# Analysis study of the bee algorithms as a mechanism for solving combinatorial problems

**Article** *in* Indonesian Journal of Electrical Engineering and Computer Science · May 2023

DOI: 10.11591/ijeecs.v30.i2.pp1091-1098



Hafedh Ali Shabat
Al-Furat Al-Awsat Technical University


Khamael Raqim
Al-Furat Al-Awsat Technical University

# Analysis study of the bee algorithms as a mechanism for solving combinatorial problems

**Hafedh Ali Shabat, Khamael Raqim Raheem**
Technical Institute of Babylon, Al-Furat Al-Awsat Technical University (ATU), Kufa, Iraq

**Article Info**

***Article history:***

Received Oct 27, 2022
Revised Jan 10, 2023
Accepted Jan 12, 2023

***Keywords:***

Artificial bee colony
Bee colony optimization
Optimization
Swarm-based optimization
Waggle dance

**ABSTRACT**

Combinatorial optimization problems are problems that have a large number of discrete solutions and a cost function for evaluating those solutions in comparison to one another. With the vital need of solving the combinatorial problem, several research efforts have been concentrated on the biological entities behaviors to utilize such behaviors in population-based metaheuristic. This paper presents bee colony algorithms which is one of the sophisticated biological nature life. A brief detail of the nature of bee life has been presented with further classification of its behaviors. Furthermore, an illustration of the algorithms that have been derived from bee colony which are bee colony optimization, and artificial bee colony. Finally, a comparative analysis has been conducted between these algorithms according to the results of the traveling salesman problem solution. Where the bee colony optimization (BCO) rendered the best performance in terms of computing time and results.



***Corresponding Author:***

Hafedh Ali Shabat
Technical Institute of Babylon, Al-Furat Al-Awsat Technical University (ATU)
Kufa, Iraq
Email: h.ali@atu.edu.iq

## 1. INTRODUCTION

Nature oftentimes exhibits swarm intelligence, which is the collective behavior of self-organized particles. Swarm intelligence methods use a technique dependent on a search that employs a distributed approach in which each agent acts independently [1]. These agents work with their neighbors to learn more about the environment.

Trendy optimization is increasingly relying on metaheuristic methods. Over the last three decades, a vast range of metaheuristic algorithms has arisen, and several of them, such as the optimization technique known as particle swarm optimization (PSO), which is gaining more and more popularity [2]. The majority of Metaheuristic algorithms are inspired by nature [3], [4], for instance, ant colony optimization (ACO) [5], PSO [6], and cuckoo search algorithms [7]. Several new Metaheuristic algorithms have been produced after the coming to light of swarm intelligence approaches like the PSO that appear in the 1995s, and these techniques have been used in practically different fields of optimization, text mining [8], scheduling and planning [9], and machine intelligence [10].

According to a recent study, the Bee algorithm acts particularly better or at least analogous to other swarm intelligence algorithms [11]. The bee algorithm is a swarm intelligence algorithm inspired by bees' behavior [12]. Since the evolution of the bee algorithm, it has been applied to solve various varieties of problems [13].

The goal of this study is to discuss two kinds of bee algorithms, optimization algorithms inspired by honey bees' natural foraging and behavior, for finding an optimal solution. Both an exploitative neighborhood search and a random explorative search are performed by this algorithm. Bees have a distinct search pattern that makes them have many methods for finding food. Two different types of bee algorithms based on behavior, communication, components, and decision-making will be presented, by employing these algorithms for solving the traveling salesman (TSP) problem. The remains parts of the paper are organized as described as follows: artificial bee colony algorithm (ABC) and bee colony optimization (BCO) algorithms were discussed in section 2 along with their methodology. Section 3 provides the suggested experimental setup and solution approach for TSP, and presents the experimental findings, while section 4 presents a conclusion.

## 2. BEE ALGORITHMS METHODS

Since the evolution of the bee colony algorithms, it has tempted much awareness for its superior characteristics. In the three last decades, various versions of BCs have been used for diverse problems. This section presents a brief review of two BCs algorithms.

