Volume 7, Issue 2, February 2017 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

# An Improved Approach for Bat Algorithm

[1]Hegazy Zaher, [2]Mahmoud El-Sherbieny, [3]Naglaa Ragaa Saeid, [4]Eman Oun
[1]Dept. of Statistics, [2, 3, 4]Dept. of Operations Research &Management
[1, 2, 3, 4]Institute of Statistical Studies and Researches, Cairo University, Egypt

DOI: 10.23956/ijarcsse/V7I2/01206

***Abstract-*** ***This paper introduces an alternative bat algorithm based on modifying the rules of the standard Bat Algorithm. The proposed algorithm gives better results compared with other known algorithms in the literature. The study considers five of the well known unconstrained benchmarking problems. The results of the proposed modification, standard Bat Algorithm, and other well-known algorithms are compared. Results indicate that proposed version is the best in terms of solution quality.***

***Keywords-*** ***Metaheuristics, Bat algorithm, Real-world problems and Unconstrained problems.***

## I. INTRODUCTION

The ongoing research in solving optimization problems has developed new optimization approaches that achieved advantages over more traditional techniques. The need for getting new optimization techniques stemmed from the fact that the traditional techniques such as mathematical programming approaches have become inefficient for solving real-life applications.

The large scale of systems, numerous design variables and practical requirements of designs in actual applications make the present day problems difficult to be handled using traditional optimization methods [1].

Recently, metaheuristic techniques are used to get a robust solution in solving complex problems. These algorithms tend to produce different solutions even when their initial conditions remain constant at each run because of their random nature. They are preferred for these functions which have several local optima as they can escape from local minima easily in spite of their slow convergence speed [2].

The basic idea behind these techniques is to simulate biological and physical systems in nature, such as natural evolution, immune system, swarm intelligence, annealing process, etc., in a numerical algorithm. These algorithms differ in the way the moves in the search space based on an associated nature-inspired strategy [3].

One of the promising metaheuristic methods, namely, the Bat Algorithm (BA), depends on simulating the echolocation behavior of bats. The capability of echolocation of micro bats is fascinating as these bats can find their prey and discriminate different types of insects even in complete darkness. They achieve this by emitting calls out to the environment and listening to the echoes that bounce back from them. They can identify the location of other objects and instinctively measure how far they are away from them by following delay of the returning sound [4].

From a quick literature review, it is found a set of researches that make a set of modifications or hybridizations on the standard BA to enhance its performance. Iztok F. made hybridization between the differential evolution strategies and the standard BA and named it as a hybrid bat algorithm. It has shown that this algorithm improves significantly the original version of BA [5].

Yilmaz S enhanced exploration and exploitation mechanisms of BA by three modifications, the first one it analyzed the structure of velocity with the inertia weighted update process. The second one it takes into consideration the difference between the solution and the global best solution to get the closest and the farthest dimension of the solution. The third modification is making hybridization between Artificial Bee Colony and BA to improve the exploration capability of BA [6].

Chen Z. removed the velocity parameter and added the inertia weight of location in BA which is determined using normal distribution and then the frequency of the micro bats emitted pulses adjust to the change of random position and optimal position of the micro bats [7].

Amir H. introduced chaos into BA to increase global search mobility for robust global optimization. This method uses chaotic maps to replace random variables which make enhancement when replaced for some random variables of standard BA [8].

Yilmaz S. improved the exploration mechanism by equalizing the loudness and pulse emission rate to the problem dimension by assigning them to each dimension of the solution separately which can perform different capabilities of exploration and exploitation simultaneously [9].

Wasi M. presented an improved self-adaptive BA for the problem of global numerical optimization over continuous domains. It has introduced two improved solution search equations. It has also used a selection probability to control the frequency of employing which leads to a new self-adaptive search mechanism for the Bat algorithm [10].

Ali A. accelerated the search process by invoking the Nelder-Mead method as a local search method in order to refine the best-obtained solution at each iteration [11].

In addition to these modifications, the researchers added some modifications to the standard BA to be able to handle a set of applications to get better results like multi-objective problems, nondeterministic polynomial time (NP) hard problems, K-means clustering problems and other optimization fields.

