

A New Variant of Firefly Algorithm For Global Optimization

¹Gazala Yasmeen, ²Dr. Syed Raziuddin,

¹PG Scholar, ²Professor and Head of Dept, ^{1,2}Dept of CSE

¹oucse66@gmail.com, ²hod_cse@deccancollege.ac.in

^{1,2}DECCAN COLLEGE OF ENGINEERING & TECHNOLOGY Darussalam, Aghapura,
Hyderabad, Telangana –India

Abstract

Firefly algorithm (FFA) is the most recent Swarm Intelligence (SI) Algorithm which is based on flashing of light by fireflies. This algorithm considers each firefly as a possible solution and brightness of each firefly depends on their performance over optimizing problem. The swarm of firefly moves towards the goal by following the brighter firefly and if there is no brighter firefly they will move randomly. The basic FFA algorithm follows a specific update strategy of attraction as well as movement of swarm of fireflies which does not produce good quality results and converge at low rate. Keeping this flaw of FFA a new algorithm is proposed that directs the randomly moving firefly to the brightest firefly in the current iteration. To maintain variety and avoid early convergence directions are given at certain refresh rate. The performance of New variant FFA (NvFFA) is validated against various SI algorithms over standard test cases which can be complex, multimodal and scalable optimization problems whose dimensions can be 10, 30 and 50. The proposed algorithm will prove that quality of solution and convergence rate is improved.

Keywords: Optimization, Metaheuristic, Firefly Algorithm, Swarm Intelligence

1. INTRODUCTION

Optimization is the field which is mostly used in decision making by most of the domains of human life. The solution of many real-life applications has been found with the help of global optimization techniques.

As the decision-making requirements have increased, the complexity of the problems has also increased. Due to this a specialised optimization field called global optimization has been introduced. The algorithm or tools for finding the optimal value (the greatest possible value or the least possible value) of a function

are called optimization techniques. There are various optimization techniques which are used to solve different kind of complex problems.

Optimization methods like simplex programming and derivative based methods does not give accurate decisions on global optimization problems therefore alternative methods like Meta-heuristics and swarm intelligence (SI) algorithms are preferred.

Meta-heuristic algorithms are nature-inspired with multiple interacting agents. They are nature inspired as they are developed by imitating the behaviour of different species in

nature.

A subset of meta-heuristics are referred to as swarm intelligence (SI) based algorithms, and these SI-based algorithms have been developed by mimicking the swarm intelligence characteristics of biological agents such as birds, fish, humans, fireflies and others.

2. RELATED WORK

To solve the complexity which arises in decision making many SI algorithms are used like algorithms based on nectar searching feature of honey bees, algorithms based on egg laying behaviour of cuckoo birds, based on flashing of light by fireflies, based on food searching behaviour of birds and other particles in nature. The hunting of prey by grey wolves is copied or imitated in Grey Wolf Optimizer (GWO). The behaviour of different species are copied for solving global optimization problem.

2.1 Particle Swarm Optimization (PSO)

Particle swarm optimization is a population based stochastic optimization technique proposed by Kennedy and Eberhart. PSO is an SI algorithm developed by mimicking the food searching behaviour of birds. The swarm of birds or school of fish.

Every individual in PSO is called as particle and the position of the particle is considered as possible solution. Each particle has position, velocity and fitness. New velocity is dependent on previous velocity. The location update strategy is dependent on two best solution:- gbest (global best) which is best solution of whole swarm and pbest (personal best) which is

best solution of the particle.

2.2 Artificial Bee Colony (ABC)

Artificial Bee Colony (ABC) is an SI algorithm based on nectar searching and collecting behaviour of real honey bees. There are 3 types of bees Employed bees, Onlooker bees and scouts. Employed bees search for honey and collect it, the onlookers observe them. The employed bee whose food source has been exhausted becomes scout. The richness of the food source collected by employed bees is determined by onlookers by special dance called waggle dance which is performed by employed bees. The position of food source represent possible solution to the optimization problem and nectar among the food source corresponds to quality (fitness) of the solution.