### 2.1. Artificial bee colony algorithm

A metaheuristic technique known as ABC is described as one in which artificial bees work together to find the best solutions to numerical optimization problems. Generally, this algorithm needs extensive resources and a larger computational time [14]. ABC is one member of the swarm intelligence that simulates the behaviors of honey bees to locate food, as presented by Karaboga [12]. Honeybees employ waggle dances and other mechanisms to enhance the position of food sources and discover a new one.

In the ABC algorithm, the bees move around in a search space that has multiple dimensions, and certain bees (employed and onlookers) prefer food sources according to their expertise and partners, and then determine the location. Some bees choose a food source at random, without any prior knowledge or experience. If the quantity of nectar from a new source is greater than the amount of nectar from the prior source (based on the memory of the bee), the new location will be kept in memory and will overwrite the old position. In this algorithm, the local search techniques that are performed by employed and onlookers bees are merged with a global search approach that is managed by onlookers and scouts [15].

In the hive, both the employed and onlooker bees are divided equally. The employed bee's number equaled the number of food sources available. At each source of food, there is a single employed bee at that place. The food source quality, which is associated with the location, represents the fitness value [16]. The technique employed by bees to find sources of food is utilized to locate the best solution [17]. The steps involved in the ABC algorithm are depicted in Figure 1.

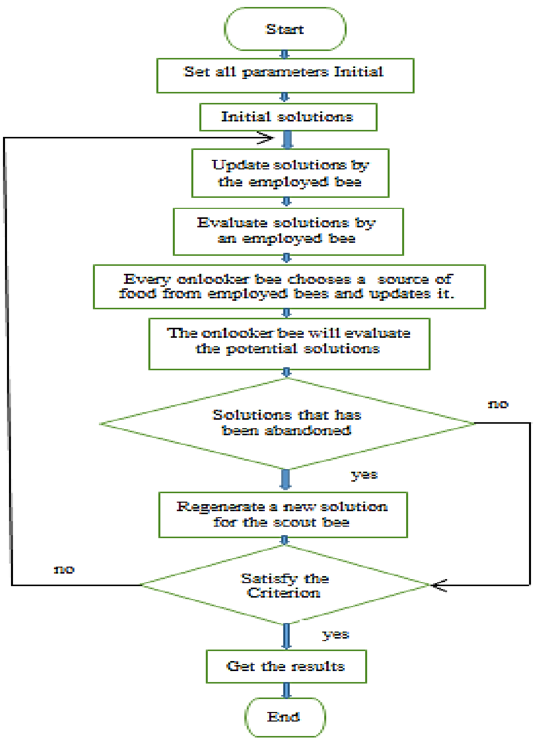


Figure 1. Steps of the ABC algorithm [14]

The investigation of food sources is done by employed bees and then sharing their information to recruit the onlooker bees. Depending on this information, the onlooker will decide which food source will choose. The source of food with higher quality will be chosen by the onlooker bees over the food source with a lower quality. When a food source is ignored by both employed and onlooker bees because it is of low quality, the employed bee transforms into a scout bee that seeks new food sources at random. The algorithm is described in more depth:

The first step that needs to be done is to generate initial food source locations that are distributed at random. The process is shown in (1):

$$F(x_i), x_i \in R^D, i \in \{1,2, \dots \dots, S_N\} \tag{1}$$

where $x_i$ represents the location of the food source as a D-dimensional vector, the objective function is denoted by $F(x_i)$ which dictates how best a solution should be, and the food source's number is indicated by $S_N$. Following that, the population is put through reiterations of three main steps, which are referred to as: updating feasible solutions, selecting feasible solutions, and avoiding suboptimal solutions, respectively. All employed bees choose a new candidate food source location to update feasible solutions. The decision is made depending on the neighbors of the previously chosen source of food. According to (2) is used to compute the location of the new food source.

