International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-5, January 2020

Extension of Bat Algorithm on Standard Benchmark Functions

Anjali, Deepak Garg, Richa Singh, Sarika Bathija

Abstract: ***Meta heuristics are superior methods of finding, producing and even modifying heuristics that are able to solve various optimization problems. All Meta-heuristic algorithms are influenced by the nature. These types of algorithms tend to mimic the behaviour of biotic components in nature and are emerging as an effective way of solving global optimization algorithms. We have reviewed that no any algorithm is best for all applications due to lack of generality (no. of parameters), non-dynamic input values. So, this paper studied BAT algorithm deeply and found weakness in terms of non-dynamic pulse rate and loudness. In order to avoid being trapped into local optima these inputs are made dynamic with inclusion of levy Flight too. Performance of this proposed Modified BAT approach is evaluated using few standard benchmark functions. For justifying the superiority of Modified BAT, its performance has been compared with standard Bat algorithm too. From simulation it is found that dynamic pulse rate and dynamic loudness improve the performance of Bat algorithm in terms of results without being stuck at local optima and is more general.***

Keywords-Global Optima, Local Optima, Heuristics, Meta-heuristic, NP-Hard, Optimization Algorithm

I. INTRODUCTION

Optimization algorithms are solved by various methods and techniques including the meta-heuristics which more or less provides the optimal solution. Humongous algorithms have been propounded based on the behaviours of various biotic components of the nature [1-10].

The standard Bat algorithm was introduced in 2010 by Xin-She Yang and this algorithm is influenced by nature. It is based on the concept of echolocation produce by bats to avoid obstacles. Loudness and pulse rate of emission is the basic concept to find the distance from the prey or obstacle. It gives robust solution on low dimensional function Standard Bat algorithm has three key features: 1. frequency tuning 2. Automatic Zooming 3. Parameter control.

Remaining part of the paper is organized into following sections. Section 1 includes the introduction of original bat and modified bat algorithm, Section 2 includes various meta-heuristic algorithms, Section 3 includes original bat algorithm, whereas Section 4 includes Modified bat algorithm, Section 5 deals with Implementation of modified bat, Section 6 Comparison of original bat and modified Bat along with various meta-heuristic algorithms on standard benchmark function. Section 7Comparison result table Section 8 conclusion and then references.

Revised Manuscript Received on January 15, 2020.

Anjali, National Institute of Technology, Kurukshetra, Haryana, India
Deepak Garg, National Institute of Technology, Kurukshetra, Haryana, India
Richa Singh, National Institute of Technology, Kurukshetra, Haryana, India
Sarika Bathija, National Institute of Technology, Kurukshetra, Haryana, India

II. METAHEURISTIC ALGORITHMS

A. Flower pollination:

Laterin2012Yangdevelopedyetanotheroptimizationproblemc alledFPA.Sincethen it has been very effectively solving many real-world problems including image processing, communicational, computer gaming, energy and power, WSN, and many more. However, many variants of it have been proposed by hybridization, modification and parameter-tuning to deal with the complex nature of optimization problems.

Flower pollination is biological inspired natural process. Its useful characteristics used in designing an algorithm called flower pollination. FPA works more efficiently as compared to GA and PSO. Therefore, FPA can solve non-linear benchmark design.

FPA depends on some idealized principle which is given below:

R1) Levy Flight biotic and cross pollination acts as global search.

R2) whereas abiotic and self-pollination process performs local search.

R3) constancy in flowers can be improved by using similarity of two flowers.

