12th International Conference on Recent Development in Engineering Science, Humanities and Management

The International Centre Goa, Panjim, Goa (India)

14th April 2019 www.conferenceworld.in ISBN: 978-93-87793-85-9

Comparative Implementation of Conventional Honey Bee Algorithm with Modified Honey Bee Algorithm

Geetu[1], Dr.Anand Sharma[2]

[1]*Research Scholar (PhD Comp. Applications), Guru Kashi University, Talwandi Sabo, Punjab, India.*

[2]*Assistant Professor, UCCA, Guru Kashi University, Talwandi Sabo, Punjab, India.*

Abstract

Load balancing is a method for distributing tasks onto multiple computers. For instance, distributing incoming HTTP requests (tasks) for a web application onto multiple web servers. The primary purpose of load balancing is to distribute the work load of an application onto multiple computers, so the application can process a higher work load. Load balancing is a way to scale an application. The research paper focuses on implementation of HBA (Honey Bee algorithm) to find the best cost involved in finding path from source to destination. The paper implements and records the reading obtained from conventional HBA and modified HBA. The parameters utilized in the operational implementation can be dynamically changes as per preference.

Keywords: Bees, Conventional HBA algorithm, load balancing, modified HBA algorithm.

I. INTRODUCTION

HBA (Honey Bee algorithm) is an optimization technique that simulates the foraging behavior of honey bees, and has been successfully applied to various practical problems. HBA belongs to the group of swarm intelligence algorithms and was proposed by Karaboga in 2005 [10, 11].

A set of honey bees, called swarm, can successfully accomplish tasks through social cooperation. In the HBA algorithm, there are three types of bees: employed bees, onlooker bees, and scout bees [1, 2]. The employed bees search food around the food source in their memory; meanwhile they share the information of these food sources to the onlooker bees. The onlooker bees tend to select good food sources from those found by the employed bees [15, 16]. The food source that has higher quality (fitness) will have a large chance to be selected by the onlooker bees than the one of lower quality. The scout bees are translated from a few employed bees, which abandon their food sources and search new ones [3, 4, 12].

In the HBA algorithm, the first half of the swarm consists of employed bees, and the second half constitutes the onlooker bees [13, 19].

The number of employed bees or the onlooker bees is equal to the number of solutions in the swarm [17, 18]. The HBA generates a randomly distributed initial population of SN solutions (food sources), where SN denotes the swarm size [5, 6, 14].

II. IMPLEMENTATION AND RESULTS

The research work shown in this section elaborates the comparison done on the account of calculating best cost involved in finding the best path from source to destination between existing HBA algorithm and modified HBA algorithm.

Different parameters considered in calculating the best cost are mentioned as under.

- nVar - Number of decision variables
- VarMin - Lower Bound of Decision Variables
- VarMax - Upper Bound of Decision Variables
- MaxIt - Refers to maximum number of iterations
- nScoutBee - Number of scout Bees
- nSelectedSite - Number of selected sites
- nEliteSite - Number of selected Elite sites
- nSelectedSiteBee - Number of recruited bees for selected sites
- nEliteSiteBee - Number of Recruited Bees for Elite Sites
- r - Neighborhood Radius
- rdamp - Neighborhood Radius damp rate

The readings assigned to different parameters are given below.

- nVar = 5
- VarMin = -10
- VarMax = 10
- MaxIt = 1000
- nScoutBee = 30
- nSelectedSite = round (0.5 * nScoutBee) = 15
- nEliteSite = round (0.4 * nSelectedSite) = 6
- nSelectedSiteBee = round(0.5 * nScoutBee) = 15
- nEliteSiteBee = 2 * nSelectedSiteBee = 30
- r = 0.1*(VarMax-VarMin)
- rdamp = 0.95

Existing HBA algorithm

Fig. 1 shows the flowchart of existing HBA algorithm [7, 8, 9].