$$v_{ij} = x_{ij} + \emptyset_{ij}(x_{ij} - x_{kj}) \tag{2}$$

Where $v_{ij}$ indicates a newly viable solution that has been updated. $x_{ij}$ stands for the solution that came before it, $x_{kj}$ is the solution that is immediately neighboring to it, and the random number $\emptyset_{ij}$ falls within the range [-1,1]. This value is the one that determines how the previous solution is modified during the next iteration so that it can evolve into the new solution. $K \in \{1,2,\dots,S_N\}$ and $j \in \{l, 2, \dots , D\}$. A difference in location along a particular dimension is denoted by the notation $(x_{ij}-x_{kj})$. Based on the fitness value, the location of the old source of food that was stored in the memory of the employed bee will be substituted by the new food source location, when the fitness value offered by the new position is superior to that of the previous one. Employed bees will share the fitness value of new sources of food with the onlooker bee when going back to the hive. According to the fitness value acquired from the employed bee, every single onlooker bee will choose one of the suggested sources of food. The probability of choosing the source of food is depicted by (3):

$$P_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \tag{3}$$

where $P_i$ is the probability of an onlooker selecting a food source which it is increase when the fitness value increases [18]. Fiti represents the fitness value of the source of food i. Then, onlooker bees will fly to the specific food source and choose a new candidate source of food location in the neighborhood of the picked food source. To avoid suboptimal solutions, within the third stage, the source of food that does not enhance the value of fitness will be left and changed to a new location which is selected by the scout bees randomly [14]. The scout bee's new random position will be calculated using (4).

$$x_{ij} = x_j^{min} + rand[0,1] * \left(x_j^{max} - x_j^{min}\right) \tag{4}$$

Where $x^{min}$ represents the lower bound of the food source location while the $x^{max}$ indicates the upper bound of the location of the source in dimension j [19]. Max cycle number (MCN) is a termination condition that is used to specify the number of iterations. The operation will be continued until the objective function's output meets a predetermined the value of threshold or the iteration number reaches the MCN. The choice of a threshold value that is equivalent to either the global minimum or maximum value is made under the type of optimization problem being addressed [20]. ABC was used in many applications such as constrained problem optimization, numerical assignment problem, and bioinformatics fields [21]. The successful application of ABC and its rapid growth was caused to the development of the other versions of the algorithm which will be presented in the next subsection.

### 2.2. Bee colony optimization algorithm

The BCO is a population-based algorithm. It mimics the behavior of bees in nature to a greater degree. This algorithm is distinguished by the presence of scout bees, an essential role of hive location, and a recruiting method that is more similar to the natural one. In BCO, artificial bees work together cooperatively to solve complicated combinatorial optimization problems [22].

At the start of the search process, all artificial bees exist in the hive. Artificial bees intercommunicate directly with each other during the process of search. Each one of bee executes a sequence of local movements and gradually builds a solution to the problem. Bee continues adding components of solution to the current partial solution till one or more viable solutions are constructed. In BCO, several iterations are performed throughout the search process. The first iteration of the process is considered finished when the bees produce one or more viable solutions for the first time. After saving the better solution found during the first iteration, will start the second iteration. In this manner, bees incrementally develop solutions to the problem in the second iteration and continue with subsequent iterations. At the end of per iteration, one or more partial solutions have been generated [23].

Artificial bees in the BCO collaboratively search to find the optimal solution, where one solution to the problem is generated by each artificial bee when a forward pass or backward pass is performed by the bees as they fly through the search space. In every forward pass, artificial bees explores the search area and generates diverse partial solutions. where the bees use a predetermined number of movements to create or enhance the solution, producing a new solution. They accomplish this through a mixture of individual exploration and cooperative past experience. After that, the bees will execute the backward pass process, where they go back to the hive again after finding a new partial solution. Information about solution quality is exchanged throughout the hive among all bees. This implies that every bee is involved in the decision-making process. The value of an objective function is calculated. After evaluating all of the possible solutions, each bee decides with a certain probability regarding whether or not to leave the formed partial solution and become an uncommitted follower, continue to expand the newly constructed partial solution without recruiting the nest-mates or dance to recruit the nest-mates before returning to the newly constructed partial solution. Relying on the quality of the solutions that are produced, every bee displays a distinct level of allegiance to the route that leads to the previously found partial solution. This level of allegiance is determined by the quality of the generated partial solutions.

