*IOSR Journal of Computer Engineering (IOSR-JCE)*
*e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 11, Issue 1 (May. - Jun. 2013), PP 75-78*
*www.iosrjournals.org*

# Firefly Algorithm for Unconstrained Optimization

Adil Hashmi[1], Nishant Goel[2], Shruti Goel[2], Divya Gupta[2]
[1]*(Assistant Professor, Maharaja Surajmal Institute of Technology, New Delhi, India)*
[2]*(Maharaja Surajmal Institute of Technology, New Delhi, India)*

**Abstract :** *Meta-heuristic algorithms prove to be competent in outperforming deterministic algorithms for real-world optimization problems. Firefly algorithm is one such recently developed algorithm inspired by the flashing behavior of fireflies. In this work, a detailed formulation and explanation of the Firefly algorithm implementation is given. Later Firefly algorithm is verified using six unimodal engineering optimization problems reported in the specialized literature.*
***Keywords -*** *Benchmark functions, Firefly Algorithm, Swarm Intelligence, Unconstrained Optimization.*

## I. INTRODUCTION

Swarm Intelligence [1] is a newly discovered research area that studies the behavior of social insects and uses their models to solve real-world problems. Swarm based systems are made up of a population of simple agents that interact with each other and their surroundings whose constituents follow simple and fixed rules. Though the individual particles are ignorant of their nature, the collective behavior leads to an intelligent global behavior.

Optimization problems [2] based on swarm intelligence have features like self-organization, decentralized control, derivative free and easier implementation which lead to an emergent behavior overcoming the limitations of conventional methods. Real world optimization problems belong to a class of hard problems [3] whose main objective is to find the minimum or the maximum of the D dimensional objective function [4] where D represents the number of variables to be optimized.

In this paper we intend to explain the implementation of Firefly algorithm [5] in detail and later find optimal solutions to six standard unconstrained test functions [6] along with the convergence speed of the algorithm. The rest of the paper is organized as follows: Section 2 gives a detailed description of firefly algorithm. Section 3 describes the test functions used in our paper briefly. The experimental results are discussed in Section 4 and Section 5 concludes the paper.

## II. FIREFLY ALGORITHM

The Firefly algorithm was introduced by Dr. Xin She yang at Cambridge University in 2007 which was inspired by the mating or flashing behavior of fireflies. Although the algorithm has many similarities with other swarm based algorithms such as Particle Swarm Optimization [7], Artificial Bee Colony Optimization [8] and Ant Colony Optimization [9], the Firefly algorithm has proved to be much simpler both in concept and implementation.

### 1.1 Flashing Behavior of fireflies

Fireflies or lightning bugs belong to a family of insects that are capable to produce natural light to attract a mate or prey. There are about two thousand firefly species which produce short and rhythmic flashes. These flashes often appear to be in a unique pattern and produce an amazing sight in the tropical areas during summer.

The intensity (I) of flashes decreases as the distance (r) increases and thus most fireflies can communicate only up to several hundred meters. In the implementation of the algorithm, the flashing light is formulated in such a way that it gets associated with the objective function to be optimized.

### 1.2 Concept

In firefly algorithm, there are three idealized rules:

- A firefly will be attracted by other fireflies regardless of their sex
- Attractiveness is proportional to their brightness and decreases as the distance among them increases
- The landscape of the objective function determines the brightness of a firefly [10].

Based on these three rules the pseudo code of the Firefly algorithm can be prepared.

**Pseudo code for FA:**

```
Objective function of f(x), where x=(x1,........,xd)
Generate initial population of fireflies;
Formulate light intensity I;
Define absorption coefficient γ;
While (t<MaxGeneration)
For i = 1 to n (all n fireflies);
For j=1 to n (all n fireflies)
If (Ij > Ii), move firefly i towards j;
end if
Evaluate new solutions and update light intensity;
End for j;
End for i;
Rank the fireflies and find the current best;
End while;
Post process results and visualization;
End procedure;
```

### 1.3 Attractiveness and Light Intensity

In the Firefly algorithm, there are two important issues: the variation of the light intensity and the formulation of the attractiveness. We know, the light intensity varies according to the inverse square law i.e.:

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

Where I(r) is the light intensity at a distance r and $I_s$ is the intensity at the source.