The structure of this paper is as follows. In Section II, the original bat algorithm is introduced. The proposed modified algorithm is described in Section III. Section IV illustrates experiments and results comparison. At the end of this paper, it's concluded with results discussion and conclusion in section V.

## II. THE STANDARD BAT ALGORITHM

BA depends on a frequency-tuning technique to increase the diversity of the solutions in the population, while at the same; it tries to balance exploration and exploitation during the search process by mimicking the variations of pulse emission rates and loudness of bats when searching for prey using the automatic zooming [12].

### *A. Initialization of Bat Algorithm*

The initial population is generated randomly for n number of bats. Each individual of the population consists of real-valued vectors with d dimensions. The following equation is used to generate the initial population.

$$x_{ij} = x_{min\ j} + rand(0,1)\left(x_{max\ j} - x_{min\ j}\right) \qquad (1)$$

where $i = 1,2, \dots, n; j = 1,2, \dots, d;$ $x_{max\ j}$ and $x_{min\ j}$ are the upper and lower boundaries for dimension j.

### *B. Solution, Frequency & Velocity*

In simulations, it uses virtual bats naturally. It has to define the rules, how their positions $x_i$ and velocities $v_i$ In a d-dimensional search space is updated at each iteration $t$. Among all bats there exists a current best solution $x^*$. The previous rules can be translated to get the new solutions $x_i^t$ and velocities $v_i^t$ at time step $t$ into the updating following equations

$$f_i = f_{min} + (f_{max} - f_{min})\beta, \qquad (2)$$

$$v_i^t = v_i^{t-1} + (x_i^t - x^*)f_i, \qquad (3)$$

$$x_i^t = x_i^{t-1} + v_i^t \qquad (4)$$

Where $\beta \epsilon [0, 1]$ is a random vector drawn from a uniform distribution. $x^*$ is the current global best solution which is located after comparing all the solutions among all the $n$ bats. While $\lambda_i f_i$ is the velocity increment, it uses $f_i$ to adjust the velocity while fixing the other factor $\lambda_i$. The range of $f_i$ differs from a problem to another based on the size domain of the problem of interest. Initially, each bat is randomly assigned a frequency which is derived uniformly from$[f_{min}, f_{max}]$. When a solution is selected from the current best solutions, a new solution for each bat is generated locally using a random walk.

$$x_{new} = x_{old} + \varepsilon A^t, \qquad (5)$$

Where $\varepsilon \epsilon [-1, 1]$ is a random number, while $A^t$ is the average loudness of all the bats at this time step. The update of the velocities and positions of bats has some similarity to the procedure in the standard particle swarm optimization. As $f_i$ controls the range of the movement of the swarming particles, BA can be considered as a balanced combination of the standard particle swarm optimization and the intensive local search controlled by the loudness and pulse rate.

### *C. Loudness and Pulse Updating*

The loudness $A_i$ and the rate $r_i$ of pulse emission have to be updated as the iterations proceed. As the loudness decrease once a bat has found its prey, while the rate of pulse emission increases, the loudness can be chosen as any value of convenience. When the loudness reaches the minimum $A_{min}$, it means that the bat found the prey and stop emitting any sound. The following equations show how the loudness $A_i$ and rate $r_i$ are updated during the iterations.

$$A_i^{t+1} = \alpha A_i^t, \qquad (6)$$

$$r_i^{t+1} = r_i^0[1 - \exp(-\gamma t)], \qquad (7)$$

Where $\alpha$ and $\Upsilon$ are constants. The constant $\alpha$ is similar to the cooling factor in the simulated annealing. For any $0 < \alpha < 1$ and $\Upsilon > 0$ such that

$$A_i^t \to 0, \ \ r_i^t \to r_i^0, \text{as } t \to \infty$$

The choice of parameters requires some experimenting. Each bat should have different values of loudness and pulse emission rate. And this can be achieved by randomization. The loudness and emission rates are updated only if the new solutions are improved that means these bats walk towards the optimal solution. The standard bat algorithm has many advantages; one of them is that it can get quick convergence at initial stages by switching from exploration to exploitation. This makes it an efficient algorithm when a quick solution is needed. In order to improve the performance, many modifications have been added to increase the diversity of the solution and to enhance the performance of the standard Bat algorithm as mentioned previously [12].