2.3 Cuckoo Search Algorithm (CSA)

Cuckoo Search Algorithm (CSA) is an SI algorithm based on egg laying characteristics of cuckoo species. The cuckoo bird lay their eggs in the other bird's nest as they don't build their own nest. The host bird sees the egg and if they identify the egg then they either throw it away or build new nest. Each egg represents a solution. Each nest has one egg (In simple case). In complicated case each nest has multiple eggs representing set of solutions. Every cuckoo bird is capable of laying one egg at a time and the high quality egg will be carried to next level.

2.4 Firefly algorithm (FFA)

The most recent among all SI algorithm is Firefly Algorithm.

Firefly algorithm (FFA) is one of the evolutionary optimization algorithms, and is inspired by fireflies behaviour in nature. Fireflies are also called as lighting bugs and are one of the most special and fascinating creatures in nature. These luminous insects belong to the beetle family Lampyridae (order Coleoptera), inhabit mainly tropical and temperate regions, and their population is estimated at around 1900 species. They are capable of producing light due to special photogenic organs situated very close to the body surface behind a window of translucent cuticle. Each firefly movement is based on attraction towards the brighter firefly.

This algorithm belongs to stochastic algorithms. This means, it uses a kind of randomization by searching for a set of solutions.

The firefly algorithms depend primarily on the variation of light intensity and the formulation of attractiveness. Each firefly emits light and light emitted can be of any colour yellow, dark yellow, white etc. Main aim is mating. In order to mate it should mate with same specie by blinking light to attract each other. The blinking of light that is rhythm of flashes, rate of flashing and its duration together forms a pattern that attracts both the male and female species towards each other. A firefly is strong if intensity of light is more otherwise it is weak.

The basic motivation behind FFA algorithm is flashing characteristics of fireflies. Most of the basic SI algorithms including FFA, shows

premature convergence and non-feasible solutions on complex multimodal global optimization problems. The FFA algorithm treats the position of every firefly as a possible solution. Every firefly is identified by its brightness and it depends on how well they perform while optimizing a problem. The movement of the firefly depends on the attraction of fireflies to the brighter firefly. The attraction among the fireflies depends on their brightness. The less bright firefly moves close to the brighter one. The attractiveness among swarm is proportional to the brightness and it is the function of distance between them. Thus the fireflies will change their flight depending upon the brighter firefly and if no such firefly is found, they will move randomly.

2.4.1 Rules of Firefly Algorithm

Based on the pattern of flashes Firefly Algorithm was developed by Xin-She Yang at Cambridge University in 2007, which uses three idealized rules:

- All the fireflies are unisex so it means that one firefly is attracted to other fireflies irrespective of their sex.
- Attractiveness and brightness are proportional to each other, so for any two flashing fireflies, the less bright one will move towards the one which is brighter. Attractiveness and brightness both decrease as their distance increases. If there is no one brighter than other firefly, it will move randomly.

- The brightness of a firefly is determined by the view of the objective function which depends upon the optimization problem.

2.4.2 Light Intensity and Attractiveness

In the firefly algorithm, there are two important points: the variation in the light intensity and formulation of the attractiveness. The attractiveness of a firefly is determined by its brightness which in turn is connected with the encoded objective function.

Intensity of light that is quantity of light emitted by firefly obeys inverse square law.

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

Where I is the light intensity and r is the distance between 2 fireflies. Intensity decreases as distance increases.

Light is absorbed by a medium so light intensity of a firefly at a distance r in a medium with fixed light absorption coefficient λ is given as

$$I_r = I_0 e^{-\lambda r} \quad (2)$$

Where I_r is the intensity at a distance r and I_0 is the intensity at a distance $r = 0$.

