Received: February 17, 2024. Revised: April 20, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Apiary Organizational-Based Optimization Algorithm: A New Nature-Inspired Metaheuristic Algorithm

Mais A. Al-Sharqi[1]* **Ahmed T. Sadiq Al-Obaidi[2]** **Safaa O. Al-mamory[3]**

[1]*Informatics Institute for Postgraduate Studies,*
Iraqi Commission for Computers and Informatics, Baghdad, Iraq
[2]*Computer Science Department, University of Technology, Baghdad, Iraq*
[3]*College of Information Technology, University of Babylon, Iraq*
* Corresponding author's Email: Phd202110697@iips.edu.iq

Abstract: The pursuit of efficient solutions in optimization has led to the development of metaheuristic algorithms (MHAs). Some of these algorithms adopt the behaviour of several different animals and insects. However, researchers still attempt to improve the performance of such algorithms. This study proposes a new metaheuristic optimization algorithm which is so-called the Apiary Organizational-Based Optimization Algorithm (AOOA). The proposed algorithm introduced a new concept of multiple populations inspired by the organizational behaviour of honeybees inside the apiary. Honeybees are a highly systematic and organized society that lives within an apiary and achieves specific tasks during their lifecycle. To develop the proposed algorithm, the key activities of the queen, drones, and workers inside the apiary are determined first. These activities are translated into several different phases to develop a mathematical model that represents the ground of the proposed AOOA. 23 classical unimodal, multimodal, and fixed-dimension benchmark functions are used to verify AOOA performance. The results are compared with 6 recent MHAs, puzzle optimization algorithm (POA), coati optimization algorithm (COA), Average and Subtraction-Based Optimizer (ASBO), guided pelican algorithm (GPA), Golden Search Optimization Algorithm (GSO) and extended stochastic coati optimizer (ESCO), showing that AOOA outperforms them in solving 21, 18, 20, 18, 21 and 15 functions, respectively. AOOA was superior in mean fitness of 17 out of 23 functions with superiority of 86%, 100%, and 50% of unimodal, multimodal, and fixed-dimension functions, respectively indicating the competitiveness of the proposed algorithm in providing more appropriate solutions.

Keywords: Apiary organizational-based optimization algorithm, AOOA, Nature-inspired, Metaheuristic, Optimization algorithm.

1. Introduction

All Metaheuristic algorithms (MHAs) are high-level problem-solving methods to find optimal or near-optimal optimization and search problem solutions. Nature is the origin of inspiration for many metaheuristic algorithms. This is why it is referred to such methods as nature-inspired algorithms. MHAs are cutting-edge techniques that employ the tradeoff of randomization with local search to find the promised solutions, tending to be more suitable for global optimization[1, 2]. MHAs can be applied to a wide range of problems in several cases such as finding the optimal solution when it is impractical or computationally infeasible. The key two elements of any metaheuristic algorithm are exploitation and exploration [1-4]. Exploration means investigating the search space on a global scale, whereas exploitation refers to concentrating on a local region using information on reasonable solutions found in a particular region. Accordingly, such algorithms are applied in several different scenarios.

MHAs achieved high performance in various applications, especially for complex issues such as NP-hard, combinatorial optimization, continuous optimization, scheduling, multi-objective optimization, and routing problems. This is due to

Received: February 17, 2024. Revised: April 20, 2024.

their extraordinary capability to select ideal features and eliminate redundant and irrelevant characteristics [5]. Flexibility and easy implementation are the main reasons behind using MHAs in different situations. These algorithms do not rely on gradient information of the objective landscape. Instead, they exploit natural operations such as evolution, swarm behaviour, and physical annealing, taking advantage of stochasticity and adaptability. MHAs investigate the solution space of a problem by repeatedly revising and enhancing alternative solutions until a suitable or near-optimal solution is discovered [6].

Researchers and practitioners frequently investigate different algorithms to identify the most efficient solution for a specific problem. However, such algorithms vary in their overall performance on NP-hard problems according to the no-free lunch theory [7] and this, in turn, leads to proposing either new algorithms or modifying available techniques. This research proposes a new metaheuristic algorithm based on the behaviour of bees which have one of the most intelligent and organized societies. This is achieved by investigating bees' behaviour and activities during the lifecycle inside the apiary. The rationale behind proposing this algorithm is that for NP-hard optimization problems, according to the "no free lunch" (NFL) theorems, no one optimization algorithm can solve all the optimization problems. Therefore, there is room for developing new algorithms to enhance the quality of solutions. The key contributions of this research are threefold:

1. The proposed algorithm simulates natural organizational behaviour and the biological process of bees.
2. The main activities of the queen, drones, and workers inside the apiary are determined and translated into seven different phases to develop a mathematical model.
3. The AOOA reduced the effort required for parameter setting and tuning. All we need during executions is to change the number of hives (represented by h) and (number of bees represented by b) to try different population sizes.
4. AOOA was verified and validated via 23 benchmark functions, seven high-dimension unimodal, six high-dimension multimodal and ten fixed-dimension multimodal functions.
5. The AOOA's results were compared with six recent competing MHAs to prove its superiority.

The rest of this research is structured as follows. In Section 2, the related works are reviewed. The apiary organizational behaviour and bees' lifecycle are discussed in Section 3. Section 4 explains the proposed Apiary algorithm, presents its pseudo-code and shows the experimental results. Finally, Section 5 concludes the key points of this work and highlights its possible future directions.

2. Related work

As the complexity of contemporary problems increased, researchers became more interested in MHAs because they can solve complex problems. This led to the introduction of many MHAs which are based on different techniques inspired by our life. Despite the diversity of the basis on which MHAs are based, they all share some characteristics as all of them are population-based which means they deal with individuals of one population within the search space. This population is guided through one or two behaviours (mostly food foraging or prey hunting) and new solutions are generated depending on one mechanism (mostly updating the current position of the individual according to some position better than the current one) to develop solutions.

For example, the Guided Pelican Algorithm (GPA) [8], an MHA that improved the shortcoming of POA. This algorithm mimics the behaviour of the pelican during hunting the prey. Like POA, GPA consists of one population with some individuals that search for optimal solutions within their environment. Each individual is assessed to determine its quality. GPA comprises two phases: the guided movement phase and the randomized movement phase. The evolution of solutions comes from generating a set of candidates, selecting the best candidate of current candidates, and comparing the best candidate to the global best previously selected (instead of a randomized target in POA) which represents the target that all individuals pursue.

Another algorithm is the Coati Optimization Algorithm (COA) [9]. COA is an MHA that simulates two behaviours of coatis: i) attacking and hunting iguanas which represents the exploration phase and ii) escaping from predators which corresponds to the exploitation phase. COA consists of one population of coatis separated into two groups, half climb the tree to scare the iguana and a half wait under the tree to catch it when it falls. Coatis continuously update their positions according to the position of the best member (the prey) to be nearer or to the position of the predator to be saver. Although COA mimics two different behaviours, they both depend on a single piece of information which is the position.

Kusuma & Dinimaharawati proposed the Extended Stochastic Coati Optimizer (ESCO) [10] which is the expansion of the shortcoming coati optimization algorithm (COA). They expanded COA and increased its stochasticity. This was done by expanding two sequential phases into three sequential phases. ESCO implements guided, random, and neighbourhood searches within one population. In addition, ESCO used three references in its guided searches: the global best unit (iguana on the tree), a randomized unit within the search space (iguana on the ground), and a randomly selected unit. ESCO segregate roles of coatis stochastically in all phases rather than in initial phases as in the COA. Finally, the best global solution is introduced as the final solution.

Noroozi M. et al. [11] utilized the advantages of Particle Swarm Optimization (PSO) and Sine Cosine Algorithms (SCA) to introduce a new hybrid metaheuristic algorithm called Golden Search Optimizer (GSO). GSO has one population and three main phases: initialization, evaluation, and updating the current population. New solutions were generated by sorting the individuals according to their fitness and replacing the worst one with a randomly selected one. To avoid premature convergence from which PSO suffers, GSO used a step size parameter to update the position of the population's individuals.

Average and Subtraction-Based Optimizer (ASBO) [12] is a new metaheuristic optimization algorithm that employs the average information and the subtraction of the best and worst population individuals to guide the population in the problem search space. ASBO comprises one population initiated with some individuals that are randomly distributed on the search region. Each individual was provided with information about its position that it shared with other individuals in the population. Iteratively, each individual up its position based on the average best and worst position which would guarantee keeping it away from local optima.

The Puzzle Optimization Algorithm (POA) [13] is a game-based MHA. Each member of the population is considered a puzzle and has points obtained from the number of pieces in their right place. Updating the population of individuals was done through two stages. Every member of the population is updated in the first stage according to the information of other members. In the second phase, every individual in the population attempts to finish their puzzle using pieces that other individuals have recommended. The final positions of puzzles after the last iteration represent the best sub-optimal solution of POA.