**Algorithm 1. Pseudo code of the BA** [7].

1. Objective function: $f(x), x = (x_{1,} \dots, x_d)$
2. Initialize bat population $x_i$ and velocity $v_i, i = 1,2, \dots, n$
3. Define frequency $f_i$ at $x_i$
4. Initialize pulse emission rate $r_i$ and loudness $A_i$

 

5. **While** (t<maximum number of iterations)
6. Generate new solutions by adjusting frequency, and updating velocities and location/solutions.
7. **If** (rand > $r_i$)
8. Select a solution among the best solutions
9. Generate a local solution around the selected best solution
10. **End If**
11. **If** (rand< Ai and f(xi)< f(x*))
12. Accept new solutions
13. Increase $r_i$ reduce $A_i$
14. **End If**
15. Ranks the bats and find current best x*
16. **End While**
17. Display results.

## III. THE PROPOSED BAT ALGORITHM BASED FREQUENCY (PBAF)

Studies show that microbats use the time delay from the emission and detection of the echo, the time difference between their two ears, and the loudness variations of the echoes to build up a three-dimensional scenario of the surrounding. They can detect the distance and orientation of the target, the type of prey, and even the moving speed of the prey such as small insects. For simplicity, it is assumed $f \epsilon [0, fmax]$. It is known that higher frequencies have short wavelengths and travel a shorter distance. For bats, the typical ranges are a few meters. So the frequency is an important effective factor that can affect the global search of the overall algorithm.

In order to improve standard BA, a modification is applied to improve exploration capabilities of BA. The new step of each bat is controlled by the position vector, the global best position, and frequency. The velocity equation is canceled, which minimize the time-consuming. This modification speeds up the convergence of the overall algorithm. Eq 3 is canceled and the equation of bat new step eq 4 is modified as follows

$$x_i^t = f_i \, . \, x_i^{t-1} + (1 - f_i) \, . \, x^* \qquad (8)$$

The effect of this modification could be illustrated by testing it over a set of benchmarking problems as follows.

## IV. EXPERIMENTAL RESULTS

### *A. Benchmark Functions*

In order to verify the proposed modification efficiency of PBAF, the algorithm is tested using 5 minimization benchmark test functions with zero optimal solution in different dimensions as seen in Table I. The values of “Best, worst, mean, median, standard deviation” are shown in Table II.

Table I Benchmark Test Functions

| NO. | Function Name | Dimension | Range | Formulation |
|---|---|---|---|---|
| 1 | Sphere | 10,30,50 | [-5.12,5.12] | $\mathbf{f_1(x) = \sum_{i=1}^{n} x_i^2}$ |
| 2 | Griewangk | 10,30,50 | [-600,600] | $f_2(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} cos(\frac{x_i}{\sqrt{i}}) + 1$ |
| 3 | Ackley Function | 10,30,50 | [-100,100] | $f_3(x) = -20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i))$ |
| 4 | Rastrigin | 10,30,50 | [-15,15] | $f_4(x) = \sum_{i=1}^{d}[x_i^2 - 10\cos(2\pi x_i) + 10]$ |
| 5 | Rosenbrock | 10,30,50 | [-15,15] | $f_5(x)=\sum_{i=1}^{d-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$ |

### *B. Parameter Settings*

Both the PBAF and other algorithms are tested with 30 independent runs for each test function. The population size (no. of Bats) is set to 50. The number of generation is set to 1000, 1500 and 2000 for dimension = 10, 30 and 50 respectively for each run. The minimum frequency value is set to 0 while the maximum value is set to 1. The implementation is done using *Matlab version R2013a.*

### *C. Comparison Between the PBAF, BA and Some Other Algorithms in the Literature*

Comparing the results with BA and previous modifications like Hybrid Bat Algorithm (HBA) [5], Modified Bat Algorithm (MBA )[9], Novel Adaptive Bat Algorithm (NABA) and Bat Algorithm with Self-Adaptive Mutation (BA-SAM ) [10]it's found that the PBAF is superior to others as illustrated in table II.