By combining inverse square law and absorption coefficient light intensity is given as

$$I_r = I_0 e^{-\lambda r^2} \quad (3)$$

$$\text{As } e^{-x} = \frac{1}{1+x} \quad (4)$$

The formula for intensity can be written as

$$I_r = \frac{I_0}{1+\lambda r^2} \quad (5)$$

Attractiveness of a firefly is proportional to light intensity of its adjacent fireflies and is given by the equation

$$A_r = \frac{A_0}{1+\lambda r^2} \quad (6)$$

Where A_r is the attractiveness at distance r and A_0 is the attractiveness at a distance $r = 0$.

The distance between two fireflies i and j is given as Cartesian distance between them with the formula

$$r_{i,j} = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2} \quad (7)$$

Firefly located at a distance x updates its location by moving towards more attractive firefly with the equation

$$x_{\text{new}} = x_{\text{old}} + A_0 e^{\lambda r^2} (x_{\text{cb}} - x_{\text{old}}) + \alpha \epsilon \quad (8)$$

Where x_{new} is the new position of the firefly and it depends on change in x_{old} that is previous position in addition to attractiveness and randomization. Here second term is attractiveness with λ being the light absorption coefficient and x_{cb} is the current best firefly. α is the randomization parameter in the range $[0,1]$ and ϵ is the random number.

Firefly algorithm is stochastic algorithm so search space is randomly searched to find optimal solutions. As search space is vast there is a probability that the firefly converge at a point and get trapped at local minima to bring it out of a specific point it is necessary to give directions so directions are assigned which result in New Variant Firefly Algorithm (NvFFA).

3. New Variant Firefly Algorithm

The New Variant Firefly Algorithm (NvFFA) directs the randomly moving firefly to search around the brightest firefly in the current iteration. The basic FFA algorithm assume

attractiveness among individuals proportional to the brightness. The less bright firefly will get attracted to the brighter firefly. If suitable firefly is not found then random movement is followed which contribute to improper convergence and produce non-feasible solution. In proposed algorithm this random movement is directed, and that firefly moves towards best solution in that iteration (global solution). The initialization strategy is done in such a way that X_{best} is the brightest firefly and may be called as global solution. The location of the randomly moving firefly is changed as $X_{best} \pm r * \Delta x$. The step and adaptive parameter is taken as r .

Pseudo Code for NvFFA Algorithm

- 1: Initialize NF (population size) and d dimension
- 2: Initialize directing factor (r)
- 3: Randomly initialize fireflies $x = x_1, x_2, x_3, \dots, x_d$
- 4: Evaluate fitness function values $f(x)$
- 5: Compute intensity for each solution member
 $I = I_1, I_2, \dots, I_{NF}$
- 6: Find the brightest firefly X_{best}
(which has the best intensity among all the flies)
- 7: while $t \leq \text{MaxIter}$ do
- 8: while $i \leq \text{NF}$ do
- 9: while $j \leq i$ do
- 10: if $I_j > I_i$ then
- 11: $x_i = x_i + A_0 e^{-\lambda r^2} (x_j - x_i) + \alpha \epsilon$
- 12: else
- 13: if $\text{rand} > 0.5$ then
- 14: $x_i = \text{rand}$
- 15: else

- 16: $x_i = X_{best} \pm \text{rand} \times \Delta X$
- 17: end if
- 18: end if
- 19: $j \leftarrow j + 1$, go to step 9.
- 20: Evaluate fitness function values $f_i(x_i)$
- 21: Compute intensity I_i
- 22: Find X_{best} (has the best brightness among all the flies)
- 23: Calculate the attractiveness
- 24: end while
- 25: $i \leftarrow i + 1$, go to step 8.
- 26: end while
- 27: Rank the fireflies and find the current best
- 28: if stop criteria not met then
- 29: $t \leftarrow t + 1$, go to step 7.
- 30: end if
- 31: end while
- Report results
- Terminate

4. Experimental Results

Simulations were carried out by assigning equal size of population to be 20 for all SI algorithms. All algorithms are run 3 times with 1000 iterations. Test functions are used to compare and validate the performance of optimization algorithm. These test functions are complex, multimodal and scalable in dimension so validation can be carried out on different dimensions.