In contrast with all previously outlined works which deal with individuals of one population with one or two behaviours, AOOA simulates multiple populations within many colonies and several behaviours by introducing the concept of "Apiary". To the best of the researchers' knowledge, there is no such research present in the literature and this is the first time to propose the concept of multiple populations with multiple beehives and several bees' organizational behaviours.

AOOA consists of several phases corresponding to the main activities of honeybees during their lifecycle. Like all MHAs, AOOA has two main search approaches utilized to optimise the randomly generated solutions, exploitation and exploration. Drone exchange and different tasks according to different workers' age stages are examples of exploration. On the other hand, exploitation and solutions development are done through more than one area. The first area comes from the queen fertilization process, and the second comes from workers' lifecycle activities. AOOA explored the best solutions in multiple populations within multiple search regions by providing multiple hives each having its independent individuals which reduce the time required to find the best solutions. In the next section, the concept, biological basis and organizational behaviour of honeybees inside the apiary were explained.

3. Apiary organizational behaviour

Honeybees are the most studied and best-understood social insects on Earth [14]. The bee colony has many secrets that have not yet been revealed [15]. Bees inside the colony carry out many tasks, each according to their assigned role [16]. Scientists shed light on parts of the individual behaviour in the bee colony such as foraging and mating, but there is still a scarcity of research on other behaviours. It is not a secret that these creatures are of unparalleled intelligence and organization. Bees live in highly systematic, cooperating, and complex societies. One of the most critical aspects of bees' behaviour is their social organization within beehives in the apiaries [17, 18]. Accordingly, such intelligent behaviours can be simulated to solve contemporary and complex optimization problems such as the NP-hard problem [14]. Here, some definitions should be highlighted first.

Definition 1: the apiary is the place where a collection of beehives is grouped and managed where each apiary consists of several beehives.

Definition 2: Bee Colony is a group of bees that live together.

Definition 3: the beehive is the home of a bee colony. Each beehive consists of many members that manage the beehive, including a queen, workers, and drones.

The behaviour of bees within each apiary hive is highly coordinated and ensures that the whole colony

Received: February 17, 2024. Revised: April 20, 2024.

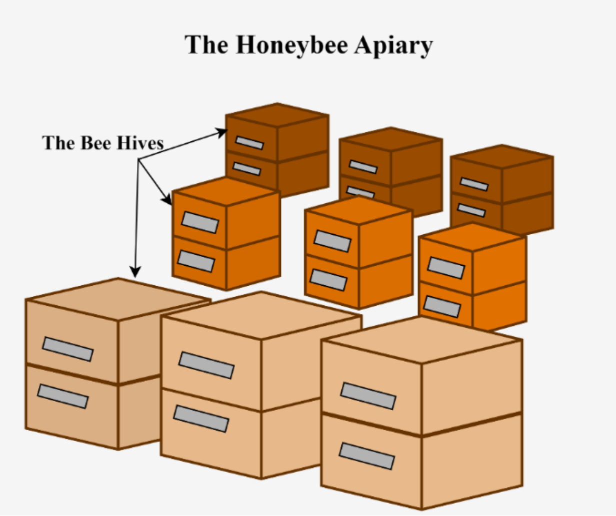


Figure. 1 A honeybee apiary

survives. The beehive is a highly marshalled and complicated structure that serves as the centre of activity for bees [14, 19]. Fig. 1 shows a honeybee apiary.

3.1 The importance of bees' organizational behavior inside the apiary

Many factors make the organizational behaviour of bees inside an apiary extremely important such as the location where bees are housed. Other significant factors are [14-18, 20-23]:

1. *Effective Hive Functioning:* The behaviour of individual bees within a colony or hive affects the general efficacy and functioning of the hive. Foraging, nursing, guarding, and reproducing are among many unique roles and tasks of bees. The survival and productivity of the colony depend on these activities being correctly done, which is ensured by effective organizational behaviour.
2. *Resource Allocation:* Bees have to divide up resources such as water, pollen, and nectar within the hive efficiently. Organizational behaviour specifies the resource gaining, saving, and distribution among bees, guaranteeing that the colony's food supply is sufficient.
3. *Interaction and Collaboration:* Bees exchange information about the position of food sources through a sophisticated communication system.
4. *Beehive Security and Survival:* Keeping a beehive safe and away from enemies and predators is deeply related to performing effective behaviour. Guard bees are responsible for hive security as they cooperate to maintain hive protection.
5. *Bees breeding and evolution:* This process produces new queens, drones, and workers as needed. The hive expansion affects the hive size and lifetime, so the behaviour of worker bees specifies the timing and effectiveness of the generation process.
6. *Bees Social Hierarchy:* Each bee in the beehive has a role. Such roles differ from queen to worker and drones, and from worker to another according to their age. This hierarchical structure imposes specific behaviour for each bee, which collectively integrates the whole colony's activities.
7. *Acclimation to Environment:* Organizational behaviour provides bees flexibility to adapt their behaviour according to environmental changes. For instance, bees may build new combs and swarm in response to the size increment of bees, or they may perform some activities relevant to changes in food source shortage or abundance.
8. *Hive Health and Extinction:* Beehive continuity is highly dependent on its organizational behaviour. Unhealthy and disorganized hives are more susceptible to illness and extinction [24, 25]. The prosperity of the overall colony and the task coordination of bees depend closely on the organizational behaviour of bees in an apiary.

3.2 The life cycle of the queen, workers, and drones

The population of bees consists of a queen, workers, and drones. Drones only appear in the fertilization season. As in all insects that undergo complete metamorphosis, each bee passes through the egg, larval, pupal, and adult stages. Depending on the egg, a newborn bee is either a worker or a drone [14, 16, 26-28]. A closer inspection of the bee community is provided as follows:

1. **The Queen:** is the most crucial member of a colony. It is responsible for laying eggs and emitting pheromones that regulate the behaviour of other bees. The behaviour of a queen inside the hive is simulated in the proposed algorithm steps.
   a) *Initiate the Bee Colony:* During its life, the queen lays about one million eggs, averaging 1500-2500 eggs daily. The worker lives 40 to 60 days, while the queen lives about seven years or less. The same egg can become a queen or a worker. The difference is in nutrition, where the worker is fed bee bread, while the queen is fed royal food, where the workers feed her. The

queen emerges from the hexagonal eye after 16 days [3].

b) *Queen Replacement:* there are several reasons behind queen replacement. One of these reasons is queen illness. Another reason is the accidental killing of the queen by beekeepers. Lack of egg production also required queen replacement. The crowded hive needs to be split; therefore, it needs a new queen to replace the old which will transfer to the new nest. A colony without a queen is more vulnerable to raids by other colonies or infestations by parasitic swarms that take over the colony [22, 26].

c) *Swarming Phenomenon:* spring is the bloom and food abundance season. It is also the most prolific breeding time. The fast growth of the larvae and the queen's assiduous egg-laying leads to a quick increase in bees. As a result, the beehive will be crowded. However, the number of colonies must also multiply since a colony may perish through disease, famine following a lousy summer, or other misfortune. Bee species would quickly become extinct if new colonies weren't established to offset this loss. Every new colony needs its queen, and a colony can only reproduce after receiving another queen. When a colony multiplies itself, the "swarming phenomenon" of the bees happens. The colony swarms about a week before the first young queen leaves her cell. If the queen is strong and can lead a giant hive, the hive expands horizontally or vertically. Still, if it cannot do so, the hive must split, and then the bees raise new queens. This process requires producing males previously to fertilize the new queens; otherwise, the reproduction process is useless [4].

d) *Queen Fertilization:* When the queen is ready to mate, she leaves the hive and flies to an area where drones from other colonies follow her. Drones compete to mate with the queen, and the winning drones transfer their sperm to the queen during mating. The queen stores the sperm in her spermatheca that can remain viable for up to several years. The queen's fertilization determines the sex of the eggs she lays, whether she develops into a female worker, a new queen, or a male drone.

2. **Workers** are female bees that comprise most of the colony. They make up more than 90% of the bee population. They come out of the hexagonal eye after 21 days. Workers go through several lifecycle stages and translocate accordingly. They perform several different tasks according to each lifecycle stage [18, 29].

    a) *Cleaning:* During the first few days of their life as adults, worker bees clean the hive cells, and look after the brood. They also collect water to help regulate the temperature and humidity.

    b) *Nursing:* After about one week, worker bees begin caring for the young larvae, feeding them pollen and nectar. They also cap the cells of the brood comb with beeswax to protect the developing larvae.

    c) Building Comb: Worker bees also produce beeswax, which they use to build the hive comb. They secrete the beeswax from glands in their abdomen and use it to create the hexagonal cells of the comb.

    d) *Guarding:* As worker bees grow up, they protect the hive by inspecting incoming bees and defending the hive against predators and other threats.

    e) *Foraging:* At about the end of their lifecycle, worker bees become foragers, leaving the hive to collect nectar, pollen from flowers, and water.

    f) *Death:* The worker bee's life span varies depending on several factors, including the time of year and the demands of the colony. In general, worker bees live for about six weeks during the summer months but can live for several months during the winter when the demands on the colony are lower.