Table II Comparison Between BA, HBA, MBA, NABA, BA-SAM And PBAF on 5 Standard Benchmark Functions.

| Fun | Function Name | Dimension | Algorithm | Best | Worst | Mean | Median | SD. |
|---|---|---|---|---|---|---|---|---|
| $f_1$ | Sphere | 10 | BA | 1.52 | 1.00e+01 | 4.69 | 4.32 | 2.26 |
| | | | HBA | 4.83e-09 | 2.89e-03 | 1.26e-04 | 5.66e-04 | 1.66e-07 |
| | | | MBA | 3.73e-03 | 1.60e-02 | 8.80e-03 | 7.74e-03 | 3.34e-03 |
| | | | NABA | 1.20e-04 | 1.72 | 2.18e-01 | 7.47e-02 | 4.20e-01 |
| | | | BA-SAM | 1.76e-06 | 3.98e-01 | 9.04e-02 | 4.64e-02 | 1.05e-01 |
| | | | PBAF | 2.04e-13 | 5.80e-11 | **1.59e-11** | 2.29e-12 | 2.55e-11 |
| | | 30 | BA | 1.28e+01 | 3.64e+01 | 2.45e+01 | 2.41e+01 | 5.97 |
| | | | HBA | 3.37e-01 | 9.97e+01 | 3.09 | 7.67e+02 | 1.72e+01 |
| | | | MBA | 1.07e-02 | 1.95e-01 | 4.61e-02 | 3.06e-02 | 4.38e-02 |
| | | | NABA | 1.05e-03 | 2.49e+01 | 2.44 | 9.18e-02 | 5.64 |
| | | | BA-SAM | 5.77e-03 | 3.73 | 2.87e-01 | 6.74e-02 | 6.85e-01 |
| | | | PBAF | 2.11e-13 | 4.65e-10 | **1.07e-10** | 9.79e-12 | 1.75e-10 |
| | | 50 | BA | 2.64e+01 | 6.68e+01 | 4.66e+01 | 4.63e+01 | 1.09e+01 |
| | | | HBA | - | - | - | - | - |
| | | | MBA | 5.87 | 2.30e+01 | 1.08e+01 | 1.02e+01 | 3.70 |
| | | | NABA | 5.51e-02 | 4.57e+01 | 6.41 | 3.55e-01 | 1.22e+01 |
| | | | BA-SAM | 1.36e-01 | 5.87 | 1.03 | 5.63e-01 | 1.24 |
| | | | PBAF | 5.20e-12 | 5.05e-10 | **1.25e-10** | 2.09e-11 | 1.86e-10 |
| $f_2$ | Griewangk | 10 | BA | 6.81 | 4.15e+01 | 1.86e+01 | 1.46e+01 | 9.19 |
| | | | HBA | 2.25e-09 | 3.97e-05 | 3.18e-06 | 8.66e-06 | 1.14e-07 |
| | | | MBA | 2.05 | 2.06e+01 | 8.12 | 6.62 | 5.39 |
| | | | NABA | 2.68 | 5.06e+01 | 1.74e+01 | 1.41e+01 | 1.21e+01 |
| | | | BA-SAM | 1.31e-01 | 1.11e+01 | 2.72 | 1.17 | 2.98 |
| | | | PBAF | 0 | 1.20e-12 | **2.33e-13** | 6.38e-15 | 4.41e-13 |
| | | 30 | BA | 4.61e+01 | 1.75e+02 | 8.89e+01 | 8.77e+01 | 2.46e+01 |
| | | | HBA | 6.38e-06 | 3.57e+01 | 6.42e-05 | 5.99e+01 | 3.12 |
| | | | MBA | 6.36e+01 | 1.82e+02 | 1.10e+02 | 1.01e+02 | 2.83e+01 |
| | | | NABA | 4.67e+01 | 1.70e+02 | 8.56e+01 | 8.21e+01 | 2.63e+01 |