The SI algorithms used here for comparing NvFFA are PSO, CSA, ABC and FFA. The benchmark functions or test functions used here Ackley, Griewank, Powell, Rastrigin, Rosenbrock and zakharov. The stopping criteria

is set to maximum number of iterations.

The dimensions of the problem are set to be 10, 30 and 50 for every dimension the results are recorded and presented in three different ways that is Average result, convergence result and robustness or stability. SI algorithms are stochastic in nature and they generate different results in different run hence may not be consistent.

The average result for 3 trials with 1000 iterations for each dimensions over six test functions which are run on 5 SI algorithms are shown from table I to table III. The average result of algorithm which shows best performance is shown in bold face.

4.1. Average Result with 10 Dimensions

Table 1 represents the average result with 10 dimensions. The average result of best performing algorithm is shown in bold face. Mean results obtained by NvFFA algorithm surpasses all algorithms when tested on functions Akley, Griewank, powell, rastrigin and zakharov. The CSA algorithm shows better result on Rosenbrock.

4.2 Average Result with 30 Dimensions

Table 2 represents the average result with 30 dimensions. The average result of best performing algorithm is shown in bold face. Mean results obtained by NvFFA algorithm surpasses all algorithms when tested on functions Akley, Griewank, powell, rastrigin and zakharov.

4.3 Average Result with 50 Dimensions

Table 3 represents the average result with 50

dimensions. The average result of best performing algorithm is shown in bold face. Mean results obtained by NvFFA algorithm surpasses all algorithms when tested on functions Akley, Griewank, Powell, Rastrigin and Zakharov.

4.4. Robustness

SI algorithms are stochastic in nature and they generate different results in different run hence may not be consistent. The consistency of algorithm is evaluated based on standard deviation in the results obtained. The null standard deviation indicate consistent nature of algorithm. NvFFA algorithm shows almost null value for standard deviation. The consistent algorithm is shown in bold face.

4.4.1 Robustness Result with 10 Dimensions

The consistency results of all the algorithms over six problems for 10 Dimensions shows that NvFFA algorithm shows null value for standard deviation. A small instability is shown on Rosenbrock problem. The robustness result of best performing algorithm is shown in bold face.

4.4.2 Robustness Result with 30 Dimensions

The consistency results of all the algorithms over six problems for 30 Dimensions shows that NvFFA algorithm shows null value for standard deviation. A small instability is shown on Rosenbrock problem. The robustness result of best performing algorithm is shown in bold face.

4.4.3 Robustness Result with 50 Dimensions

The consistency results of all the algorithms

over six problems for 50 Dimensions shows that NvFFA algorithm shows null value for standard deviation. A small instability is shown on Rosenbrock problem. Stability of proposed

algorithm is not affected for problems with higher dimensions. The robustness result of best performing algorithm is shown in bold face.

Problems	ABC	CSA	PSO	FFA	NvFFA
Ackley	5.7516e-001 (7.0711e-001)	6.0517e-008 (6.1819e-008)	4.5498e+000 (9.6973e-002)	7.0385e-004 (1.4206e-004)	8.8818e-016 (0000)
Griewank	2.9137e-002 (1.6203e-002)	1.6863e-002 (8.698e-003)	3.7688e-001 (6.375e-002)	6.5716e-003 (5.6911e-003)	1.0000e-255 (0000)
Powell	2.2202e+000 (3.7259e-002)	2.5098e-012 (1.8029e-012)	7.2684e+001 (6.9917e+001)	2.3909e-003 (8.4079e-004)	1.0000e-255 (0000)
Rastrigin	8.4234e+000 (2.7712e-001)	8.5818e+000 (1.7828e+000)	6.0213e+001 (6.5713e+000)	8.9547e+000 (1.7233e+000)	1.0000e-255 (0000)
Rosenbrock	3.1109e+001 (1.3052e+001)	6.2282e-001 (3.7018e-001)	1.0882e+003 (2.4158e+002)	8.0916e+000 (8.7807e-001)	8.9985e+000 (2.3396e-003)
Zakharov	5.1685e+001 (3.1368e+001)	1.5901e-009 (1.6484e-009)	1.2751e+001 (8.1769e-001)	4.6423e-007 (1.6855e-007)	1.0000e-255 (0000)