3. **Drones** are male bees responsible for mating with the queen. They do not have stingers and do not gather food or care for the young. Drones are only produced during certain times of the year when the queen bee needs to mate [20, 23] Drones make up about 10% of the bee population during the spring season. They fly to distances of up to 30 kilometers and contribute to the preservation of cell temperature. The drone comes out of the hexagonal eye after 24 days. According to [19, 30], it takes five to ten days for them to develop the ability to fly and ten to fourteen days to reach sexual maturity and become capable of fertilization. The worker-to-drone ratio in a beehive exhibits significant variability. The outcome can vary based on multiple variables, such as the season, the magnitude of the colony, and the accessibility of nourishment and assets. Nevertheless, a thriving bee colony often maintains a worker-to-drone

Received: February 17, 2024. Revised: April 20, 2024.

Figure. 2 The queen, drones, and workers

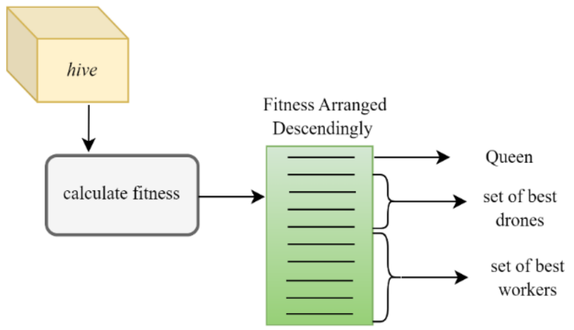


Figure. 3 Fitness calculation and selecting queen, drones, and workers

ratio of at least 10:1, indicating a significantly larger population of worker bees in comparison with drones [27]. This ratio highly affects the bees' colony efficiency because worker bees play a fundamental role in supporting the colony system, providing sustenance, and maintaining alternative food resources, whereas drones are privileged for the colony and are specifically generated during the fertilization period. If there is a shortage of food, the colony decreases drone generation since they are not necessary for colony survival.

a) *Drone exchange:* Drones play a crucial role in honeybee reproduction and the maintenance of genetic variety. Upon being liberated from their hives, drones tend to navigate towards regions where additional hives are situated, irrespective of whether those hives belong to them or not. This flight can result in drones from one hive mating with queens from another, leading to genetic mixing and diversity within the population. Drone exchange can also help prevent inbreeding within a colony, which can lead to congenital abnormalities and bad colony health. Introducing new drones from other hives reduces the chance of inbreeding, ensuring a healthier and more diverse population of bees. Fig. 2 shows the queen, drone, and worker.

b) *Fading out:* During fertilization, drones attempt to reach and mate with the queen. Only potent and high-fitness drones succeed in this task, whereas the rest perish. Most drone bees can mate more than once, but in some cases, the drone genitalia, or at least the end phallus, is torn away in mating. This may lead immediately to the drone dying after mating.

4. The proposed apiary organizational-based optimization algorithm (AOOA)

A beehive comprises several components that collectively create the perfect environment for bees to live and thrive. By examining the intelligent behaviour of bees inside the apiary, it was found that seven main stages can be converted into a nature-inspired metaheuristic algorithm that simulates the organizational behaviour of honeybees inside their apiary. Hence, AOOA is proposed which includes several phases.

4.1 AOOA phases

By investigating beehive members and their organizational behaviour inside the apiary, it can be concluded that the essential activities can form seven phases of the proposed algorithm, including:

1. *Initial population:* this phase simulates the behaviour of the queen. Her behaviour encompasses initiating *the Bee Colony* where beehive members are represented mathematically. For each apiary, there is a population of individuals that are randomly generated and can be represented by (*N*). This includes the number of hives, which is represented by (*h*), each with many bees (*b*). The number of bees is suggested to be between 10 and 100 to ensure no increase in the problem size and time complexity. A hive with ten bees represents the simplest hive that consists of one queen, one drone, and eight workers. Eq. (1) shows the relationship between *N*, *h*, and *b*.

$$N = h \, x \, b \tag{1}$$

Where $1 \leq h \leq m$, $m \in Z^{+}$ and $10 \leq b \leq 100$
The population consists of one unique queen (Q), which is selected as the best solution, and several drones (d) with a ratio of either 10% or 20% of bees as well as the number of workers (w) to be from 80% to 90% of bees (biologically inspired ratios). Each individual in the apiary population has a quality measurement (i.e.,

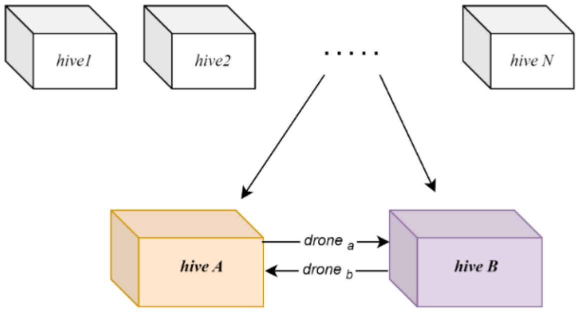


Figure. 4 The exchange of two drones between two randomly selected hives

Table 1. The main parameters of AOOA

Parameter	Description	Value
h	Number of hives	1, 2, …, *m*
b	Number of bees in each hive	integer number between 10-100
N	Population size	*h x b*
D	Drones' ratio in each hive	10% - 20% of *b*
W	Worker's ratio in each hive	90% - 80% of *b*
$fert_{ratio}$	Ratio of best-selected drones to fertilize the queen and produce new individuals	20% - 40% of *d*
fd_{ratio}	Number of bees that fade out each time	10% - 30% of *b*
S_{ratio}	The swarming ratio determines when the hive splits	integer number

fitness) which is calculated by the fitness function. Fig. 3 exhibits the fitness calculation process.

2. *Drones Exchange:* this phase simulates the behaviour of drones. Mathematically, drone exchange is performed by choosing two hives a, b and drones $d1, d2$ randomly, and then performing a swap operation. Fig. 4 shows the process of drone exchange.
3. *Queen Fertilization and Bees Breeding:* this phase simulates the fertilization behaviour of the queen. Mathematically, each hive has its queen. Queen fertilization generates new individuals by selecting a queen (individual with the highest fitness) and a set of best drones. Each drone fertilizes the queen to produce K new bees. Fertilization ratio ($fert_{ratio}$) is a control parameter that is added to control the number of drones to generate new individuals. The value of $fert_{ratio}$ is assumed to be between 20% and 40% of drones. Biologically, only robust drones can fertilize the queen based on this ratio. If the difference between the queen's features and the fertilized drone is less than or equal to half of the drone's features, the newly generated bees will take most of their features from the queen and less from the fertilized drone. Otherwise, the newly generated bees will take half of their features from the queen and the other half from the fertilized drone. Eq. (2) shows the fertilization process.

$$New\_b_{H_i} = \begin{cases} Q_{H_i} + r.(d_{H_i} - Q_{H_i}), & if \left(\frac{d_{H_i} - Q_{H_i}}{d_{H_i}}\right) < 0.5 \\ {}^{1}/_{2}\,(Q_{H_i}) + r.\ {}^{1}/_{2}\,(d_{H_i}), & O.W \end{cases} \quad (2)$$

where Q_{H_i}, d_{H_i} represent the queen and drone of the i^th^ hive, respectively. While *r* represents a random number within [0,1).

4. *Worker Lifecycle:* this phase simulates the behaviour of workers. Mathematically, each worker has an age representing the probability of changing an existing bee to produce a new one. Biologically, the lifecycle of workers ranges from one day to a maximum of sixty days. The worker's life cycle has been represented as the period of the worker's age in days.
1 to 10: applying simple translocation on the worker (i),
11 to 25: applying 2-opt worker (i),
26 to 50: applying 3-opt worker (i),
51 to 60: applying 2-opt worker (i),
Otherwise, dropping worker (i).
According to Eq. (3), each period performs a specific number of translocations on the selected worker to produce a new bee:

$$NW_{H_i} = W_{H_i} + r \cdot \left(\frac{W_{H_i}^{\max\_g} - W_{H_i}^{g}}{W_{H_i}^{\max\_g}}\right) \times W_{H_i} \quad (3)$$

where W_{H_i} is the worker of the ith hive, *r* is a random number within [0,1), and w^{max_g} is the maximum age of the worker.

5. *Queen Investiture:* this phase simulates the replacement behaviour of the queen. Mathematically, if the newly generated bee has fitness better than the queen, it will be selected

to be the new queen, whereas the old one will die. A threshold, Θ, is selected to compare the fitness of the current queen with the newly generated worker bee.