R4) Switching amid local and global search is carried by random probability belongs to [0, 1].[9]

B. Particle Swam optimization:

It is an intelligent algorithm that is based on swarm nature inspired by insects & bird's social activities. Here particle is relatable to birds, fishes or ants. These particles fly through problem search space in which the evaluation of fitness values is done by fitness function. This fitness function needs to be optimized as well as their velocities that help them fly.[8]

C. Firefly algorithm:

Fireflies are the tiny beetles with wings. They have the ability to produce light with very little or no heat, it's called a cold light. To attract mates, a firefly emits some light. [7]

Pros: FA is having high convergence rate and robust. Finds optimum solution in less population

D. Ant Colony Optimization:

The ant colony optimization (ACO) a meta-heuristic algorithm was introduced in 1992 by Dorigo. Using this probabilistic technique, computational problems can easily be solved, which further using graphs helps to find good paths. This algorithm elevates from ant's behavior. Ants in the real-world scenario stand to wander in a random fashion to find food, when found,

They return to their settlement leaving pheromone tracks. This helps other ants not to roam in random motion but follow the trails and return and reinforce it if they find food.
Many new algorithms are introduced and they all have many applications. Another algorithm which was introduced in 2009 by Yang and Suash is cuckoo search. Later it was accepted that these algorithms are very efficiently solving non-linear engineering design problems by Yang and Deb. [6]

III. BAT ALGORITHM

Bat algorithm is a heuristic algorithm proposed by yang in 2010. It is based on the echolocation capability of bats. Initially bats don't know the location of the food. Solution can be expressed by the equation:
$X_{i\ i=}X_{min\ +}$rand (0,1) ($X_{max\ -}$ Xmin).where
$X_{max\ =}$upper range and $X_{min\ =}$lower range
And the new position can be found by the equation

$$f_i = f_{\min} + \beta\left(f_{\max} - f_{\min}\right),$$

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

$$x_i^t = x_i^{t-1} + v_i^t,$$

Where x_i^t and v_i^t are the position and velocity of the bat in the population at the t iteration.[3][7]

Following are the steps of bat algorithm: -

Objective function f(x), x=[x_1, x_2, ……x_d]^T

Initialize the bat population x_i (i=1, 2 …n) and v_i

Define pulse frequency f_i at x_i Initialize pulse rate r_i and the loudness A_i

While (t< Max number of iteration)

Generate new solutions by adjusting frequency,

And updating velocities and locations/solutions

If (rand> r_i)

Select a solution among the best solutions

Generate a local solution around the selected best solution

End if.

Generate a new solution by flying randomly

If (rand< A_i & f (x_i) <f(x_*))

Accept new solutions

Increase r_i and reduce A_i

End if.

Rank the bats and find the current best x_*

End While

Post process results and visualization.

Fig 1. Bat algorithm

IV. MODIFIED BAT ALGORITHM

This Algorithm improves the weakness of standard bat Algorithm by making the pulse rate and loudness dynamic in each iteration by the following equation: -

r_I= (current iteration/Max number of iterations) * pulse
$A_{i\ =}$ ((Max Iteration-current iteration)/max iteration) *loud

A. Dynamic Loudness in BAT:

Original bat uses high loudness for diversification and low loudness for intensification. Thus, by making the loudness dynamic decreases the no of input parameter. (equation given below)
Assigned value:
Loud bat=0.5
Dynamic loud=((MaxIter-Current)/MaxIter) *loud.

B. Dynamic pulse rate in bat:

Original bat uses low pulse rate for diversification and high pulse rate for intensification thus by making pulse rate dynamic we decrease the no. of input parameter (equation given below)
Assigned value:
Pulse (bat) =0.5
Dynamic pulse= (Current/MaxIter) *Pulse

Objective function f(x), x=[x_1, x_2 … x_d] ^{T}

Initialize the bat population x_I (i=1, 2 …n) and v_i

Define pulse frequency f_i at x_i Initialize pulse rate r_i and the loudness A_i

While (t< Max number of iteration)

Generate new solutions by adjusting frequency,

And updating velocities and locations/solutions

If (rand> r_i)

Select a solution among the best solutions

Generate a local solution around the selected best solution

End if.

Generate a new solution by flying randomly

If (rand< A_i & f (x_i) <f(x_*))

Accept new solutions

Make r_i and A_i dynamic by the equation

r_I = (i/Max number of iterations) * pulse

$A_{i\ =}$((Max Iteration-current)/max iteration)*loud

End if.

Rank the bats and find the current best x_*

End While

Post process results and visualization.