Table I: Average and Robustness results obtained with 10 Dimensional problems

Problems	ABC	CSA	PSO	FFA	NvFFA
Ackley	3.7128e+000 (5.3442e-001)	6.4939e-001 (1.0939e+000)	7.7515e+000 (1.4923e-001)	2.3131e-003 (1.4678e-004)	8.8818e-016 (0000)
Griewank	9.1728e-002 (6.7534e-002)	1.1979e-004 (1.8052e-004)	1.0115e+000 (1.7426e-002)	1.1142e-005 (7.2364e-006)	1.0000e-255 (0000)
Powell	1.6123e+003 (6.2461e+002)	1.2081e-001 (1.2688e-001)	6.9455e+003 (4.1035e+002)	4.9080e+000 (4.0757e+000)	1.0000e-255 (0000)
Rastrigin	1.0790e+002 (2.3402e+001)	8.0382e+001 (1.5884e+001)	3.3376e+002 (1.2259e+001)	5.0745e+001 (1.2426e+001)	1.0000e-255 (0000)
Rosenbrock	8.1352e+002 (4.8935e+002)	5.6542e+001 (4.8627e+001)	5.5050e+004 (1.2196e+004)	5.9050e+004 (1.3441e+000)	2.8985e+001 (2.2653e-002)
Zakharov	5.2012e+002 (3.3698e+001)	8.0076e+001 (3.5421e+001)	2.2293e+002 (2.1325e+001)	3.6819e+001 (1.5716e+001)	1.0000e-255 (0000)

Table II. Average and Robustness Results with 30 Dimensional problem

Problems	ABC	CSA	PSO	FFA	NvFFA
Ackley	4.8952e+000 (6.6624e-001)	2.5857e+000 (5.2334e-001)	9.0846e+000 (2.4636e-001)	1.5460e-002 (4.4040e-003)	8.8818e-016 (0000)
Griewank	5.0747e-001 (2.2724e-001)	3.3526e-003 (3.2118e-003)	1.0623e+000 (6.3653e-003)	9.3048e-004 (1.5310e-004)	1.0000e-255 (0000)
Powell	6.9769e+003 (2.5635e+003)	1.0252e+001 (7.7626e+000)	3.0475e+004 (5.7466e+003)	2.8268e+001 (1.7772e+001)	1.0000e-255 (0000)
Rastrigin	2.8015e+002 (3.6356e+001)	1.9725e+002 (3.2796e+001)	7.2789e+002 (2.9205e+001)	6.7752e+001 (1.0549e+001)	1.0000e-255 (0000)
Rosenbrock	6.6672e+003 (5.8560e+003)	1.9429e+002 (3.7913e+001)	4.0278e+005 (1.4431e+004)	9.3211e+001 (3.1824e-001)	4.8999e+001 (2.064e-003)
Zakharov	1.1242e+003 (7.9229e+001)	4.1474e+002 (1.1599e+002)	8.2043e+002 (3.2851e+001)	2.3102e+002 (3.2851e+001)	1.0000e-255 (0000)

Table III. Average and Robustness Results with 50 Dimensional problem

4.3 Convergence

The convergence of algorithms to optimum solutions over iterations are depicted on graphs. The graphs are drawn as semilog on y-axis and normal on x-axis. The graphs are average of 3 trials and 1000 iterations. Convergence means finding global best solution in number of iterations. Algorithm is best if it converges in less number of iterations. The converging graph of each test function for 10 dimensions are shown from Fig 1 to Fig 6.