6. *Fading out:* this phase simulates the fading out behaviour of drones. Mathematically, a control parameter called fd_{ratio} is added to control the number of dying bees. The value of fd_{ratio} is assumed to be between 10% and 30% of bees (a nature-inspired ratio).
7. *Swarming:* This phase simulates another behaviour of the queen which is so-called the swarming phenomenon. Mathematically, a control parameter called Sratio is added to control the separation process of hives. Sratio is assumed to be user-defined and specified according to each problem. Moreover, a counter, Inc_val, is added to be compared with S_{ratio}. Table 1 shows the main parameters of the proposed AOOA.

4.2 AOOA Pseudo Code

Algorithm 1 shows the proposed AOOA pseudo-code which comprises seven phases simulating bees' organizational behaviour within the apiary.

algorithm 1: AOOA

 # Initialization:

1 Generate N solutions using Eq. (1)
2 Compute *f* of solutions.
3 Select Q_i
4 Divide the remaining solutions into two (*w*) and (*d*)
5 Set w^g to 1
6 While (not termination condition) do

 # Drone Exchange:

7 For $i = 1$ to N do
8 For $j = 1$ to $exch_{ratio}$ do
9 Select two hives h_a and h_b, randomly
10 Select two (*d*)s randomly, d_1 from h_a and d_2 from h_b
11 $h_a \leftarrow d_2$
12 $h_b \leftarrow d_1$
13 End for

 # Fertilization and Bees Breeding:

14 For $m = 1$ to *fert_ratio* do
15 Select d_{best} and w_{best} in h_i
16 Fertilize Q_i with d_{best} using Eq. (2) to generate several *new_b*(s)

 # Worker Lifecycle:

17 Fertilize Q_i with w_{best} using Eq. (3) to generate several *new_b*(s)
18 Case Age of w^g do
19 1 to 10: apply simple fertilization on w^g
20 11 to 25: apply 2-optima fertilization w^g
21 26 to 50: apply 3-optima fertilization w^g
22 51 to 60: apply 2-optima fertilization w^g
23 Otherwise: drop w^g
24 End case
25 End for

 # Queen Investiture

26 If ($f(new\_b) - f(Q_i) \geq \Theta$) OR (no convergence to the optimal solution in h_i), then
27 Drop Q_i and go to step 14
28 End if

 # Fading out

29 Drop d_{worst} and w_{worst} from h_i

 # Swarming

30 $h=0$
31 If Size of $h_i \geq S_{ratio}$ then:
32 Split h_i into two hives as below:
33 ½ w_{h_i} remain in h_i, the second ½ w_{h_i} will be transferred to the new hive *h(n+1)*
34 ½ d_{h_i} remain in h_i, the second ½ d_{h_i} will be transferred to the new hive *h(n+1)*
35 Q_{h_i} remains, go to step 14 to generate the queen of new hive *h(n+1)*
36 End if
37 $h=h+1;$
38 $N=N+h;$
39 End for
40 End While
41 End.

Below is a list of symbols used in algorithm 1

f	objective function
Q_i	best solution selected as the Queen
w	worker
d	drone
w^g	the age of the worker
$exch_{ratio}$	number of drones to exchange
Θ	the threshold of fitness
d_{best}	best drone
w_{best}	best worker
New_b	new generated bee
d_{worst}	drone with the worst fitness
w_{worst}	worker with the worst fitness
termination condition	either find the goal or reach max iteration)

4.3 The relationship between the main parameters of AOOA

The relationships among parameters of AOOA are either proportional, inversely proportional, or indirect. The first parameter is the number of hives in the apiary h. By increasing (h), the number of bees (b) will also increase, and consequently, the populations N, d, and w will increase, while $Fert_{ratio}$ and S_{ratio} will not be affected. Moreover, the number of fading-out bees increases by the boosting of b, as shown in Eq. (4).

$$no. of\ fading\ bees = Fd_{ratio}\ x\ b \qquad (4)$$

Where Fd_{ratio} is between $10\% - 30\%$. As the population size (N) increases, bees' resort to increasing the number of hives (h) via the swarming phenomenon. The increment in drones (d) leads to a growth in bees, and population size ($Fert_{ratio}$, and Fd_{ratio}). In contrast, the number of workers goes down because the apiary population (AP) comprises one queen, about 10% drones, and 90% workers as shown in Eq. (5).

$$AP = Q + 0.1\ x\ d + 0.9\ x\ w \qquad (5)$$

The workers' increment also implies boosting the number of bees, population size, and $fert_{ratio}$ and this, in turn, leads to reduce the number of drones. As $fert_{ratio}$ increases, both the number of bees and the population size increase accordingly. Raising Fd_{ratio} can decrease the bees' number, population size, drones, and workers. Finally, an increment in the S_{ratio} will affect the number of apiary hives positively.

Table 2. The relationship among AOOA parameter

	h	***b***	***N***	***d***	***w***	$Fert_{ratio}$	Fd_{ratio}	S_{ratio}
H	–	∝	∝	∝	∝	–	–	–
B	–	–	∝	∝	∝	–	∝	–
N	∝	–	–	–	–	–	–	–
D	–	∝	∝	–	κ	∝	∝	–
W	–	∝	∝	κ	–	∝	–	–
$Fert_{ratio}$	–	∝	∝	–	–	–	–	–
Fd_{ratio}	–	κ	κ	κ	κ	–	–	–
S_{ratio}	∝	–	–	–	–	–	–	–

$\propto$= directly proportional, κ = inversely proportional, $-$ = $indirectly\ affected\ or\ not\ affected\ at\ all$

The relationship among these parameters is illustrated in Table 2.

5. Experiments and results

5.1 Benchmark AOOA with competing algorithms

To validate the performance of the AOOA, a set of 23 benchmark classical functions was used from the literature [11, 31]. The functions are all of the global minima and classified into three main groups, 7 unimodal high-dimension test functions which were used to assess the local searchability around promising solutions (exploitation ability) and convergence rate, 6 multimodal high-dimension test functions with one global optima and multiple local optima which were employed to assess the proposed AOOA against diversification and exploration ability to avoid local optima, and 10 fixed dimension multimodal functions. The dimensions delineate the search space's area, so when the dimensionality increases, the size of the search space expands exponentially, resulting in a greater quantity of suboptimal sites. Moreover, the majority of metaheuristic algorithms exhibit their highest performance when they are applied to such functions. Actual performance evaluations, on the other hand, need functions to be highly dimensional. By reviewing earlier literature, the commonly used setting of dimensionality is 30 [32, 33]. The functions, range, dimension, and optimal solution (f_{min}) are presented in Tables 3-5.

Population initialization is the initial step of AOOA which comprises the random allocation of K bees to each hive. Each bee is depicted as a single row encompassing N columns.

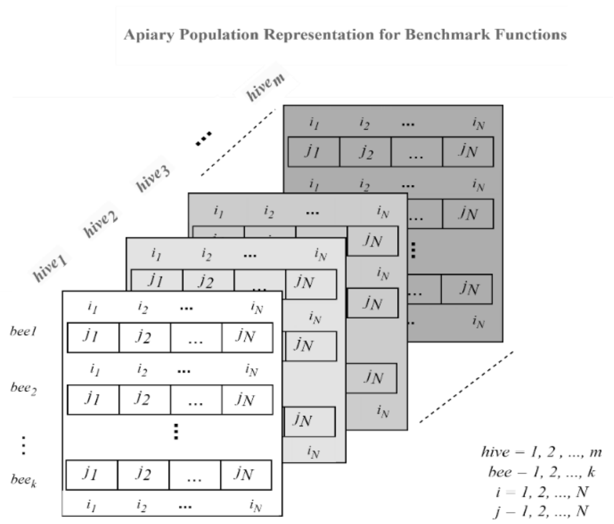


Figure. 5 The Representation of the Apiary Population in the Benchmark Functions

Received: February 17, 2024. Revised: April 20, 2024.