Fig 2. Modified Bat Algorithm

Levy Flight in Bat:

Moreover, levy flight can be used in bat as compared to simple random walk

V. FUNCTIONS USED

In this paper the performance of standard bat and modified bat algorithm is evaluated on the basis of the following benchmark functions:

A. Rosenbrock function:

$$Z=-\sin(x)*(\sin(x^2/pi))^{20}-\sin(y)*(\sin(2*y^2/pi))^{20}$$

Description:

Dimensions: d

It is unimodal in nature and its global minimum lies in a narrow, parabolic valley.

Global Minimum: $F(x^*) = 0$, at $x^* = (1,..,1)$[18]

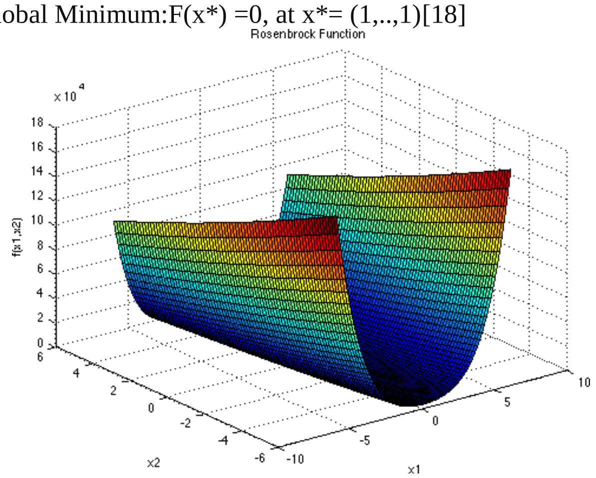

Fig 3. Rosenbrock f(n)

B. Wayburen function

$$F(x) = (x_1^6+x_2^4-17)^2+(2x_1+x_2-4)^2$$

Description:

Global optimum: $f(x_i)=0$ for $x=[1,2]$.

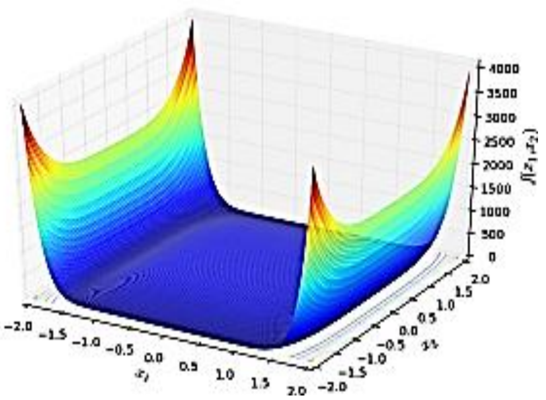

Fig 4. Wayburen f(n)

C. Rotated Ellipse Function

$$F_{107}(x)=7x_1^2-6*3^{1/2}x_1x_2+13x_2^2$$

Description:

This function is Differentiable, Non-Scalable, Continuous, Non-Separable, Unimodal [18]

VI. IMPLEMENTATION OF MODIFIED BAT

```
% Start the iterations -- Bat Algorithm (essential part)  %
xbest=ones(N_gen,d).*100;fbest=ones(N_gen,1).*100;favg=ones(N_gen,1).*100,
for t=1:N_gen,
% Loop over all bats/solutions
r=(t/N_gen)*r; %DYNAMIC PULSE
A=((N_gen-t)/N_gen)*A; %DYNAMIC LOUDNESS
        for i=1:n,
          Q(i)=Qmin+(Qmin-Qmax)*rand;
          v(i,:)=v(i,:)+(Sol(i,:)-best)*Q(i);
          S(i,:)=Sol(i,:)-v(i,:);
          % Apply simple bounds/limits
          S(i,:)=simplebounds(S(i,:),Lb,Ub);
          % Pulse rate
          if rand>r
          % The factor 0.001 limits the step sizes of random walks
              S(i,:)=best+0.001*randn(1,d);
          end
     % Evaluate new solutions
           Fnew=Fun(S(i,:));
     % Update if the solution improves, or not too loud
           if (Fnew<=Fitness(i)) & (rand<A) ,
                Sol(i,:)=S(i,:);
                Fitness(i)=Fnew;
           end
```