The bees expand previously constructed partial solutions via the second forward pass, then use the backward pass again, and after that go back to the hive. within the hive, bees partake decision-making process again, execute a third forward pass, and so on. When one or more viable solutions are found, the iteration comes to a close [24]. The forward and backward passes are seen in Figures 2 and 3.

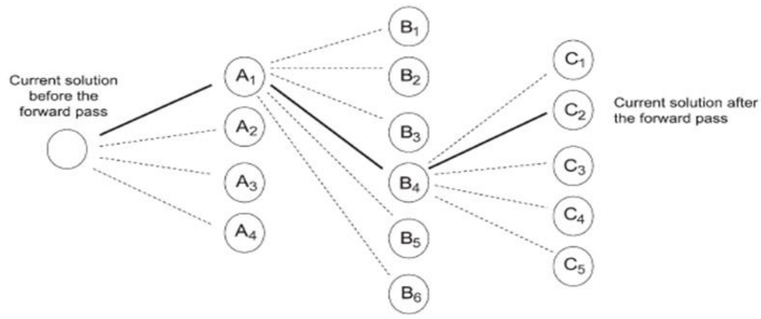


Figure 2. Forward pass phase [25]

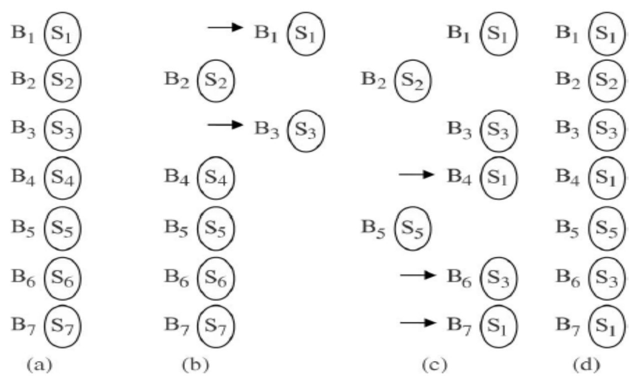


Figure 3. Backward pass phase [25]

To create all possible solutions (one solution per bee), the forward and backward passes are alternated through the process [26]. When all of the solutions have been fulfilled, the best is chosen and used to modify the global best solution, by accomplishing an iteration of the BCO. All solutions are discarded at this moment, then a new iteration can begin. The algorithm repeats the process until a terminating condition is reached. The maximum allowable central processing unit (CPU) time or the max number of forward-backward passes, and so on are examples of possible stopping conditions. Finally, the best-found solution (also known as the global solution) is declared the final solution.

Similarly to dynamic programming, The BCO also works in steps to address combinatorial optimization problems [27], where one optimization variable is used in each of the steps. Let us represent a bounded number of pre-stages by ST = (st1, st2 , ... , stm), where m represents the stages number. The number of bees used in the search process is denoted by B, while the total iterations count is denoted by I, and Sj (j=1, 2,..., m) represents a collection of partial solutions for stage stj [28]. The following steps depicted the BCO algorithm [22]:

```
Step1: beginning(set the bee numbers (B) and the iterations number I, Pick the set of stage
ST=(st1,st2,…….,stm), which represents the number of productive movements during the one
forward pass.
Then, locate any viable solution x of the problem(this solution represents the initial
solution).
Step2: Set i:=1, Execute the next steps until i=I:
Step3: Set j:=1, Repeat the following steps until j=m: ( m is the number of stages)
Forward pass: Let bees leave the hive and generate partial solutions (B) from Sj at stage
stj.
Backward pass: All bees return to the hive. it communicates with one another about the quality
of the partial
 solution, and become an uncommitted follower.
         Expand the same partial solution without attracting nestmates, or dance to recruit
nestmates
         before going back to the generated partial solution.
         Adjust, j:=j+1.
Step4: If the best solution xi is better than the current best solution x, then the current
best solution should be updated ( x:=xi).
Step5: Increment the I, i=i+1.
```