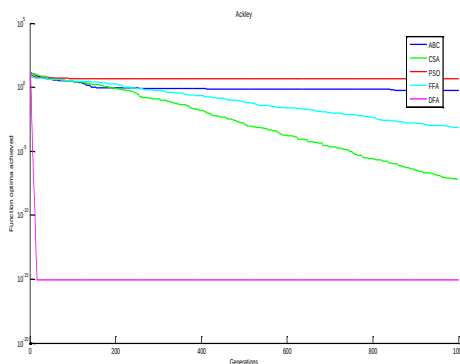


Fig.1. Convergence characteristics on Ackley with 10 Dimensions

The graph shows the convergence characteristics of Ackley function over various SI algorithms. Faster convergence is obtained in NvFFA algorithm. The algorithm takes lesser iterations to produce good quality of solution.

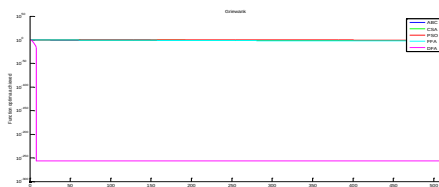


Fig.2. Convergence characteristics on Griewank

with 10 Dimensions

The graph shows the convergence characteristics of Griewank function over various SI algorithms. Faster convergence is obtained in NvFFA algorithm. The algorithm takes lesser iterations to produce good quality of solution.

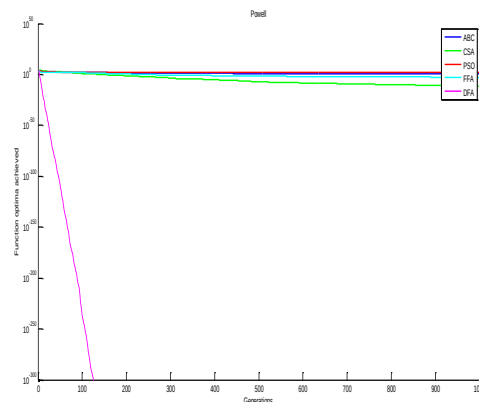


Fig.3. Convergence characteristics on Powell with 10 Dimensions

The graph shows the convergence Characteristics of Powell function over various SI algorithms. Faster convergence is obtained in NvFFA algorithm. The algorithm takes lesser iterations to produce good quality of solution.

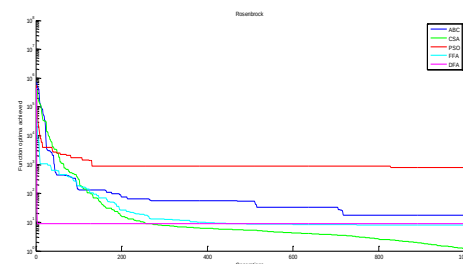


Fig.4. Convergence characteristics on Rosenbrock with 10 Dimensions

The graph shows the convergence characteristics

of Rosenbrock function over various SI algorithms. Faster convergence is obtained in NvFFA algorithm. The algorithm takes lesser iterations to produce good quality of solution.

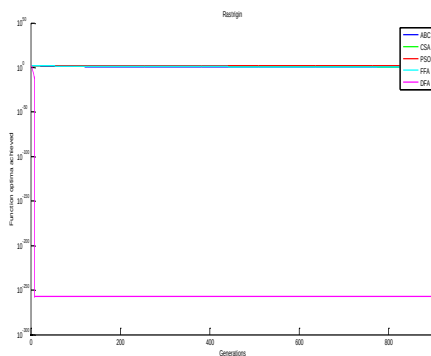


Fig.5. Convergence characteristics on Rastrigin with 10 Dimensions

The graph shows the convergence characteristics of Rastrigin function over various SI algorithms. Faster convergence is obtained in NvFFA algorithm. The algorithm takes lesser iterations to produce good quality of solution.