Fig 5. Modified bat code

VII. COMPARISON OF VARIOUS ALGORITHMS ON STANDARD BENCHMARK FUNCTIONS

7.1. Performance of algorithm on Rosenbrock function:

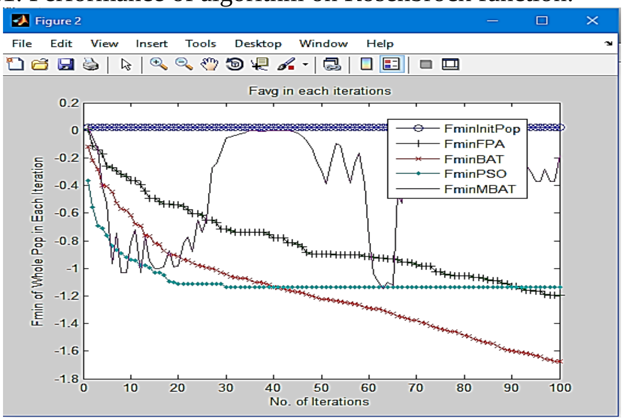

Fig 6. Rosenbrock f(n) graph

7.2. Performance of Algorithm on wayburen function:

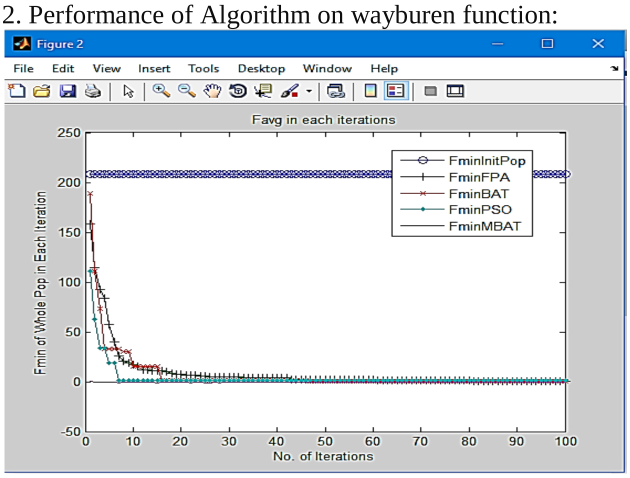

Fig 7. Wayburen f(n)graph

7.3. Performance of algorithm on Rotate Ellipse function:

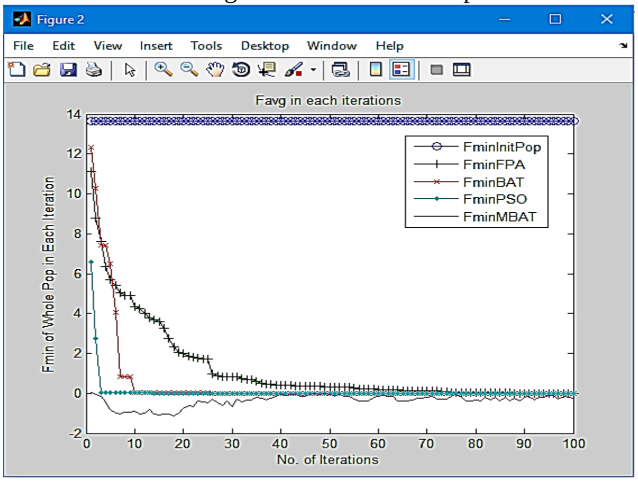

Fig 9. Rotated Ellipse f(n) graph

From the above 3 graph, it is depicted that all the functions when applied on several different algorithms they try to achieve their global minimum average values.

All the 3 graphs are showing that on all the 3 functions Modified bat has achieved closer value to its global minimum value i.e, 0.