## 3. RESULTS AND DISCUSSION

TSP is a well-known combinatorial optimization problem and is NP-hard in nearly all of its forms. It is addressed by determining the best route across a list of cities, each of which is visited exactly once, then the salesman returns to the initial city with the shortest distance possible. The dilemma of the traveling salesman is merely used to illustrate the properties of the performed algorithms in this study.

Experiments are designed to evaluate the performance of the ABC and BCO algorithms by comparing algorithm implementation in terms of the ability to solve combinatorial problems and algorithm efficiency. The algorithms were implemented using python with JetBrains PyCharm Community Edition 2017.1.4 x64 and executed on a machine that has the Corei3 CPU with a processing speed of 1.5 GHz and 8 GB memory. Initially, the data was produced at random, consisting of several nodes, each representing one city. To keep things simple, nine cities will be chosen. Figures 4 and 5 show the data spread in 3D and 2D space, respectively.

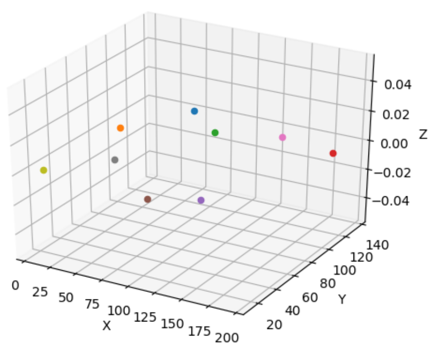


Figure 4. Data distribution in three-dimensional space

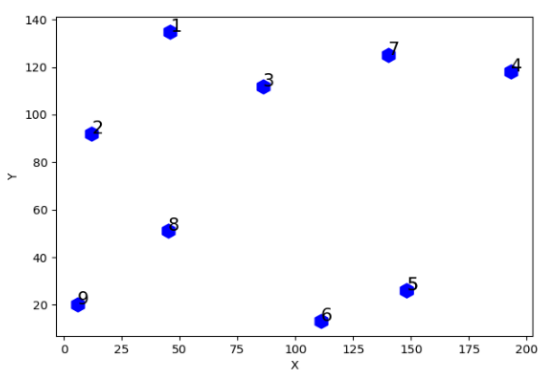


Figure 5. Two-dimensional data distribution

The search space, as it is commonly known, consists of a set of possible solutions. The search space of N cities in the TSP case has (N-1)! potential solutions [29]. That is to say, there will be 40320 different paths to get from one city to another. The ABC and BCO algorithms will be used to discover the shortest possible route that passes to each city exactly once.

The first experiment was performed to determine the optimal path by applying the ABC algorithm, Each algorithm has a set of control parameters that are necessary for it to operate effectively. A control parameter's first value is a maximum number of cycles: in this experiment, the MCN value has taken equal to 1500. The second variable in this experiment is the maximum population size set to be 100. The number of runs of the next parameter was set to 80 for the experiment. It should be remembered that each run has MCN. Dimension is another parameter, and it is dependent on the number of cities. The population is divided into 50% employed bees, 50% onlooker bees, and 20% scout bees. The algorithm found the shortest path through travel in the cities 1-3-7-4-5-6-8-9-2 with distance=549.882 and elapsed time=0:14:38.694266 as shown in Figure 6.