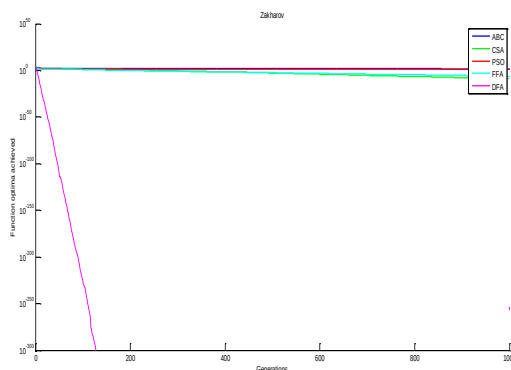


Fig.6. Convergence characteristics on Zakharov with 10 Dimensions

The graph shows the convergence characteristics of Zakharov function over various SI algorithms. Faster convergence is obtained in NvFFA algorithm. The algorithm takes lesser iterations to produce good quality of solution.

5. CONCLUSION

Firefly algorithm was developed with the inspiration of flashing characteristics of fireflies. In this algorithm the less bright firefly moves towards the brighter firefly. If brighter firefly is not found then it will move randomly in search space.

This strategy of random movement leads to computational cost and premature convergence with poor quality solutions. To avoid this flaw New variant Firefly algorithm is proposed.

NvFFA overcomes this problem by identifying firefly with best brightness and directing randomly moving firefly in search of better solution. NvFFA find the brightest firefly in each iteration that is global best and directs the randomly moving firefly towards it.

Simulation results indicate NvFFA has better convergence rate and it produces better quality of solution and stability as compared to other algorithms. NvFFA performs better even if complexity is increased that is problems with higher dimensions and shows null deviation indicating that it is more stable and robust as compared to other SI algorithms.

6. REFERENCES

- [1] Saibal K. Pal, C.S Rai, Amrit Pal Singh, "Comparative Study of Firefly Algorithm and Particle Swarm Optimization for Noisy Non-Linear Optimization Problems" I.J. Intelligent Systems and Applications, 2012, 10, 50-57
- [2] R. Fletcher, Practical Methods of Optimization. John Wiley and Sons, 2000.
- [3] C. L. J. F. Bonnans, J. C. Gilbert and C. A. Sagastizabal, Numerical Optimization: Theoretical and Practical Aspects. Springer, 2003.
- [4] E. G. Talbi, Metaheuristics: From Design to Implementation. John Wiley and Sons, 2009.
- [5] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Computer Engineering Department, Engineering Faculty, Erciyes University, Turkey, Tech. Rep. TR06, 2005.
- [6] D. Karaboga and B. Basturk, "An artificial bee colony (ABC) algorithm for numeric function optimization," in IEEE Swarm Intelligence Symposium, Indianapolis, Indiana, USA, May 2006.
- [7] X.-S. Yang and D. Suash, "Cuckoo search via levy flights," in Proceedings of the World Congress on Nature and Biologically Inspired Computing (NaBIC 2009).
- [8] X.-S. Yang, "Firefly algorithms for multimodal optimization," in Proceedings of the 5th international conference on Stochastic algorithms: foundations and applications, ser. SAGA'09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 169–178.
- [9] R. Eberhart and J. Kennedy, "Particle swarm optimization," in Proceedings of IEEE Int. Conference on Neural Networks, Piscataway, NJ, November 1995, pp. 1114–1121.
- [10] K. Renato, A and S. C. Leandro, dos, "PSO e: Particle swarm with exponential distribution," in Proc. of IEEE Congress on Evolutionary Computation, Sheraton Vancouver Wall Center Hotel, Canada, July 2006, pp. 1428–1433.
- [11] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," Advances in Engineering Software, vol. 69, no. 0, pp. 46 – 61, 2014.
- [12] T. Apostolopoulos and A. Vlachos, "Application of the firefly algorithm for solving the economic emissions load dispatch problem," International Journal of Combinatorics , vol. Article ID 523806, 2011.
- [13] L.-Y. Tseng and C. Chen, "Multiple trajectory search for unconstrained/constrained multi-objective optimization," in CEC'09: Proceedings of the Eleventh conference on Congress on Evolutionary Computation. Piscataway, NJ, USA: IEEE Press, 2009, pp. 1951– 1958.