VIII. EXPERIMENTAL RESULTS

8.1. Performance on Rosenbrock Function

1	FminInitPc	FminFPA	FminBAT	FminPSO	FminMBAT
2	0.022481	0.010461	-0.11718	-0.36488	0.022481
3	0.022481	-0.11444	-0.21748	-0.55686	-0.04774
4	0.022481	-0.14145	-0.28402	-0.69214	-0.13464
5	0.022481	-0.17153	-0.39671	-0.71104	-0.41276
6	0.022481	-0.25712	-0.40349	-0.75993	-0.53078
7	0.022481	-0.27165	-0.44399	-0.83673	-0.96678
8	0.022481	-0.29431	-0.52709	-0.86447	-0.74368
9	0.022481	-0.31626	-0.57013	-0.89132	-1.03278
10	0.022481	-0.33388	-0.58091	-0.91711	-1.02851
11	0.022481	-0.36191	-0.61358	-0.93466	-0.8072
12	0.022481	-0.36575	-0.67973	-0.94179	-0.72272
13	0.022481	-0.3912	-0.69205	-0.97593	-1.02848
14	0.022481	-0.44157	-0.75989	-0.98094	-0.76944
15	0.022481	-0.49281	-0.7666	-0.99301	-0.94908

Fig 10. Rosenbrock f(n) comparison data

8.2. Performance of algorithms on Rotate ellipse function

1	FminInitPop	FminFPA	FminBat	FminPSO	FminBAT
2	13.65072	11.14294	12.33153	6.584824	0.014865
3	13.65072	8.78396	10.27396	2.766142	-0.06446
4	13.65072	7.604483	7.423705	0.052355	-0.12948
5	13.65072	6.370987	7.423015	0.047837	-0.45264
6	13.65072	5.6901	6.481104	0.043908	-0.81154
7	13.65072	5.414058	4.061585	0.036706	-1.01181
8	13.65072	5.023771	0.839654	0.026682	-1.04972
9	13.65072	4.911662	0.837309	0.024372	-0.94193
10	13.65072	4.906099	0.834725	0.02117	-0.93158
11	13.65072	4.361676	0.047157	0.019986	-0.91636
12	13.65072	4.245408	0.043125	0.017721	-1.01889
13	13.65072	4.006924	0.040303	0.015572	-0.98854
14	13.65072	3.763669	0.040143	0.013206	-0.82213
15	13.65072	3.694077	0.037868	0.009101	-1.04499

Fig 11. Rotated ellipse f(n) comparison data

8.3. Performance of algorithm on wayburen function:

FminInitPop	FminFPA	FminBAT	FminPSO	FminMBAT
208.3457	158.5397	189.0198	110.7487	0.045776
208.3457	114.6216	111.3257	62.63748	-0.17833
208.3457	92.54289	73.01269	33.67373	-0.25864
208.3457	83.59689	33.27502	33.6712	-0.43009
208.3457	57.72645	33.2746	18.66207	-0.4225
208.3457	39.65869	33.20782	18.65759	-0.70481
208.3457	25.87787	32.8296	1.009598	-0.96499
208.3457	20.19167	29.9913	1.001464	-0.82544
208.3457	18.83841	29.9883	0.996732	-0.86382
208.3457	17.13751	15.19156	0.994372	-0.42269
208.3457	15.36809	15.18661	0.991297	-0.53462
208.3457	11.81505	15.18355	0.98771	-0.40716
208.3457	11.61952	15.17769	0.982587	-0.83045
208.3457	11.03786	15.1749	0.97634	-0.63662

Fig 12. Wayburen f(n) comparison data

Above tables represents the data after all the computations performed on all 3 functions. If we average out the data values for each algorithm. It will show the same results as graphs i.e, Global minimum values are achieved in a way that modified bat is showing more closer results to its global local values than any other algorithm.

