engineering from Mashhad Ferdowsi University, Iran, in 2008. He also received his M.Sc. and Ph.D. degrees in Electrical Engineering from University of Birjand, Birjand, in 2011 and 2016, respectively. His areas of interest include Computational Intelligence, Smart Grids, Micro Grids as well as Electrical Vehicles.