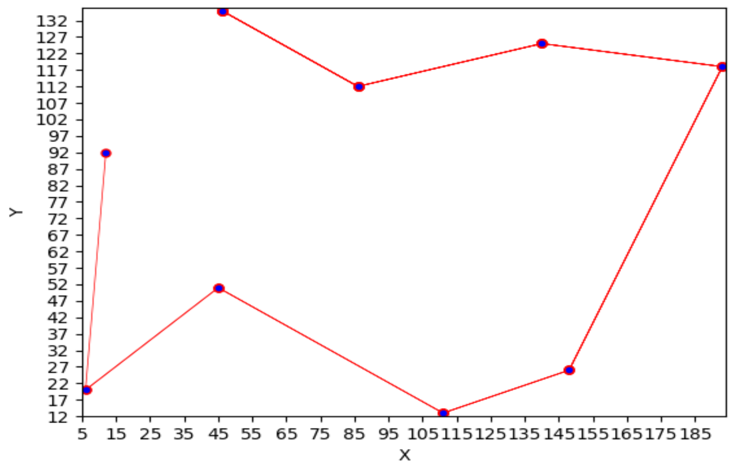


Figure 6. Diagram of the best path with the ABC algorithm

The second experiment is conducted to evaluate the performance of BCO. The performance is compared with the original ABC for solving the TSP problem. The parameters of the algorithm were set as follows: several bees=300, stages number=3, probability of bees to be scout bee=10%. The BCO satisfied the shortest path through travel the cites 4-7-3-1-2-9-8-6-5 with distance=447.4063266633628 and Elapsed time=0:00:26.876552, as shown in Figure 7.

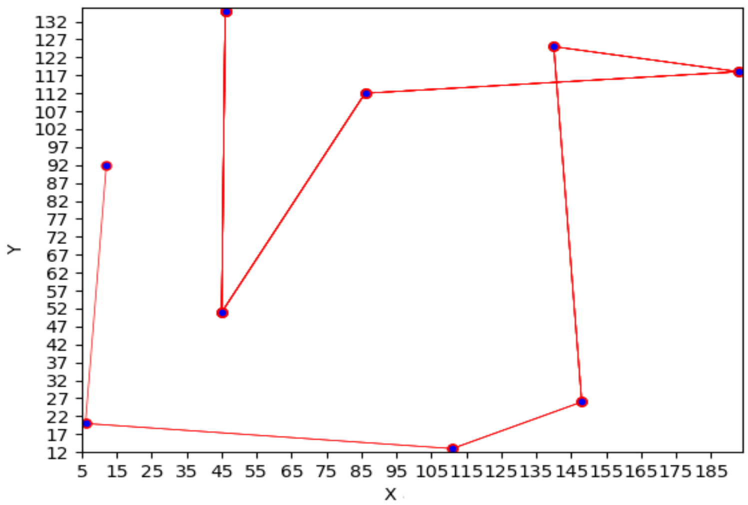


Figure 7. Diagram of the best path with the BCO algorithm

The ABC integrates a local search strategy used by employees and onlookers bees with a global search strategy controlled by onlookers and Scouts. That is, it used a selection process (greedy selection) to obtain the best solution, Consequently, the algorithm often demands a lot of processing resources and time. On the other hand, the BCO strategy is built on cooperation. Artificial bees canFig be better efficient when they work together. The BCO provided high-quality outcomes, where it was able to get values that are extremely near to the optimal value of the objective function. The BCO requires very low time to identify the optimal solution. In other terms, it could produce a good solution in a reasonable computing time.

## 4. CONCLUSION

The purpose of this work is to provide a general understanding of the two well-known bee technologies and to conduct an in-depth analysis of the manner in which they behave. ABC and BCO are two of the most prominent swarm intelligence strategies for obtaining appropriate solutions for optimizing problems in a fair amount of computation time. To achieve this objective, this paper analyzes these technologies' popular variants and their use to solve the TSP problem and locate the better technique for finding the optimal solution. This study found that ABC and BCO algorithms are well-suited for solving combinatorial problems. They successfully solved the TSP problem, where ABC achieved a good solution with a feasible cost. Also, the BCO solved the problem with a shorter distance and less time. The outcomes demonstrate that the BCO method outperforms the ABC algorithm in finding the optimal solution regarding distance and elapsed CPU time. Consequently, the conclusion is that the BCO algorithm can reach significantly better solutions to combinatorial problems.