When the medium is given the light intensity can be determined as follows:

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

To avoid the singularity at r=0 in (1), the equations can be approximated in the following Gaussian form:

$$I(r) = I_0 e^{-\gamma r^2} \quad (3)$$

As we know, that a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies and thus the attractiveness β of a firefly is determined by equation (4) where $\beta_0$ is the attractiveness at r=0.

$$\beta = \beta_0 e^{-\gamma r^m} \qquad (m \geq 1) \quad (4)$$

### 1.4 Distance

The distance between any two fireflies i and j at $x_i$ and $x_j$ respectively, the Cartesian distance is determined by equation (5) where $x_{i,k}$ is the kth component of the spatial coordinate $x_i$ of the ith firefly and d is the number of dimensions.

$$r_{ij} = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2} \quad (5)$$

### 1.5 Movement

The movement of a firefly i is attracted to another more attractive (brighter) firefly j is determined by

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \epsilon \quad (6)$$

Where the second term is due to the attraction while the third term is randomization with α being the randomization parameter and ϵ being the vector of random numbers drawn from a Gaussian distribution.

It is worth pointing out that (6) is a random walk partial towards the brighter fireflies and becomes a simple random walk if $\beta_0$= 0. The parameter set used in this work is described later.

### 1.6 Convergence

For any large number of fireflies (n), if n>>m, where m is the number of local optima of an optimization problem, the convergence of the algorithm can be achieved. Here, the initial location of n fireflies is distributed uniformly in the entire search space, and as the iterations of the algorithm continue fireflies converge into all the local optimum. By comparing the best solutions among all these optima, the global optima are achieved.

By adjusting parameters γ and α, the Firefly algorithm can outperform both the algorithms Harmony S Search algorithm and PSO. It can also find the global optima as well as the local optima simultaneously and effectively.

## III. UNCONSTRAINED TEST FUNCTIONS

Six well known test functions are used to test the performance of the firefly algorithm. Convergence speed and precision of the algorithm are evaluated through these functions. The functions used in this work are unimodal and are free of constraints. Table 1 briefly describes all the test functions used in the experiment.

TABLE1. Test functions (D-dimension)

| S. No | Range | D | Function | Formulation | F(x) |
|---|---|---|---|---|---|
| 1 | [-10,10] | 10 | Step | $f(x)=\sum_{i=1}^{n}(\lvert x_i+0.5\rvert)$ | 0 |
| 2 | [-2,2] | 10 | Sphere | $f(x)=\sum_{i=1}^{n} ix_i^2$ | 0 |
| 3 | [-10,10] | 10 | Sum square | $f(x)=\sum_{i=1}^{n} ix_i^2$ | 0 |
| 4 | [-100,100] | 10 | Trid10 | $f(x)=\sum_{i=1}^{n}(x_i-1)^2-\sum_{i=2}^{n} x_i x_{i-1}$ | -210 |
| 5 | [-0.5,10] | 10 | Zakharov | $f(x)=\sum_{i=1}^{n} x_i^2+(\sum_{i=1} 0.5ix_i)^2+(i=1\sum_{i=1} 0.5ix_i)^4$ | 0 |
| 6 | [-5,10] | 10 | Rosenbrock | $f(x)=\sum_{i=1}^{n-1}[100\left(x_{i+1}-x_i^2\right)^2+(x_i-1)^2]$ | 0 |

## IV. EXPERIMENTAL ANALYSIS

The firefly algorithm is tested using six standard test functions reported in the literature. The experimental environment is implemented in MATLAB programs and executed on a DELL Studio15 Computer with the configuration of Intel Core I3 CPU M370 at 2.40 GHz and 4GB RAM. In each experimental run, the test functions are processed 10 times for each population size to measure the CPU time taken and to find the best objective function values.

The parameter settings for firefly algorithm are shown in Table 2 whereas the experimental results for objective function values and processing time are shown in Table 3. It must be noted that, since unconstrained test functions were used, only the objective function values were reported and not the solution vectors. The dimension of the algorithm for all the test functions was kept constant equal to 10.

TABLE2. Parameters set for Firefly algorithms

| FIREFLY ALGORITHM |
|---|
| α (randomness): 0.2 |
| γ (absorption): 1.0 |
| β : 0 |
| $\beta_0$ : 0 |
| Population Size : 10,20,40 |

Table 3 gives the best, mean and worst values of the optima obtained by the algorithm along with the elapsed time. As we can see, the firefly algorithm has obtained optimum value for almost all the test functions except the Rosenbrock function. The possible reason for this could be its nature for performing local search and unable to completely get rid of them. It was also observed that as we increased the population size, the firefly algorithm reached a more optimal solution but converged slowly. This is because the computational complexity of firefly algorithm is $O(n^2)$, where n is the population size.