Table 3. The characteristics of the unimodal benchmark functions

Function	**Range**	**Dim**	f_{min}
$F_1(X) = \sum_{i=1}^{D} x_i^2$	[-100, 100]	30	0
$F_2(X) = \sum_{i=1}^{D} \lvert x_i \rvert + \prod_{i=1}^{D} \lvert x_i \rvert$	[-10, 10]	30	0
$F_3(X) = \sum_{i=1}^{D} \left(\sum_{j=1}^{i} x_j \right)^2$	[-100, 100]	30	0
$F_4(X) = max\{\lvert x_i \rvert, 1 \le i \le D$	[-100, 100]	30	0
$F_5(X) = \sum_{i=1}^{D} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	[-30, 30]	30	0
$F_6(X) = \sum_{i=1}^{D} ([x_i + 0.5])^2$	[-100, 100]	30	0
$F_7(X) = \sum_{i=1}^{D} ix_i^4 + random(0,1)$	[-1.28, 1.28]	30	0

Table 4. The characteristics of the multimodal benchmark functions

Function	**Range**	**Dim**	f_{min}
$F_8(X) = \sum_{i=1}^{D} -x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)$	[-500, 500]	30	-12569
$F_9(X) = \sum_{i=1}^{D} [x_i^2 - 10 \cos 2\pi x_i + 10]$	[-5.12, 5.12]	30	0
$F_{10}(X) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^{D} x^2}\right) - exp\left(\frac{1}{D} \sum_{i=1}^{D} \cos 2\pi x_i\right) + 20 + e$	[-32, 32]	30	0
$F_{11}(X) = \frac{1}{4000} \sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[-600, 600]	30	0
$F_{12}(X) = \frac{\pi}{D} \left\{ 10sin(\pi y_1) + \sum_{i=1}^{D-1} (y_i - 1)^2 [1 + 10sin^2(\pi y_{i+1})] + (y_D - 1)^2 \right\} + \sum_{i=1}^{D} u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_{i+4}}{4} u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	[-50, 50]	30	0
$F_{13}(X) = 0.1 \left\{ sin^2(3\pi x_1) + \sum_{i=1}^{D} (x_i - 1)^2 [1 + sin^2(3\pi x_{i+1})] + (y_D - 1)^2 [1 + sin^2(2\pi x_D)] \right\} + \sum_{i=1}^{D} u(x_i, 5, 100, 4)$	[-50, 50]	30	0

Table 5. The characteristics of the fixed-dimension benchmark functions

Function	Range	Dim	f_{min}
$F_{14}(X) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + (x_i - a_{ij})^6}\right)^{-1}$	$[-65.53, 65.53]^2$	2	1
$F_{15}(X) = \sum_{i=1}^{11} \left[a - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}\right]^2$	$[-5, 5]^4$	4	0.00030
$F_{16}(X) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	$[-5, 5]^2$	2	-1.0316
$F_{17}(X) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)cos x_1 + 10$	$[-5, 5]^2$	2	0.398
$F_{18}(X) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 - 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	$[-2, 2]^2$	2	3
$F_{19}(X) = -\sum_{i=1}^{4} c_i exp\left(-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})\right)^2$	$[1, 3]^3$	3	-3.86
$F_{20}(X) = -\sum_{i=1}^{4} c_i exp\left(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})\right)^2$	$[0, 1]^6$	6	-3.32
$F_{21}(X) = -\sum_{i=1}^{5} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	$[0, 10]^n$	4	-10.1532
$F_{22}(X) = -\sum_{i=1}^{7} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	$[0, 10]^n$	4	-10.4028
$F_{23}(X) = -\sum_{i=1}^{10} [(X - a_i)(X - a_i)^T + c_i]^{-1}$	$[0, 10]^n$	4	-10.5363

The determination of the position of each queen is accomplished by the use of the notation *(i,j)*, where *i* denotes the row and *j* represents the column as illustrated in Fig. 5.

The efficiency of AOOA was proved by comparing its performance with six different optimization algorithms: POA, COA, ASBO, GPA, GSO, and ESCO. These algorithms are selected because all of them are recent metaheuristic algorithms. They can be categorized into two groups: the first group is new algorithms created from scratch which includes POA, COA, and ASBO. The second group is the development of existing ones which overcome the shortcomings of the original algorithms including GPA, GSO, and ESCO.

The experiments were performed through fifty iterations and ten independent runs using 23 different benchmark functions, 10 independent runs, and 50 iterations in each run. The performance of AOOA is evaluated statistically using the arithmetic mean and the standard deviation derived from ten different runs. The standard deviation demonstrates the stability of this method, while the arithmetic mean displays how well AOOA works on average. Standard deviation

Table 6. Comparison of AOOA with competing algorithms in solving unimodal functions

F	Parameter	POA	COA	ASBO	GPA	GSO	ESCO	AOOA
1	**mean**	3.6364	0.0424	0.0124	2.0003×10^{2}	2.5476×10^{4}	0.0000	0.0000
	Std	1.3906×10^{1}	0.0681	0.0076	6.7222×10^{1}	8.6026×10^{3}	0.0000	0.0000
	best	0.0000	0.0037	0.0023	7.6136×10^{1}	1.1069×10^{4}	0.0000	0.0000
	worst	6.4000×10^{1}	0.3077	0.0306	3.3011×10^{2}	4.5043×10^{4}	0.0000	0.0000
	mean rank	4	3	2	5	6	1	1
2	**mean**	0.0000	0.0000	0.0000	5.9243×10^{17}	2.9481×10^{33}	0.0000	0.0000
	Std	0.0000	0.0000	0.0000	2.7788×10^{18}	1.1671×10^{34}	0.0000	0.0000
	Best	0.0000	0.0000	0.0000	0.0000	3.9329×10^{22}	0.0000	0.0000
	worst	0.0000	0.0000	0.0000	2.0003×10^{19}	5.4553×10^{34}	0.0000	0.0000
	mean rank	1	1	1	3	2	1	1
3	**mean**	3.2228×10^{4}	4.3760×10^{2}	3.6309×10^{2}	3.2715×10^{3}	4.0693×10^{4}	0.0097	**0.0000**
	Std	2.9756×10^{4}	8.8119×10^{2}	6.5978×10^{2}	1.0851×10^{3}	2.1833×10^{4}	0.0349	**0.0000**
	best	0.0000	1.9532	2.0772×10^{1}	1.2071×10^{3}	1.2621×10^{4}	0.0000	**0.0000**
	worst	1.1119×10^{5}	4.1855×10^{3}	2.8712×10^{3}	5.4733×10^{3}	9.7657×10^{4}	0.1645	**0.0000**
	mean rank	6	4	3	5	7	2	1
4	**mean**	3.5818×10^{1}	1.0377	0.2120	1.8999×10^{1}	5.6858×10^{1}	0.0006	**0.0002**
	Std	2.9754×10^{1}	0.4585	0.0829	6.7488	8.6944	0.0009	**0.0003**
	best	0.0000	0.2776	0.1012	1.1093×10^{1}	4.1105×10^{1}	0.0000	0.0000
	worst	8.8000×10^{1}	1.8408	0.4024	4.2141×10^{1}	6.8931×10^{1}	0.0043	**0.0009**
	mean rank	6	4	3	5	7	2	1
5	**mean**	8.2305×10^{5}	2.4775×10^{1}	2.4081×10^{1}	9.2654×10^{3}	4.9669×10^{7}	2.3951×10^{1}	**3.24**
	Std	3.6957×10^{6}	1.6774	0.1239	6.9375×10^{3}	1.8904×10^{7}	0.0262	3.2
	best	2.4000×10^{1}	2.3971×10^{1}	2.3786×10^{1}	2.1812×10^{3}	1.6667×10^{7}	2.3854×10^{1}	**0.0333**
	worst	1.7361×10^{7}	3.2077×10^{1}	2.4339×10^{1}	3.1916×10^{4}	9.9967×10^{7}	2.3982×10^{1}	9.61
	mean rank	6	4	3	5	7	2	1
6	**mean**	6.9909×10^{1}	5.1718	3.7859	1.6244×10^{2}	2.4017×10^{4}	4.8053	4.1163
	Std	2.9976×10^{2}	0.4684	0.5651	6.5430×10^{1}	6.3396×10^{3}	0.4337	0.862517
	best	6.0000	4.0283	2.0291	7.9222×10^{1}	1.2186×10^{4}	3.8911	$\mathbf{4.718\times10^{-8}}$
	worst	1.4120×10^{3}	6.0239	4.4553	3.2928×10^{2}	3.9807×10^{4}	5.6261	6.895502
	mean rank	5	4	1	6	7	3	2
7	**mean**	0.2189	0.0358	0.0527	0.2462	2.2771×101	0.0088	**0.0000**
	Std	0.6327	0.0208	0.0339	0.1271	1.1237×101	0.0059	**0.0000**
	best	0.0040	0.0018	0.0125	0.0836	6.5735	0.0000	0.0000
	worst	3.0286	0.0897	0.1312	0.5279	5.2111×101	0.0233	0.00011535
	mean rank	5	3	4	6	7	2	1

(Std), best values, worst value, and mean rank of the benchmark functions of unimodal F_1-F_7, multimodal F_8-F_{13}, and fixed dimension functions F_{14}-F_{23} are reported in Tables 6-8. For each benchmark, the best values in the tables are indicated in bold type.