| | | | BA-SAM | 7.18 | 8.68e+01 | 3.38e+01 | 2.67e+01 | 2.17e+01 |
| | | | PBAF | 1.11e-16 | 9.63e-14 | **4.37e-14** | 3.89e-14 | 4.19e-14 |
| | | 50 | BA | 7.35e+01 | 2.59e+02 | 1.61e+02 | 1.63e+02 | 4.18e+01 |
| | | | HBA | - | - | - | - | - |
| | | | MBA | 2.16e+02 | 4.26e+02 | 3.19e+02 | 3.21e+02 | 6.19e+01 |
| | | | NABA | 7.73e+01 | 2.35e+02 | 1.49e+02 | 1.41e+02 | 4.04e+01 |
| | | | BA-SAM | 1.34e+01 | 1.20e+02 | 5.91e+01 | 5.21e+01 | 2.90e+01 |
| | | | PBAF | 0 | 1.55e-13 | **3.87e-14** | 2.89e-14 | 5.07e-14 |
| $f_3$ | Ackley Function | 10 | BA | 1.10e+01 | 1.68e+01 | 1.34e+01 | 1.35e+01 | 1.40 |
| | | | HBA | 6.31e-04 | 2.00e+01 | 1.16e+01 | 9.26 | 1.78e+01 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | MBA | 3.61e-02 | 1.79 | 1.67e-01 | 6.91e-02 | 3.60e-01 |
| | | | NABA | 1.20e-01 | 1.15e+01 | 3.10 | 4.96e-01 | 4.36 |
| | | | BA-SAM | 2.09e-02 | 6.44 | 9.93e-01 | 2.68e-01 | 1.54 |
| | | | PBAF | 7.46e-12 | 1.46e-08 | **3.63e-09** | 1.40e-09 | 4.96e-09 |
| | | 30 | BA | 1.29e+01 | 1.76e+01 | 1.55e+01 | 1.55e+01 | 1.03 |
| | | | HBA | 5.43e-04 | 9.85e+01 | 2.53e-03 | 2.15e+02 | 1.94e+01 |
| | | | MBA | 7.53 | 1.38e+01 | 1.11e+01 | 1.10e+01 | 1.63 |
| | | | NABA | 3.19e-01 | 1.42e+01 | 6.51 | 4.39 | 5.14 |
| | | | BA-SAM | 1.76 | 1.21e+01 | 4.09 | 3.59 | 2.45 |
| | | | PBAF | 1.08e-08 | 4.07e-05 | **1.13e-05** | 7.93e-06 | 1.35e-05 |
| | | 50 | BA | 1.35e+01 | 1.77e+01 | 1.57e+01 | 1.55E+01 | 9.41e-01 |
| | | | HBA | - | - | - | - | - |
| | | | MBA | 1.31e+01 | 1.63e+01 | 1.45e+01 | 1.45e+01 | 8.07e-01 |
| | | | NABA | 3.21 | 1.55e+01 | 1.03e+01 | 1.18e+01 | 4.01 |
| | | | BA-SAM | 3.85 | 1.08e+01 | 6.68 | 6.28 | 1.91 |
| | | | PBAF | 2.71e-07 | 3.22e-05 | **9.23e-06** | 4.51e-06 | 1.15e-05 |
| $f_4$ | Rastrigin | 10 | BA | 8.59e+01 | 1.94e+02 | 1.27e+02 | 1.28e+02 | 2.53e+01 |
| | | | HBA | 5.12 | 2.38e+01 | 1.55e+01 | 4.46 | 1.69e+01 |
| | | | MBA | 1.46e+01 | 3.48e+01 | 2.49e+01 | 2.55e+01 | 4.35 |
| | | | NABA | 1.34e+01 | 5.08e+01 | 2.85e+01 | 2.68e+01 | 9.33 |
| | | | BA-SAM | 6.73 | 3.83e+01 | 2.33e+01 | 2.56e+01 | 8.85 |