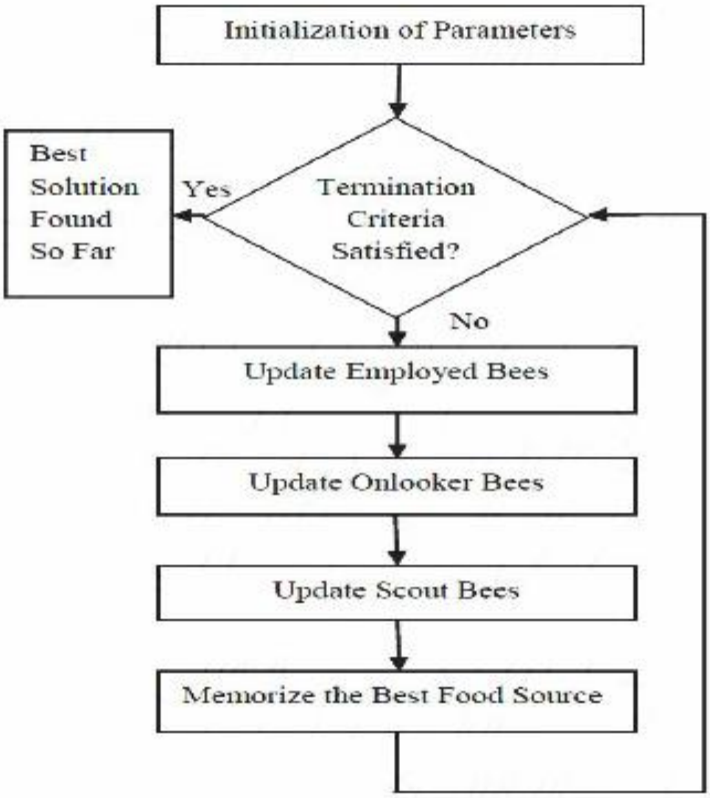

Fig. 1: Figure shows the flowchart of existing HBA algorithm

Fig. 2 below shows the initially calculated best cost as per existing HBA algorithm as 48.4561.

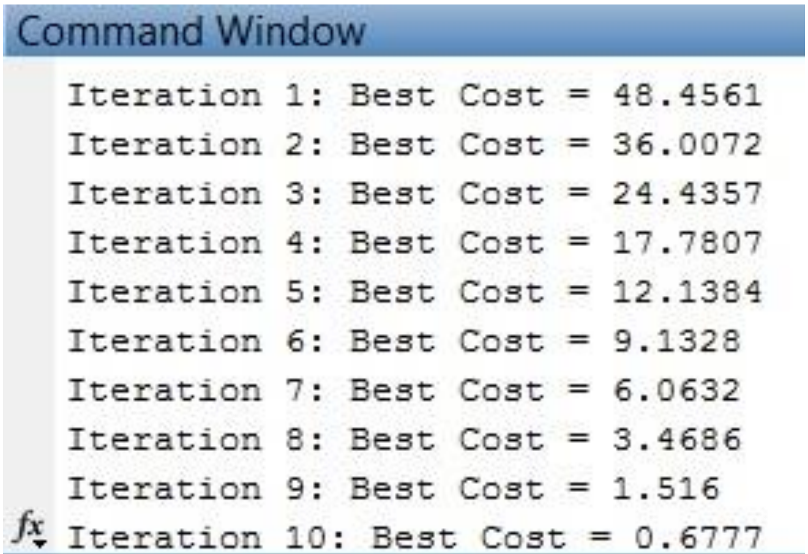

Fig. 2: Figure shows the readings of initial iterations

ISBN: 978-93-87793-85-9

Thereafter, the existing HBA algorithm undergoes multiple iterations (1000 in this case) and the finally obtained best cost is 8.9449e-49 as shown in Fig. 3.

Fig. 3: Figure shows the readings of final iterations

Fig. 4 shows the generated graph of existing HBA algorithm having number of iterations as x-axis and the calculated best cost as y-axis.