The first test of AOOA is to solve unimodal functions, F1-F7, which have a single optimal solution (f_{min}=0) with various large search spaces (except F7 which has a small search space). The goal is to find the unique optimal solution by exploiting promising regions in a short time. In this study, the used dimension for all functions is 30. Table 6 shows the results of unimodal functions. AOOA outperforms competing algorithms in mean rank with the first rank in six out of seven functions: F1, F2, F3, F4, F5, andF7. At the same time, it comes in the second rank in F6. The results' precision with less than 10^{-4} is rounded to 0.0000. In some functions, some competing algorithms also come in first place on a mean ranking like ESCO in F1 and POA, COA,

Table 7. Comparison of AOOA with competing algorithms in solving multimodal functions

F	Parameter	POA	COA	ASBO	GPA	GSO	ESCO	AOOA
8	**mean**	-2.2324×10^3	-3.4020×10^3	-3.0055×10^3	-5.5483×10^3	-2.6046×10^3	-3.3724×10^3	**-6.5099×10^2**
	Std	4.4147×10^2	4.2619×10^2	6.1490×10^2	8.5996×10^2	6.2191×10^2	4.3061×10^2	8.70×10^2
	best	-3.1517×10^3	-4.2537×10^3	-4.7884×10^3	-6.8697×10^3	-3.6837×10^3	-4.2086×10^3	**-1.21190×10^3**
	worst	-1.4180×10^3	-2.6287×10^3	-2.0362×10^3	-3.8963×10^3	-1.2388×10^3	-2.6239×10^3	-3.6741×10^3
	mean rank	7	3	5	2	6	4	1
9	**mean**	1.5437	3.6085	8.7277	1.1965×10^2	2.2540×10^2	0.0000	0.0000
	Std	4.5913	1.1264×10^1	1.9805	2.4053×10^1	4.2901×10^1	0.0000	0.0000
	best	6.7213×10^1	6.7213×10^1	6.7213×10^1	6.7213×10^1	6.7213×10^1	6.7213×10^1	**0.0000**
	worst	1.9175×10^1	5.2404×10^1	1.4001×10^1	1.7450×10^2	2.9682×10^2	0.0000	0.0000
	mean rank	2	3	4	5	6	1	1
10	**mean**	5.5532	0.0430	2.5481	5.3988	$1.8732\text{x}10^1$	0.0000	0.0000
	Std	7.2591	0.0348	0.3645	0.8562	0.7345	0.0001	**0.0000**
	best	0.0000	0.0088	2.0085	3.9091	$1.6595\text{x}10^1$	0.0000	0.0000
	worst	$1.6729\text{x}10^1$	0.1308	3.4214	6.7558	$1.9817\text{x}10^1$	0.0006	**0.0001**
	mean rank	6	3	4	5	7	1	1
11	**mean**	0.8503	0.1481	0.3167	2.3189	2.3085x102	0.0000	0.0000
	Std	2.0821	0.2431	0.1512	0.4336	6.9652x101	0.0000	0.0000
	best	0.0000	0.0010	0.0948	1.3487	8.1073x101	0.0000	0.0000
	worst	8.4397	0.7658	0.5993	3.4639	3.3599x102	0.0000	0.0001
	mean rank	5	3	4	6	7	1	1
12	**mean**	1.8112	0.7146	0.0916	$1.5973\text{x}10^1$	$5.9599\text{x}10^7$	0.8379	**0.025611**
	Std	0.2402	0.2273	0.1935	7.6043	4.6406x107	0.1560	**0.065284**
	best	1.7600	0.3883	0.0078	4.5788	1.0076x106	0.4525	**0.0000**
	worst	2.8865	1.1477	0.7299	3.2081x101	1.5818x108	1.1132	**0.2212**
	mean rank	5	3	2	6	7	4	1
13	**mean**	$1.2984\text{x}10^6$	3.1785	8.3868	$7.6457\text{x}10^1$	$1.6530\text{x}10^8$	3.0718	**1.9357**
	Std	$5.1387\text{x}10^6$	0.2635	0.9470	$9.8608\text{x}10^1$	$9.3905\text{x}10^7$	0.1189	0.7250
	best	3.0144	2.4778	6.9080	8.1068	$2.1460\text{x}10^7$	2.7266	**0.0324**
	worst	$2.3426\text{x}10^7$	3.6744	$1.0174\text{x}10^1$	$4.2394\text{x}10^2$	$3.6046\text{x}10^8$	3.2059	3.8897
	mean rank	6	3	4	5	7	2	1

ASBO, and ESCO in F2. The results indicate the exploitation ability of AOOA and well performance in the large search space in addition to the small search space. It is worth noting that the good exploitation capability of AOOA is because the process of generating new solutions comes from two phases (queen fertilization and workers' lifecycle). Moreover, AOOA reached the global optima in four functions: F1, F2, F3, and F7. The stability of the proposed algorithm was proven by the Std values. AOOA's Std was equal to the mean in five functions: F1, F2, F3, F4, and F7.

The second test of AOOA is to solve six multimodal functions, F8-F13, which have a single global optimum with multiple local optima. All functions have global optima (f_{min}=0) except F8 which has a global optima f_{min} =-12569 with various large to vary large search spaces. The used dimension for all functions is 30. The key challenge is to find the global optima and avoid being stuck in the local optima. Table 7 exhibits the results of multimodal functions. It can be observed that the mean rank of AOOA is in the first of all six multimodal functions: F8-F13. In addition, it outperformed the competing algorithms in the best solution value in F8, F9, F10,F12, and F13. Furthermore, AOOA reached the optimal solution in F9, F10, and F11. From the competing algorithms, ESCO's mean rank was also in the first in F9, F10, and F11. The exploration capability of AOOA proved its ability to solve very large space problems. AOOA stability was demonstrated by the Std values which were equal to the mean in three functions: F9, F10, and F11.

The third test of AOOA is to solve ten fixed-dimension multimodal functions: F14-F23. The fixed-dimension function is small in range and dimensions and has different f_{min}. When investigating the AOOA results depicted in Table 8, it can be seen that the proposed algorithm is in the first mean rank in five functions: F14, F18, F19, F22, and F23. It

Table 8. Comparison of AOOA with competing algorithms in solving fixed-dimension multimodal functions

F	Parameter	POA	COA	ASBO	GPA	GSO	ESCO	AOOA
14	**mean**	9.7338	4.7257	4.5799	1.7969	30.007	6.0382	**0.9900**
	Std	4.2440	3.9108	2.6687	1.0624	8.4975×10^{1}	3.7220	**0.524964**
	best	1.0291	0.9980	1.9920	0.9980	0.9980	1.0023	0.9980
	worst	1.2670×10^{1}	1.3619×10^{1}	1.0763×10^{1}	4.9505	4.1895×10^{2}	1.2671×10^{1}	1.9920
	mean rank	6	4	3	2	7	5	1
15	**mean**	0.1071	0.0053	0.1123	0.0057	0.0384	0.0036	0.0110
	Std	0.0549	0.0085	0.0403	0.0077	0.0380	0.0062	**0.0017**
	best	0.0023	0.0004	0.0257	0.0007	0.0013	0.0004	0.0074
	worst	0.1484	0.0338	0.1484	0.0204	0.1170	0.0226	**0.0156**
	mean rank	6	2	7	3	5	1	4
16	**mean**	-0.4391	-1.0311	-0.0387	-1.0315	-0.9048	-1.0300	-0.9766
	Std	0.4548	0.0008	0.0903	0.0001	0.2622	0.0025	0.0694
	best	-0.9216	-1.0316	-0.2956	-1.0316	-1.0316	-1.0316	-1
	worst	0.0000	-1.0285	0.0000	-1.0313	-0.2477	-1.0201	-0.74995
	mean rank	6	2	7	1	4	3	5
17	**mean**	2.5935	0.4060	1.1446	0.3981	0.6136	0.4042	0.4028
	Std	3.0319	0.0339	1.2706	0.0000	0.8370	0.0101	0.0029
	best	0.4438	0.3981	0.6438	0.3981	0.3981	0.3981	0.4002
	worst	1.2729×10^{1}	0.5578	6.1148	0.3983	4.3170	0.4458	0.4092
	mean rank	7	4	6	1	5	3	2
18	**mean**	3.9242×10^{1}	1.1356×10^{1}	2.8000×10^{1}	3.0012	1.3103×10^{1}	3.0668	**3.0000**
	Std	0.0011	0.0011	0.0011	0.0011	0.0011	0.0011	0.0000
	best	3.0000	3.0000	3.0000	3.0001	3.0001	3.0001	**3.0000**
	worst	1.7139×10^{2}	8.4157×10^{1}	2.7800×10^{2}	3.0043	9.3723×10^{1}	3.3429	3.0000
	mean rank	7	4	6	2	5	3	1
19	**mean**	-0.0495	-0.0495	-0.0495	-0.0495	-0.0147	-0.0495	**-0.8170**
	Std	0.0000	0.0000	0.0000	0.0000	0.0132	0.0000	0.0000
	best	-0.0495	-0.0495	-0.0495	-0.0495	-0.0495	-0.0495	**-0.918**
	worst	-0.0495	-0.0495	-0.0495	-0.0495	0.0000	-0.0495	**-0.8000**
	mean rank	2	2	2	2	3	2	1
20	**mean**	-1.1154	-3.1067	-0.5785	-3.2873	-2.0987	-2.9316	0.0000
	Std	0.5684	0.0846	0.4211	0.0664	0.6779	0.2761	0.0000
	best	-2.7233	-3.2627	-1.6231	-3.3222	-3.1465	-3.1995	0.0000
	worst	-0.1895	-2.9434	-0.0849	-3.1208	-0.9088	-2.2183	0.0000
	mean rank	5	2	6	1	4	3	7
21	**mean**	-0.4190	-5.7729	-3.9940	-6.3070	-2.4484	-3.9766	-4.7371
	Std	0.0993	2.3804	3.8261	3.2695	1.9795	0.6626	0.14585
	best	-0.6601	-8.8823	-1.0153×10^{1}	-1.0148×10^{1}	-9.3624	-4.8331	-5.6660
	worst	-0.3172	-2.3876	-0.4965	-2.6235	-0.5020	-2.8793	-4.1884
	mean rank	7	2	4	1	6	5	3
22	**mean**	-0.4701	-4.5937	-4.2655	-7.3651	-1.9404	-3.9079	**-7.5457**
	Std	0.1947	1.8577	3.3589	3.2304	0.9546	1.3811	0.9718
	best	-1.0086	-8.9320	-1.0403×10^{1}	-1.0384×10^{1}	-4.1265	-8.7333	-10.0001
	worst	-0.2936	-2.3489	-0.9100	-2.7460	-0.5520	-2.0361	-5.0885
	mean rank	7	3	4	2	6	5	1
23	**mean**	-0.6132	-5.3194	-2.4003	-6.1046	-2.3495	-3.7469	**-6.5499**
	Std	0.2298	2.3417	1.5470	3.6205	1.5521	0.6773	0.0000
	best	-1.2409	-9.8285	-5.1285	-1.0520×10^{1}	-7.8994	-4.8052	-6.5499
	worst	-0.3774	-2.4063	-0.5556	-2.4904	-0.7323	-2.0363	**-6.5498**
	mean rank	7	3	5	2	6	4	1