## REFERENCES

[1] L. Yang, J. Qi, J. Xiao, and X. Yong, "A literature review of UAV 3D path planning," in *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, Jun. 2015, vol. 2015-March, no. March, pp. 2376–2381, doi: 10.1109/WCICA.2014.7053093.

[2] X. S. Yang, "Metaheuristic optimization: Algorithm analysis and open problems," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6630 LNCS, 2011, pp. 21–32.

[3] E.-G. Talbi, *Metaheuristics*. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2009, doi: 10.1007/978-3-319-61756-5_11.

[4] X. S. Yang, "Firefly algorithms for multimodal optimization," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5792 LNCS, pp. 169–178, 2009, doi: 10.1007/978-3-642-04944-6_14.

[5] M. Dorigo, "Optimization, learning and natural algorithms," *Politecnico di Milano*, 1992.

[6] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942–1948, doi: 10.1109/ICNN.1995.488968.

[7] X. S. Yang and S. Deb, "Engineering optimisation by cuckoo search," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 4, pp. 330–343, May 2010, doi: 10.1504/IJMMNO.2010.035430.

[8] H. A. Shabat and N. A. Abbas, "Enhance the performance of independent component analysis for text classification by using particle swarm optimization," in *3rd International Conference on Advanced Science and Engineering, ICOASE 2020*, Dec. 2020, pp. 1–6, doi: 10.1109/ICOASE51841.2020.9436547.

[9] D. Klusáček and V. Chlumský, "Planning and metaheuristic optimization in production job scheduler," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10353 LNCS, 2017, pp. 198–216.

[10] E. G. Talbi, "Machine learning into metaheuristics: a survey and taxonomy," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–32, Jul. 2021, doi: 10.1145/3459664.

[11] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, Aug. 2009, doi: 10.1016/j.amc.2009.03.090.

[12] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," *Technical Report TR06, Erciyes University*, no. TR06, p. 10, 2005, [Online]. Available: http://mf.erciyes.edu.tr/abc/pub/tr06_2005.pdf.

[13] D. Karaboga and B. Akay, "A survey: algorithms simulating bee swarm intelligence," *Artificial Intelligence Review*, vol. 31, no. 1–4, pp. 61–85, Jun. 2009, doi: 10.1007/s10462-009-9127-4.

[14] A. Banharnsakun, T. Achalakul, and B. Sirinaovakul, "Artificial bee colony algorithm on distributed environments," in *Proceedings - 2010 2nd World Congress on Nature and Biologically Inspired Computing, NaBIC 2010*, Dec. 2010, pp. 13–18, doi: 10.1109/NABIC.2010.5716309.

[15] D. Junaedi and N. U. Maulidevi, "Solving curriculum-based course timetabling problem with artificial bee colony algorithm," in *Proceedings - 1st International Conference on Informatics and Computational Intelligence, ICI 2011*, Dec. 2011, pp. 112–117, doi: 10.1109/ICI.2011.28.

[16] M. Boopathi, R. Sujatha, C. Senthil Kumar, and S. Narasimman, "Quantification of software code coverage using artificial bee colony optimization based on markov approach," *Arabian Journal for Science and Engineering*, vol. 42, no. 8, pp. 3503–3519, Aug. 2017, doi: 10.1007/s13369-017-2554-7.

[17] W. Y. Hsu and Y. P. Hu, "Artificial bee colony algorithm for single-trial electroencephalogram analysis," *Clinical EEG and Neuroscience*, vol. 46, no. 2, pp. 119–125, 2015, doi: 10.1177/1550059414538808.

[18] A. Banharnsakun, "Feature point matching based on ABC-NCC algorithm," *Evolving Systems*, vol. 9, no. 1, pp. 71–80, Mar. 2018, doi: 10.1007/s12530-017-9183-y.