Table I: Original Vs Modified Bat Performance

Rosenbrock f(n)		Rotated Ellipse f(n)		Wayburen f(n)	
FminBat	FminMBAT	FminBAT	FminMBAT	FminBAT	FminMBAT
-0.11718	0.022481	12.33153	0.014865	189.0198	0.045776
-0.21748	-0.04774	10.27396	-0.06446	111.3257	-0.17833
-0.28402	-0.13464	7.423705	-0.12948	73.01269	-0.25864
-0.39671	-0.41276	7.423015	-0.45264	33.27502	-0.43009

Retrieval Number: E5889018520 /2020©BEIESP
DOI:10.35940/ijrte.E5889.018520

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

-0.40349	-0.53078	6.481104	-0.81154	33.2746	-0.4225
-0.44399	-0.96678	4.061585	-1.01181	33.20782	-0.70481
-0.52709	-0.74368	0.839654	-1.04972	32.8296	-0.96499
-0.57013	-1.03278	0.837309	-0.94193	29.9913	-0.82544
-0.58091	-1.02851	0.834725	-0.93158	29.9883	-0.86382
-0.61358	-0.8072	0.047157	-0.91636	15.19156	-0.42269
-0.67973	-0.72272	0.043125	-1.01889	15.18661	-0.53462
-0.69205	-1.02848	0.040303	-0.98854	15.18355	-0.40716
-0.75989	-0.76944	0.040143	-0.82213	15.17769	-0.83045
-0.7666	-0.94908	1.003786	-1.04499	15.1749	-0.63662
-0.503775	**-0.65372**	**3.691507**	**-0.72637**	**45.84565**	**-0.53103**

***Note: Below red row is the average out of the tabular data for original & modified bat for Rosenbrock, rotated ellipse & wayburen f(n).** On all the functions modified bat has achieved closer values to its global local i,e 0 than original bat and in table its average for rosenbrock is -0.65372, for wayburen its -0.72637 & -0.53103.

IX. CONCLUSION

Based on deep theoretical surveying of weakness of standard BAT algorithm it has been identified that the proposed Modified Bat should give better results than the old one due to three main inclusion which are Dynamic pulse rate, Dynamic loudness, Levy flight. To prove superiority of Modified BAT over old one, Practical comparison has been done on Rosenbrock rotated ellipse function and on all its variants. On comparison of results also it has been proved that modified bat gives better fmin (minimum fitness in every iteration) and favg (average fitness in every iteration) thus it is more general with better performance. [1]