| | | | PBAF | 1.98e-11 | 3.03e-08 | **8.08e-09** | 3.74e-09 | 1.09e-08 |
| | | 30 | BA | 4.04e+02 | 6.07e+02 | 4.87e+02 | 4.81e+02 | 5.60e+01 |
| | | | HBA | 1.62 | 3.98e+01 | 1.29e+01 | 1.26e+03 | 5.03 |
| | | | MBA | 2.72e+01 | 2.11e+02 | 1.63e+02 | 1.77e+02 | 4.40e+01 |
| | | | NABA | 8.00e+01 | 3.08e+02 | 1.71e+02 | 1.68e+02 | 6.15e+01 |
| | | | BA-SAM | 8.69e+01 | 2.50e+02 | 1.70e+02 | 1.71e+02 | 4.72e+01 |
| | | | PBAF | 2.27e-13 | 1.93e-06 | **3.12e-07** | 1.63e-09 | 6.82e-07 |
| | | 50 | BA | 7.05e+02 | 1.13e+03 | 8.89e+02 | 8.71e+02 | 9.97e+01 |
| | | | HBA | - | - | - | - | - |
| | | | MBA | 1.85e+02 | 5.58e+02 | 3.84e+02 | 4.00e+02 | 1.21e+02 |
| | | | NABA | 2.29e+02 | 5.20e+02 | 3.70e+02 | 3.71e+02 | 7.82e+01 |
| | | | BA-SAM | 2.02e+02 | 5.49e+02 | 3.68e+02 | 3.74e+02 | 8.47e+01 |
| | | | PBAF | 0 | 8.49e-07 | **1.75e-07** | 1.00e-08 | 3.1e-07 |
| $f_5$ | Rosenbrock | 10 | BA | 1.50e+03 | 1.22e+05 | 4.10e+04 | 2.60e+04 | 3.66e+04 |
| | | | HBA | 6.34e-02 | 5.10e+02 | 6.22e+01 | 1.15e+02 | 7.73 |
| | | | MBA | - | - | - | - | - |
| | | | NABA | 5.87e-01 | 4.89e+04 | 2.55e+03 | 1.31e+01 | 1.00e+04 |
| | | | BA-SAM | 2.72 | 4.70e+02 | 3.24e+01 | 1.03e+01 | 8.75e+01 |
| | | | PBAF | 2.63 | 8.03 | **5.91** | 6.04 | 1.86 |
| | | 30 | BA | 7.38e+04 | 1.38e+06 | 3.78e+05 | 3.47e+05 | 2.56e+05 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | HBA | 8.28 | 2.17e+02 | 6.59e+01 | 4.00e+03 | 2.00e+01 |
| | | | MBA | - | - | - | - | - |
| | | | NABA | 2.79e+01 | 1.88e+05 | 1.23e+04 | 7.73e+01 | 4.00e+04 |
| | | | BA-SAM | 3.88e+01 | 3.24e+04 | 2.86e+03 | 1.97e+02 | 7.86e+03 |
| | | | PBAF | 1.74e-01 | 3.02e+01 | **1.70e+01** | 1.74e+01 | 9.06 |
| | | 50 | BA | 1.96e+05 | 2.06e+06 | 8.14e+05 | 7.14e+05 | 4.48e+05 |
| | | | HBA | - | - | - | - | - |
| | | | MBA | - | - | - | - | - |
| | | | NABA | 2.11e+02 | 4.76e+05 | 4.29e+04 | 9.66e+02 | 1.18e+05 |
| | | | BA-SAM | 3.14e+02 | 1.08e+06 | 4.12e+04 | 1.27e+03 | 1.96e+05 |
| | | | PBAF | 1.89e+00 | 4.39e+01 | **2.08e+01** | 2.02e+01 | 1.23e+01 |