came in the second mean rank in F17 and the third, fourth, and fifth in F21, F15, and F16, respectively. Std was equal to zero in four functions: F18, F19, F20, and F23 and near zero in the remaining six functions which reflects the stability of AOOA. Moreover, AOOA reached the global optima in F18. Our algorithm came in the last position of mean rank in F20. This is due to the flat nature of the surface and the global optima located in a very narrow region.

Although among the three function groups, the fixed dimension was the challenge where our algorithm needed to perform better; AOOA's performance exceeded the competing algorithms in terms of mean rank in half the number of the functions. Moreover, the nearest competitor is the GPA which was in the first mean rank in F16, F17, F20, and F21 and the second mean rank in F14, F18, F19, F22, and F23. GPA came in the third rank of mean in F15.

5.2 Sensitivity analysis

Population size plays a key role in optimization algorithms and significantly affects the output of these algorithms. It is reasonable to inspect the impact of changing the main parameters that affect the population size during iterations which in turn uncover the power of the proposed algorithm. From this perspective, two apiary sizes, Small Scale Apiary(SSA) and Medium Scale Apiary (MSA), were used to test AOOA against different population sizes. In each run, AOOA iterated on different hive numbers (h) and bee numbers (b) values for SSA and MSA, forming various population sizes (N) ranging from 90 individuals (according to Eq. (1)) where h=3 and b=30 to 560 individuals in which h=8 and b=70. Table 9 shows the various apiary scales with different population sizes tried during experiments for each iteration.

The best and mean fitness score of each function: F1-F23 resulted from fifty iterations and ten independent runs using SSA and MSAare shown in Table 10. It can be seen that the impact of using various population sizes is equal for best and mean fitness in F1 and F2 of unimodal functions. The mean fitness result of MSA is better than SSA in F3-F7 of

Table 9. Various Population Sizes of SSA and MSA

P	**Value**					
	SSA					
h	3	h	3	h	3	h
b	30, 40, 50	b	30, 40, 50	b	30, 40, 50	b
N	90, 120, 150	N	90, 120, 150	N	90, 120, 150	N

Table 10. best and mean fitness results using SSA and MSA

F	Apiary Scale	Best Fitness	mean fitness
1	SSA	0	0
	MSA	0	0
2	SSA	0	0
	MSA	0	0
3	SSA	0	0.000144
	MSA	0	**0.000012**
4	SSA	0	0.000576
	MSA	0	**0.000252**
5	SSA	0.121546	14.7706
	MSA	**0.0333**	**9.6100**
6	SSA	1.92×10^{-7}	4.816719
	MSA	**4.718×10^{-8}**	**4.116314**
7	SSA	1.05×10^{-4}	9.98×10^{-3}
	MSA	**8.51422×10^{-7}**	**3.87487×10^{-5}**
8	SSA	-11530.53	-6027.84
	MSA	**-12119.03**	**-6509.97**
9	SSA	2.49×10^{-8}	1.04×10^{-4}
	MSA	**0**	**4.76×10^{-7}**
10	SSA	0	0.000146
	MSA	0	**0.000020**
11	SSA	5.84E-11	0.000147
	MSA	**0**	**0.000019**
12	SSA	5.01×10^{-5}	0.081379
	MSA	**6.28×10^{-6}**	**0.025611**
13	SSA	0.119018	2.519000
	MSA	**0.032456**	**1.935705**
14	SSA	1.02945	2.8434
	MSA	**0.99800**	**0.99000**
15	SSA	0.0083	0.012734
	MSA	**0.0074**	**0.011093**
16	SSA	-1	-0.9565
	MSA	-1	**-0.9766**
17	SSA	**0.4002102**	**0.40287**
	MSA	0.4002106	0.41169
18	SSA	3	3
	MSA	3	3
19	SSA	**-0.918**	-0.812
	MSA	-0.915	**-0.817**
20	SSA	-0.0000772	-0.0000772
	MSA	-0.0000772	-0.0000772
21	SSA	**-5.666024929**	**-4.737192362**
	MSA	-2.5	-2.5
22	SSA	**-10.0001**	**-7.54575**
	MSA	-2.5	-2.5
23	SSA	-6.549915	-6.549902
	MSA	-6.549915	**-6.549907**

Received: February 17, 2024.   Revised: April 20, 2024.

unimodal functions, F8-F13 of multimodal and F14-F19 and F23 of fixed dimension functions while F20 MSA and SSA were equal and achieved the same results. In F21 and F22 SSA best and mean fitness is better than MSA which indicates that problems nature affects the results. In all functions except F21 and F22, MSA's best fitness results were either better than or equal to SSA results which indicates the effect of using multiple populations with more hives and bees in improving the results besides increasing the computational cost.

5.3 Performance Analysis

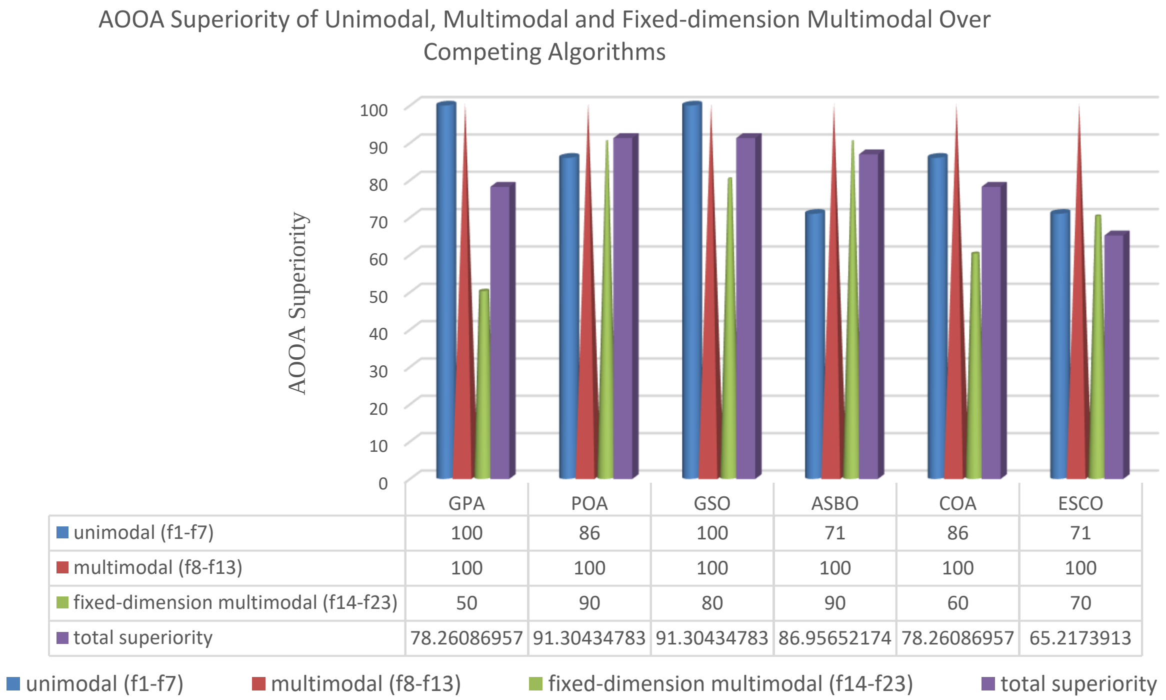
In this section, the results and findings analysis will be discussed. The discussion is grouped into two parts: performance comparison with competing algorithms and limitations.