REFERENCES

1. Yang, X. S., “Bat algorithm for multi-objective optimisation.”, arXiv preprint arXiv:1203.6571. (2012).
2. Yang, X. S., & Hossein Gandomi, A.” Bat algorithm: a novel approach for global engineering optimization. Engineering Computations”, 29(5),464-483. (2012).
3. Yang, X. S.. “A new metaheuristic bat-inspired algorithm. In Nature inspired cooperative strategies for optimization”, (NICSO 2010) (pp. 65-74). Springer, Berlin, Heidelberg. (2010).
4. Z.A. Abas, m.r.Ramli,m.i.Desa, n.Saleh, a.n. Hanafia, n. Aziz, z.z. abidin, a.s.Shibghatullah, af.n.a. Rehman, H.Musa, “ A supply model for nurse workforce projection in maleysia, Health care manage.Sci, pp. 1-14(2017).
5. E.S. Ali, “Optimization of power system stabalizers using bat search algorithmInt.J. Electr. Power Energy Syst”., pp. 683-690.
6. Dorigo, M., & Stützle, T. “The ant colony optimization metaheuristic: Algorithms, applications, and advances”, In Handbook of metaheuristics (pp. 250- 285). Springer, Boston, MA. (2003).
7. Yang, X. S. “Bat Algorithm, Firefly algorithm, Levy flights and global optimization”, In Research and development in intelligent systems XXVI (pp. 209-218). Springer, London. (2010).
8. Kennedy, J. “Particle swarm optimization. In Encyclopedia of machine learning”, Springer, Boston, MA. pp. 760-766 (2011).
9. Alam, D. F., Yousri, D. A., & Eteiba, M. B. “Flower pollination algorithm based solar PV parameter estimation. Energy Conversion and Management”, 101, pp.410-422 (2015).
10. Fister Jr, I., Yang, X. S., Fister, I., Brest, J., &Fister, D. “A brief review of nature inspired algorithms for optimization”, arXiv preprintarXiv:1307.4186(2013).
11. IztokFister Jr., Simon Fong, Janez Brest, I.F., “A Novel Hybrid Self-Adaptive Bat Algorithm”, The Scientific World Journal, (Article ID 709738), pp. 1(2014).
12. Jamil et al., “Improved bat algorithm for global optimization. Appl. Soft Comput.”, (2013).
13. Komarasamy, A. WahiAn, “Optimized k-means clustering technique using bat algorithm”, Eur. J. Sci. Res., 84 (2) , pp. 263-273(2012).
14. Lihong Guo, Gai-Ge Wang, Heqi Wang, D.W., “An Novel Hybrid Bat Algorithm with Harmony Search for Global Numerical Optimization”, The Scientific World Journal (Article ID 125625), pp. 9(2013).
15. J. Lin, C. Chou, C. Yang, H. TsaiA Chaotic, “Levy Flight Bat Algorithm for Parameter Estimation in Nonlinear Dynamic Biological Systems”.
16. M. Naderi, E. Khamehchi, “Application of DOE and metaheuristic bat algorithm for well placement and individual well controls optimization”.
17. D. Garg and P. Kumar, “A Survey on Metaheuristic Approaches and its Evaluation for Load Balancing in Cloud Computing”, In: Springer Nature CCIS, Advanced Informatics for Computing Research ICAICR, vol. 955, pp. 585-599 (2018).
18. D. Garg, P. Kumar, “Evaluation and Improvement of Load Balancing Using Proposed Cuckoo Search in CloudSim”, In: Communications in Computer and Information Science, Springer, Singapore, Advanced Informatics for Computing Research ICAICR, Vol. 1075, pp. 343-358, (2019).
19. A. ku. Shakya, D. Garg, P. Ch. Nayak, "Hybrid Live VM Migration: An Efficient Live VM Migration Approach in Cloud Computing", In: Springer Nature CCIS, Advanced Informatics for Computing Research ICAICR, Vol. 955, pp. 600-611, (2018).
20. P. Ch. Nayak, D. Garg, A. Ku. Shakya, P. Saini, "A research paper of existing Live VM Migration and a Hybrid VM Migration approach in Cloud Computing", In: IEEE 2nd International Conference on Trends in Electronics and Informatics (ICOEI 2018), pp. 721-726, (2018).
21. A. Harkawat, S. Kumari, P. Pharkya, D. Garg, “Load Balancing Task Scheduling Based on Variants of Genetic Algorithm: Review Paper”,in Communications in Computer and Information Science, Springer, Singapore, ICICCT, 750, pp.318-325, (2017).
22. D. Garg, P. Garg, “Basis Path Testing Using SGA & HGA With ExLB Fitness Function” in Elsevier, Procedia Computer Science, Vol. 70, pp. 593-602, Dec (2015).

AUTHORS PROFILE

Author Name: Anjali, Profession: Student(MCA) Institute: NIT, kurukshetra, Haryana, India Research Work (Previous paper): Evaluation and Extension of meta-heuristic algorithms including bat, flower pollination, particle swarm optimization, firefly & ant colony optimization on Rosenbrock function.

Author Name: Deepak Garg, Profession: Assistant Professor (MCA) Institute: NIT, kurukshetra, Haryana, India Research Work:Includes Metaheuristics and Load balancing in Cloud and have published more than 20 Research paper.

Author Name: Sarika Bathija, Profession: Student(MCA) Institute: NIT, kurukshetra, Haryana, India Research Work (Previous paper): Evaluation and Extension of meta-heuristic algorithms including bat, flower pollination, particle swarm optimization, firefly & ant colony optimization on Rosenbrock function.

Author Name: Richa Singh, Profession: Student(MCA) Institute: NIT, kurukshetra, Haryana, India Research Work (Previous paper): Evaluation and Extension of meta-heuristic algorithms including bat, flower pollination, particle swarm optimization, firefly & ant colony optimization on Rosenbrock function.