The previous results shows that the proposed algorithm gives better solutions than others in all benchmark problems and in all dimensions.

The confidence of the proposed algorithm is tested using t-test in table III. The results show that the proposed modification is superior to others at a degree of freedom 29 and confidence intervals 95% and 99% where the value of t-tabled at $\alpha = 0.05$ is $1.699$ and at $\alpha = 0.01$ is $2.462$.

Table III T-Test Between Pbaf And Others

| **t-value** | | | | |
|---|---|---|---|---|
| **Function Name** | **Algorithm** | **10 Dimension** | **30 Dimension** | **50 Dimension** |
| Sphere Function | BA | 11.37** | 22.48** | 23.42** |
| | HBA | 41** | 0.98 | - |
| | MBA | 14.43** | 5.76** | 15.99** |
| | NABA | 2.84** | 2.37* | 2.88** |
| | BA-SAM | 4.7** | 2.29* | 4.55** |
| Griewangk Function | BA | 11.09** | 19.79** | - |
| | HBA | 15** | 0.02 | 21.1** |
| | MBA | 8.25** | 21.29** | 28.23** |
| | NABA | 7.88** | 17.83** | 20.2** |
| | BA-SAM | 5** | 8.53** | 11.16** |
| Ackley Function | BA | 52.42** | 82.42** | 91.38** |
| | HBA | 3.57** | 0.007 | - |
| | MBA | 2.54** | 37.3** | 98.41** |
| | NABA | 3.89** | 6.94** | 14.07** |
| | BA-SAM | 3.53** | 9.14** | 19.16** |
| Rastrigin Function | BA | 27.49** | 47.63** | 49.8** |
| | HBA | 5.02** | 14.05** | - |
| | MBA | 31.35** | 20.29** | 17.38** |
| | NABA | 16.3** | 15.23** | 25.91** |
| | BA-SAM | 14.42** | 22.1** | 23.8** |
| Rosenbrock Function | BA | 6.13** | 8.09** | 9.95** |
| | HBA | 38.78** | 12.2** | - |
| | MBA | - | - | - |
| | NABA | 1.39 | 1.68 | 1.99* |
| | BA-SAM | 1.66 | 1.98* | 1.18 |

## V. DISCUSSION AND CONCLUSION

### *A. Discussion*

The results in Table II show that PBAF gets the better solutions of all benchmark functions. It clearly demonstrates that the proposed algorithm outperforms the original BA on all benchmark test functions for all the dimensionalities. It is noted that in the proposed algorithm equation 3 of the standard algorithm is removed, which leads to less consume time. Thus the exploration and convergence characteristics of the proposed algorithm are much better than the standard BA and it also shows that PBAF is better than other proposed modified algorithms in the literature.

### *B. Conclusion*

In this paper, it's introduced a new modification on BA and it's evaluated using a number of benchmark problems on numeric optimization. The proposed modified algorithm has been implemented and tested on several state-of-the-art benchmark optimization problems. The results (i.e., both final solution quality and convergence characteristics) clearly demonstrate that the proposed algorithm is superior on the original BA. Furthermore, it's compared with a set of other proposed modified algorithms and proved its superiority on them.

## REFERENCES


[1] S. S. Rao, *Engineering Optimization*, vol. 56, no. 772801201. 2009.

[2] X. Yang, *Firefly Algorithm*. 2010.

[3] A. L. Parrill, "Introduction to Evolutionary Algorithms," *Evolutionary Algorithms in Molecular Design*, vol. 8, pp. 1–13, 2008.

[4] X.-S. Yang, "A New Metaheuristic Bat-Inspired Algorithm," *Nat. Inspired Coop. Strateg. Optim. (NICSO 2010)*, pp. 65–74, 2010.

[5] I. Fister, D. Fister, and X. S. Yang, "A hybrid bat algorithm," *Elektroteh. Vestnik/Electrotechnical Rev.*, vol. 80, no. 1–2, pp. 1–7, 2013.

[6] S. Yilmaz and E. U. Kucuksille, "Improved Bat Algorithm (IBA) on Continuous Optimization Problems," *Lect. Notes Softw. Eng.*, vol. 1, no. 3, pp. 279–283, 2013.

[7] C. Zhen, Z. Yongquan, and L. U. Mindi, "A simplified Adaptive Bat Algorithm Based on Frequency," *J. Comput. Inf. Syst.*, vol. 9: 16 (201, pp. 6451–6458, 2013.

[8] A. H. Gandomi and X. S. Yang, "Chaotic bat algorithm," *J. Comput. Sci.*, vol. 5, no. 2, pp. 224–232, 2014.

[9] S. Yilmaz, E. U. Kucuksille, and Y. Cengiz, "Modified bat algorithm," *Elektron. ir Elektrotechnika*, vol. 20, no. 2, pp. 71–78, 2014.

[10] M. Wasi, U. Kabir, and M. S. Alam, "Bat Algorithm with Self-adaptive Mutation: A Comparative Study on Numerical Optimization Problems," *Int. J. Comput. Appl.*, vol. 100, no. 10, pp. 975–8887, 2014.

[11] A. Fouad Ali, "Accelerated Bat Algorithm for Solving Integer Programming Problems.," *Egypt. Comput. Sci. J.*, vol. 39, no. 1, p. 25, 2015.

[12] X.-S. Yang and X. He, "Bat algorithm: literature review and applications," *Int. J. Bio-Inspired Comput.*, vol. 5, no. 3, pp. 141–149, 2013.


 