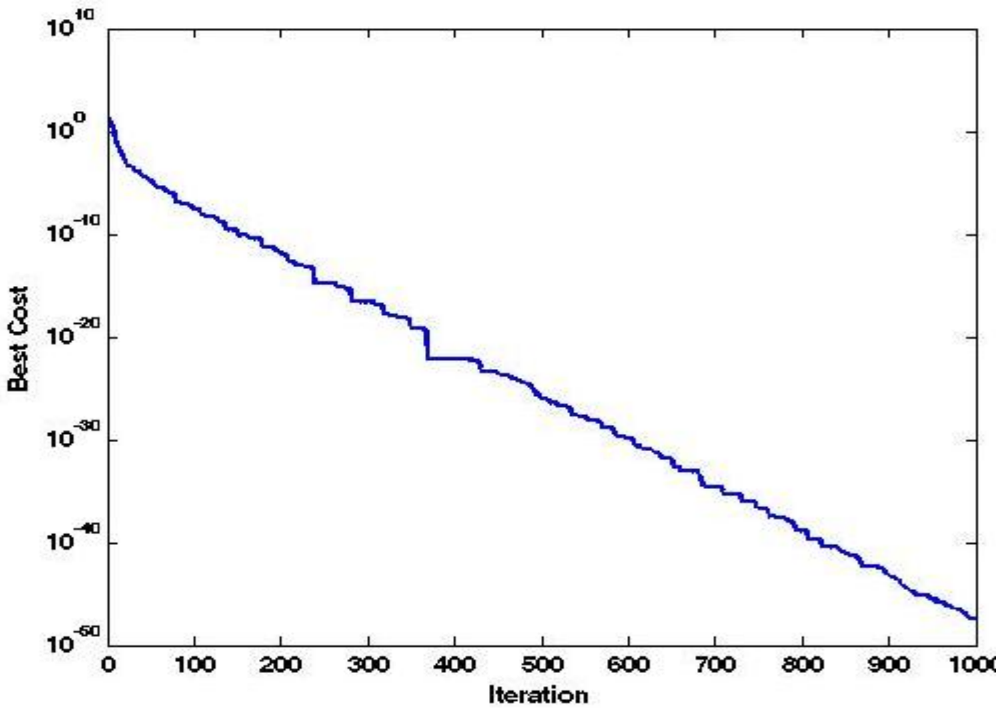

Fig. 4: Figure shows the generated graph having number of iterations as x-axis and the calculated best cost as y-axis.

Modified HBA algorithm

Fig. 5 shows the flowchart of modified HBA algorithm.