[19] A. Banharnsakun, "A MapReduce-based artificial bee colony for large-scale data clustering," *Pattern Recognition Letters*, vol. 93, pp. 78–84, Jul. 2017, doi: 10.1016/j.patrec.2016.07.027.

[20] A. Banharnsakun, T. Achalakul, and B. Sirinaovakul, "The best-so-far selection in artificial bee colony algorithm," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 2888–2901, Mar. 2011, doi: 10.1016/j.asoc.2010.11.025.

[21] K. S. Kaswan, S. Choudhary, and K. Sharma, "Applications of artificial bee colony optimization technique: survey," in *2015 International Conference on Computing for Sustainable Global Development, INDIACom 2015*, 2015, pp. 1660–1664.
[22] M. Dell and D. Teodorovic, "Bee colony optimization - A cooperative learning approach to complex transportation problems," *Advanced OR and AI Methods in Transportation*, pp. 51–60, 2005.
[23] D. Teodorovic, P. Lucic, G. Markovic, and M. D. Orco, "Bee colony optimization: principles and applications," in *2006 8th Seminar on Neural Network Applications in Electrical Engineering*, 2006, pp. 151–156, doi: 10.1109/NEUREL.2006.341200.
[24] R. Forsati, A. Keikha, and M. Shamsfard, "An improved bee colony optimization algorithm with an application to document clustering," *Neurocomputing*, vol. 159, no. 1, pp. 9–26, Jul. 2015, doi: 10.1016/j.neucom.2015.02.048.
[25] N. Todorovic and S. Petrovic, "Bee colony optimization algorithm for nurse rostering," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 2, pp. 467–473, Mar. 2013, doi: 10.1109/TSMCA.2012.2210404.
[26] S. Abualhaija and K.-H. Zimmermann, "D-Bees: A novel method inspired by bee colony optimization for solving word sense disambiguation," *Swarm and Evolutionary Computation*, vol. 27, pp. 188–195, Apr. 2016, doi: 10.1016/j.swevo.2015.12.002.
[27] A. S. Girsang, C. W. Tsai, and C. S. Yang, "A fast bee colony optimization for traveling salesman problem," in *Proceedings - 3rd International Conference on Innovations in Bio-Inspired Computing and Applications, IBICA 2012*, Sep. 2012, pp. 7–12, doi: 10.1109/IBICA.2012.44.
[28] M. Dell'Orco, M. Marinelli, and M. G. Altieri, "Solving the gate assignment problem through the fuzzy bee colony optimization," *Transportation Research Part C: Emerging Technologies*, vol. 80, pp. 424–438, Jul. 2017, doi: 10.1016/j.trc.2017.03.019.
[29] N. A. Ra' *et al.*, "MGA-TSP: modernised genetic algorithm for the travelling salesman problem," *International Journal of Reasoning-based Intelligent Systems*, vol. 11, no. 3, p. 215, 2019, doi: 10.1504/IJRIS.2019.102541.

## BIOGRAPHIES OF AUTHORS

**Hafedh Ali Shabat** received the B.Sc. degree in computer science from the University of Babylon, Iraq, the M.Sc. degree in information technology-AI department from the University of UKM, Malaysia., and the Ph.D. degree in information technology with specialization in text mining from the University of Babylon, Iraq. His research areas are text mining, classification, machine learning, named entity recognition, and optimization. he used to hold an administrative post with the technical Institute of Babylon, AL-Furat AL-Awsat Technical University (ATU) director of the Centre of the computer. He can be contacted at email: h.ali@atu.edu.iq.

**Khamael Raqim Raheem** received the B.Sc. degree in computer science from the University of Babylon, Iraq, the M.Sc. degree in information technology-AI department from the University of UKM, Malaysia., and the Ph.D. degree in information technology with specialization in image processing from the University of babylon, Iraq. Her research areas are text mining, image processing, named entity recognition, and recommendation system. She can be contacted at email: khmrakrah@atu.edu.iq.