TABLE3. Objective function and elapsed time values for firefly algorithm (N-population size)

| Test functions | N | Firefly Algorithm | | | | | |
|---|---|---|---|---|---|---|---|
| | | Objective Function values | | | Elapsed Time (in seconds) | | |
| | | Best | Worst | Mean | Best | Worst | Mean |
| Sphere function | 10 | $1.233e^{-6}$ | 0.0625 | 0.0020 | 0.226 | 0.456 | 0.355 |
| | 20 | $6.139e^{-7}$ | $1.160e^{-6}$ | 8.0460 | 0.606 | 1.495 | 1.784 |
| | 40 | $5.455e^{-7}$ | $1.177e^{-6}$ | 5.2461 | 1.186 | 5.587 | 6.253 |
| Sum Square Function | 10 | $3.74e^{-6}$ | $1.54e^{-5}$ | $8.23e^{-7}$ | 0.498 | 0.359 | 0.487 |
| | 20 | $5.51e^{-6}$ | $1.98e^{-5}$ | $1.24e^{-5}$ | 0.919 | 0.971 | 1.140 |
| | 40 | $3.83e^{-6}$ | $1.16e^{-5}$ | $0.57e^{-5}$ | 1.827 | 3.071 | 4.187 |
| Step Function | 10 | $4.56e^{-7}$ | $1.22e^{-6}$ | $9.06e^{-7}$ | 0.421 | 0.749 | 0.529 |
| | 20 | $4.67e^{-7}$ | $1.35e^{-6}$ | $9.65e^{-7}$ | 1.490 | 1.817 | 1.482 |
| | 40 | $4.22e^{-7}$ | $1.75e^{-6}$ | $9.37e^{-7}$ | 5.663 | 6.606 | 6.058 |
| Trid10 Function | 10 | -209.9 | -209.9 | -209.9 | 0.452 | 0.648 | 0.525 |
| | 20 | -209.9 | -209.9 | -209.9 | 1.560 | 2.122 | 1.73 |
| | 40 | -209.9 | -209.9 | -209.9 | 5.819 | 5.990 | 6.489 |
| Zakharov Function | 10 | -0.09 | -0.09 | -0.09 | 0.562 | 0.765 | 0.649 |
| | 20 | -0.09 | -0.09 | -0.09 | 1.763 | 1.997 | 1.910 |
| | 40 | -0.09 | -0.09 | -0.09 | 4.946 | 6.817 | 6.370 |
| Rosenbrock Function | 10 | $-9.94e^{+3}$ | $-9.95e^{+3}$ | $-9.94e^{+3}$ | 0.478 | 0.646 | 0.555 |
| | 20 | $-9.96e^{+3}$ | $-9.98e^{+3}$ | $-9.97e^{+3}$ | 1.411 | 1.723 | 1.627 |
| | 40 | $-9.99e^{+3}$ | $-9.99e^{+3}$ | $-9.99e^{+3}$ | 4.940 | 5.898 | 5.218 |

## V. CONCLUSION

In this paper, implementation of recently introduced firefly algorithm was explained in detail. Later its performance was evaluated using six standard test functions on the basis of precision and convergence speed. It was observed that the firefly algorithm gave more accurate results when population size was increased. At the same time, it was also observed that convergence speed of the algorithm towards the optimal solution decreased with increase in population size. Although, the firefly algorithm had advantages of being precise, robust, easy and parallel implementation, it also had disadvantages like slow convergence speed, getting trapped into local optima and no memorizing capability.

## REFERENCES


[1] E. Bonabeau, M. Dorigo, G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems* (Santa Fe Institute Studies in the Sciences of Complexity, NY: Oxford University Press, 1999).

[2] Ausiello, Giorgio; et al. (2003), Complexity and Approximation (Corrected ed.), *Springer, ISBN 978-3-540-65431-5*.

[3] Xin-She Yang and Amir H. Gandomi, Bat Algorithm: A Novel Approach for Global Engineering Optimization, *Engineering Computations, Vol. 29, Issue 5, pp. 464--483 (2012)*.

[4] MA Sasa, Xue Jia, Fang Xingqiao, Liu Dongqing, "Research on Continuous Function Optimization Algorithm Based on Swarm Intelligence", 5th International Conference on Computation, 2009, pg no. 61-65.

[5] X. S. Yang, Firefly Algorithm: Stochastic Test Functions and Design Optimisation, *Int. J. Bio- Inspired Computation, 2010, Vol. 2, No. 2, pp.78–84*.

[6] Oldenhuis, Rody, Many test functions for global optimizers, Mathworks, 1 November 2012.

[7] R. Poli, J. Kennedy, T. Blackwell, *Particle swarm optimization: An overview* (Springer Science and Business Media, LLC 2007).

[8] Pei-Wei TSai, Jeng-Shyang Pan, Bin-Yih Liao, Shu-Chuan Chu, Enhanced Artificial Bee Colony Optimization , *International Journal of Innovative Computing, Information and Control, Volume 5, Number 12, December 2009*.

[9] Yang, X. S. (2008), *Nature-Inspired Metaheuristic Algorithms* ( Luniver Press).

[10] X. S. Yang, Firefly algorithms for multimodal optimization, Stochastic Algorithms: Foundations and Applications (SAGA 2009, Lecture Notes in Computer Sciences, Vol. 5792, 2009)169-178.