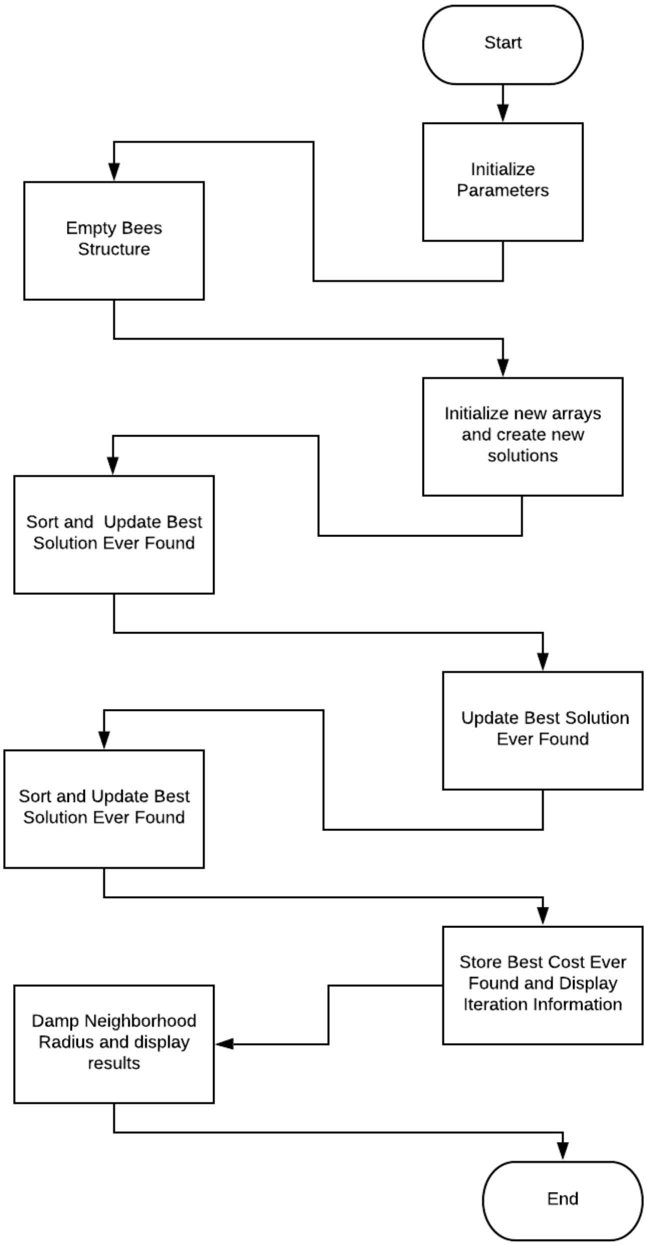

Fig. 5: Figure shows the flowchart of modified HBA algorithm.

Fig. 6 shows the initial readings obtained as per modified HBA algorithm. The first iteration calculates the best cost to 10.7814 as compared to 48.4561 in case of existing HBA algorithm. So, the modified HBA algorithm reduced the best cost more than four times the cost calculated by existing algorithm.

ISBN: 978-93-87793-85-9

Command Window

```
Iteration 1: Best Cost = 10.7814
Iteration 2: Best Cost = 5.1428
Iteration 3: Best Cost = 3.2681
Iteration 4: Best Cost = 1.6164
Iteration 5: Best Cost = 0.49147
Iteration 6: Best Cost = 0.15669
Iteration 7: Best Cost = 0.042043
Iteration 8: Best Cost = 0.0060094
Iteration 9: Best Cost = 0.0032892
fx Iteration 10: Best Cost = 0.0024486
```

Fig. 6: Figure shows the initial readings obtained as per modified HBA algorithm

Fig. 7 shows the final readings obtained after running multiple iterations (1000 in this case) as per modified HBA algorithm. The last obtained iteration shows the calculated best cost as low as 1.0607e-49 which is much lower than the 8.9449e-49 as calculated by existing HBA algorithm.

Command Window

```
Iteration 992: Best Cost = 4.5813e-49
Iteration 993: Best Cost = 4.5813e-49
Iteration 994: Best Cost = 3.9407e-49
Iteration 995: Best Cost = 3.9407e-49
Iteration 996: Best Cost = 3.738e-49
Iteration 997: Best Cost = 3.6908e-49
Iteration 998: Best Cost = 3.6908e-49
Iteration 999: Best Cost = 3.6908e-49
Iteration 1000: Best Cost = 1.0607e-49
```

Fig. 7: Figure shows the final readings obtained after running multiple iterations

Fig. 8 shows the generated graph of modified HBA algorithm having number of iterations as x-axis and the calculated best cost as y-axis.

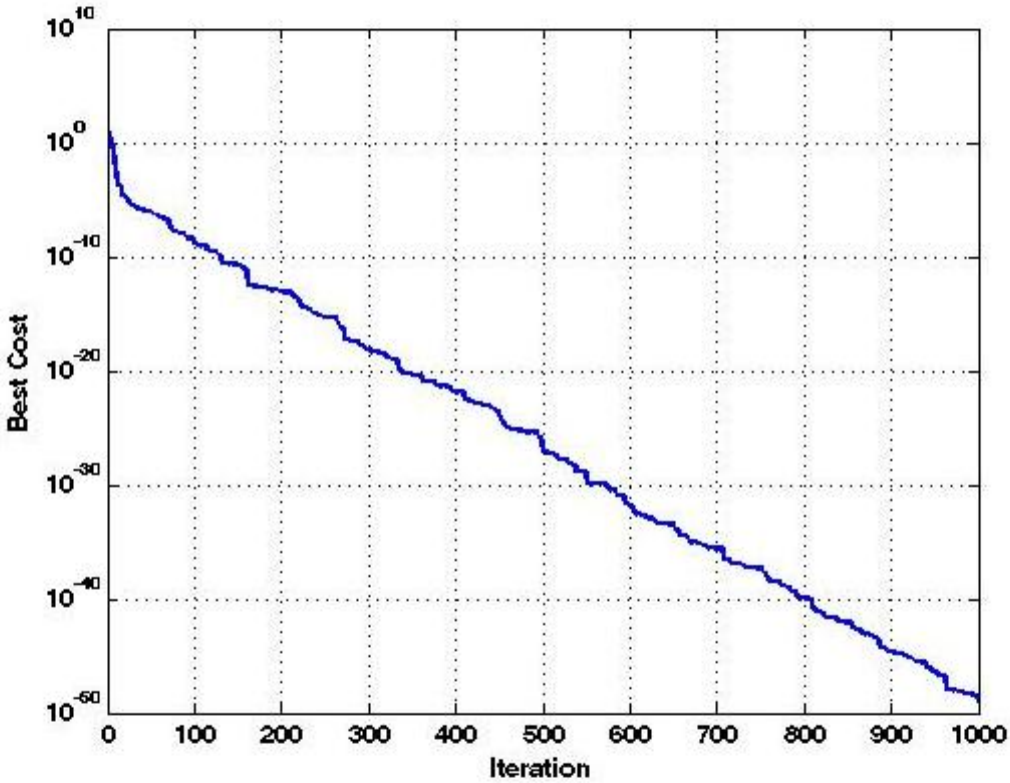

Fig. 8: Figure shows the generated graph of modified HBA algorithm having number of iterations as x-axis and the calculated best cost as y-axis.

Table 1 shows the comparative values obtained by executing two different algorithms under study.

Table 1: Table shows best cost values obtained on executing algorithms under study at iteration 1 and iteration 1000.

Iterations	Existing HBA algorithm	Modified HBA algorithm
Iteration 1	48.4561	10.7814
Iteration 1000	8.94E-49	1.06E-49

Fig. 9 and Fig. 10 shows the graphical representation of the values mentioned in Table 1. Fig. 9 illustrates the iteration 1 readings and Fig. 10 shows the readings obtained at iteration 1000.

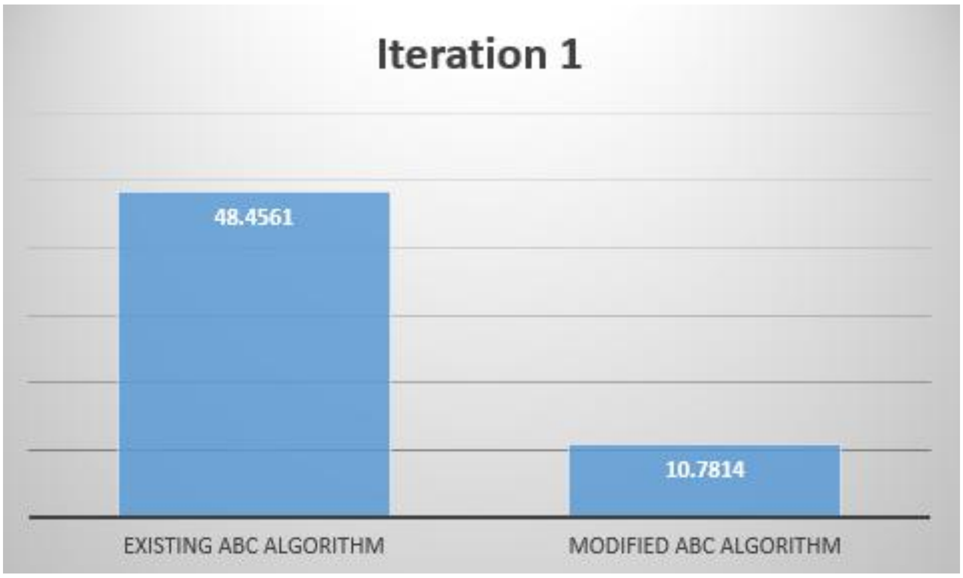

Fig. 9: Figure shows the graphical representation of reading of best cost obtained against iteration 1 on executing the two algorithms under study.

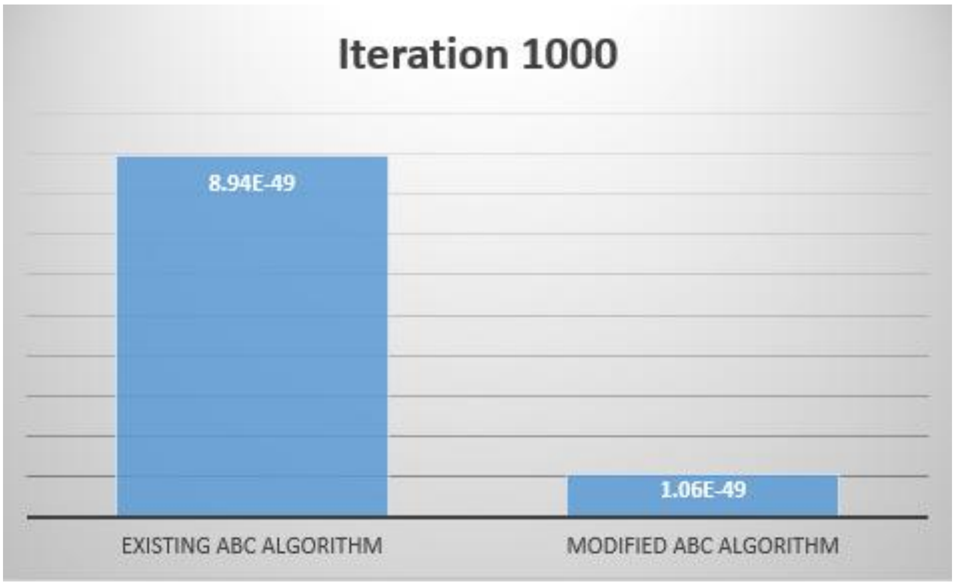

Fig. 10: Figure shows the graphical representation of reading of best cost obtained against iteration 1000 on executing the two algorithms under study.

III. CONCLUSION

On the basis of the study conducted and the obtained readings, it can be concluded that modified HBA algorithm in much more efficient in calculating best cost as compared to conventional HBA algorithm.

REFERENCES

1. D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," Applied Soft Computing, vol. 8, no. 1, pp. 687–697, 2008.
2. Gagandeep Jagdev et al., "Analyzing Commercial Aspects and Security Concerns Involved in Energy Efficient Cloud Computing" in International Journal of Scientific and Technical Advancements (IJSTA), ISSN-2454-1532, 2016.
3. A. L. Bolaji, A. T. Khader, M. A. Al-Betar, and M. A. Awadallah, "Artificial bee colony algorithm, its variants and applications: a survey," Journal of 3eoretical & Applied Information Technology, vol. 47, no. 2, 2013.
4. J. C. Bansal, H. Sharma, and S. S. Jadon, "Artificial bee colony algorithm: a survey," International Journal of Advanced Intelligence Paradigms, vol. 5, no. 1-2, pp. 123–159, 2013.
5. B. Jadon and D. Karaboga, "A survey on the applications of artificial bee colony in signal, image, and video processing,"Signal, Image and Video Processing, vol. 9, no. 4, pp. 967–990, 2015.
6. Gagandeep Jagdev et al., "Implementation of DES and AES Cryptographic Algorithms in Accordance with Cloud Computing", International Journal of Research Studies in Computer Science and Engineering (IJRSCSE), ISSN: 2349-4840 (P), ISSN: 2349-4859(O), DOI: http://dx.doi.org/10.20431/2349-4859.0404001**,** Volume 4, Issue 4, 2017, pp – 1 - 14.
7. R. Ramesh, C. Gomathy, D. Vaishali et al., "Bio inspired optimization for universal spatial image steganalysis,"Journal of Computational Science, vol. 21, pp. 182–188, 2017.
8. X.-M. Gao, S.-F. Yang, and S.-B. Pan, "Optimal parameter selection for support vector machine based on artificial bee colony algorithm: a case study of grid-connected pv system power prediction," Computational intelligence and neuroscience, vol. 2017, Article ID 7273017, 14 pages, 2017.
9. Gagandeep Jagdev et al., "Analyzing Working of DES and AES Algorithms in Cloud Security", International Journal of Research Studies in Computer Science and Engineering (IJRSCSE), ISSN: 2349-4840 (P), ISSN: 2349-4859(O), DOI: http://dx.doi.org/10.20431/2349-4859.0403001**,** Volume 4, Issue 3, 2017, pp – 1 - 9.
10. G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," Applied mathematics and computation, vol. 217, no. 7, pp. 3166–3173, 2010.
11. B. Akay and D. Karaboga, "Artificial bee colony algorithm for large-scale problems and engineering design optimization," Journal of Intelligent Manufacturing, vol. 23, no. 4, pp. 1001–1014, 2010.
12. W. Gao, S. Liu, and L. Huang, "A global best artificial bee colony algorithm for global optimization," Journal of Computational and Applied Mathematics, vol. 236, no. 11, pp. 2741–2753, 2012.
13. Gagandeep Jagdev et al., "Elaborating Security Algorithms in Cloud Environment", International Journal of Research Studies in Computer Science and Engineering (IJRSCSE), ISSN: 2349-4840 (P), ISSN: 2349-4859(O), Volume 5, Issue 1, 2018, pp.1-8, DOI: http://dx.doi.org/10.20431/2349-4859.0501001

14. W.-L. Xiang and M.-Q. An, "An efficient and robust artificial bee colony algorithm for numerical optimization," Computers & Operations Research, vol. 40, no. 5, pp. 1256–1265, 2013.
15. J. Luo, Q. Wang, and X. Xiao, "A modified artificial bee colony algorithm based on converge-onlookers approach for global optimization," Applied Mathematics and Computation,vol. 219, no. 20, pp. 10253–10262, 2013.
16. W.-F. Gao, S.-Y. Liu, and L.-L. Huang, "A novel artificial bee colony algorithm based on modified search equation and orthogonal learning,"IEEE Transactions on Cybernetics, vol. 43, no. 3, pp. 1011–1024, 2013.
17. Gagandeep Jagdev et al., "Comparative Analysis and Implementation of Cryptographic Algorithms in Cloud Computing", International Journal of Research Studies in Computer Science and Engineering (IJRSCSE), ISSN: 2349-4840 (P), ISSN: 2349-4859(O), Volume 5, Issue 1, 2018, pp.1-9, DOI: http://dx.doi.org/10.20431/2349-4859.0501003.
18. W.-F. Gao, L.-L. Huang, S.-Y. Liu, and C. Dai, "Artificial bee colony algorithm based on information learning," IEEE transactions on cybernetics, vol. 45, no. 12, pp. 2827–2839, 2015.
19. D. C. Tran, Z. Wang, Z. Wu, and C. Deng, "A novel hybrid data clustering algorithm based on artificial bee colony algorithm and k-means," Chinese Journal of Electronics, vol. 24,no. 4, pp. 694–701, 2015.