In terms of the first part, AOOA mean fitness outperformed POA and COA in solving 6 out of 7 of the F1-F7 unimodal functions with 86% superiority. While it outperformed GPA and GSO in solving 7 out of 7 of the F1-F7 function with 100% superiority. AOOA overcame ASBO and ESCO in solving 5 out of 7 of the F1-F7 function with 71% superiority.

In multimodal functions, AOOA mean fitness outperformed POA, COA, ASBO, GPA, and GSO in solving all functions F8-F13 with 100% superiority while it overcame ESCO in solving F8, F12, and F13 with 50% superiority.

The performance of AOOA mean fitness was better than POA and ASBO in solving 9 out of 10 fixed-dimension multimodal functions with 90% superiority. For GSO, ESCO, COA, and GPA, AOOA was better in solving 8, 7, 6, and 5 functions out of 10, respectively with 80%, 70%, 60% and 50% superiority. AOOA overcomes GPA, POA, GSO, ASBO, COA, and ESCO in solving 18, 21, 21, 20, 18, and 15 with the percentage of total superiority of 78.26%, 91.3%, 91.3%, 86.95, 78.26, and 65.21, respectively for 23 benchmark functions. Fig. 6 illustrates the superiority of AOOA over the competing algorithms.

AOOA was superior in solving 17 out of 23 benchmark functions with a total performance of approximately 74%. In unimodal functions, AOOA achieved 86% superiority by solving 6 out of 7. AOOA solved 6 out of 6 multimodal functions with 100% superiority. According to fixed dimension functions, AOOA was superior in solving half the number of functions; this reveals that AOOA performance is better in a narrow and large range of high-dimension functions and has a good balance between exploitation and exploration capability. This balance is achieved through simulating more than one behaviour (queen fertilization and worker lifecycle) that evolute new better individuals to exploit the search space and more than one behaviour (drone exchange, queen investiture, and swarming) that explores the search space. On the other hand, the fading-out phase eliminates the worst individuals to help keep a good population. Fig. 7 shows the AOOA performance of the mean fitness over all 23 functions.



	GPA	POA	GSO	ASBO	COA	ESCO
unimodal (f1-f7)	100	86	100	71	86	71
multimodal (f8-f13)	100	100	100	100	100	100
fixed-dimension multimodal (f14-f23)	50	90	80	90	60	70
total superiority	78.26086957	91.30434783	91.30434783	86.95652174	78.26086957	65.2173913

Figure. 6 AOOA superiority over competing algorithms

 DOI: 10.22266/ijies2024.0630.61

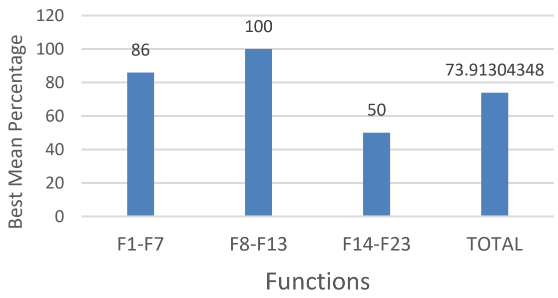


Figure. 7 AOOA Performance against benchmark functions

The second part analyzes the algorithm limitations. AOOA was tested using only 23 benchmark functions. The proposed algorithm can be tested using other benchmark functions like IEEE CEC 2017 [34], IEEE CEC 2020 [35], etc.

As with any metaheuristic optimization algorithm, AOOA relies on the exploitation and exploration concepts which need more balance to enhance the fitness results of some fixed-dimension functions (especially where the mean rank is in the last) to hopefully reach the optimal solution.

On the other hand, AOOA needs to be tested on practical NP-hard problems. Moreover, the computational cost and resources are required to be checked especially for large datasets with various parameter settings.

6. Conclusion

This paper introduced a new nature-inspired metaheuristic optimization algorithm which was the so-called AOOA. Its main structure was based on the organizational behaviour of honey bees inside the apiary. The research demonstrated how the nature-inspired algorithm gains its efficiency from the queen, workers, and drones' activities. Examples of such activities are drone exchange between beehives, queen fertilization, workers' activities during their lifecycle, queen investiture, and fading out. Bees' activities were converted in the proposed algorithm into a mathematical model to find the problems' optimal solutions. AOOA exhibited robust performance and promising outcomes by introducing the concept of "Apiary" which employed multiple populations within many colonies and several behaviours.To prove the proposed algorithm's exploration and exploitation ability, it was applied to 23 benchmark functions, involving seven unimodal, six multimodal, and ten fixed-dimension multimodal classical functions. The results were statistically assessed using the mean and standard deviation measures. The overall findings were promising in comparison with GPA, POA, GSO, ASBO, COA, and ESCO metaheuristic optimization algorithms. AOOA outperformed GPA, POA, GSO, ASBO, COA, and ESCO in solving 18, 21, 21, 20, 18, and 15 with the percentage of total superiority of 78.26%, 91.3%, 91.3%, 86.95, 78.26, and 65.21, respectively for 23 benchmarks functions. The AOOA was superior in solving 17 out of 23 benchmark functions with a total performance of approximately 74%. AOOA achieved 86% superiority by solving 6 out of 7 of unimodal functions, 6 out of 6 multimodal functions with 100% superiority, and 5 out of 10 fixed dimension functions with 50% superiority.

Future work is aimed to enhance the exploitation ability of AOOA and explore several ways that can optimize diversification. Moreover, the balancing between the fertilization ratio and population size increment will be handled. In addition, investing the time increment of very large search regions. Additional computations for memory consumption can be involved. AOOA can be used to solve many other NP-hard problems such as the Area coverage problem, Robot Path Planning, Engineering optimization problems, etc.

Conflicts of Interest

The authors declare no conflicts of inter to publish the results.

Author Contributions

Conceptualization, Ahmed T. Sadiq Al-Obaidi and Safaa O. Al-mamory; methodology, Mais A. Al-Sharqi and Ahmed T. Sadiq Al-Obaidi; software, Mais A. Al-Sharqi; validation, Ahmed T. Sadiq Al-Obaidi and Safaa O. Al-mamory; formal analysis, Mais A. Al-Sharqi and Ahmed T. Sadiq Al-Obaidi; investigation, Mais A. Al-Sharqi; resources, Mais A. Al-Sharqi; data curation, Mais A. Al-Sharqi; writing—original draft preparation, Mais A. Al-Sharqi; writing—review and editing, Safaa O. Al-mamory and Mais A. Al-Sharqi; visualization, Mais A. Al-Sharqi; supervision, Ahmed T. Sadiq Al-Obaidi and Safaa O. Al-mamory; project administration, Ahmed T. Sadiq Al-Obaidi and Safaa O. Al-mamory.

Funding

This research received no external funding.

Data Availability Statement

The Brandimarte dataset is available at https://github.com/guillaumebour/flexible-job-shop/tree/master/test_data/Brandimarte_Data/Text. Hurink dataset is available at https://github.com/guillaumebour/flexible-job-shop/tree/master/test_data/Hurink_Data/Text.

References

[1] K. W. Huang, Z. X. Wu, C. L. Jiang, Z. H. Huang, and S. H. Lee, "WPO: A Whale Particle Optimization Algorithm", *International Journal of Computational Intelligence Systems*, Vol. 16, No. 1, 2023, doi: 10.1007/s44196-023-00295-6.

[2] K. Rajwar, K. Deep, and S. Das, "An exhaustive review of the metaheuristic algorithms for search and optimization: taxonomy, applications, and open challenges", *Artif Intell Rev*, Vol. 56, No. 11, pp. 13187–13257, 2023, doi: 10.1007/s10462-023-10470-y.

[3] J. He, Z. Peng, L. Zhang, L. Zuo, D. Cui, and Q. Li, "Enhanced crow search algorithm with multi-stage search integration for global optimization problems", *Soft comput*, Vol. 27, No. 20, pp. 14877–14907, 2023, doi: 10.1007/s00500-023-08577-z.

[4] X. Yang, Nature-inspired optimization algorithms, *First edition. London: Elsevier*, 2014.

[5] I. Al-Shourbaji, P. Kachare, S. Fadlelseed, A. Jabbari, A. G. Hussien, F. Al-Saqqar, L. Abualigah, and A. Alameen, "Artificial Ecosystem-Based Optimization with Dwarf Mongoose Optimization for Feature Selection and Global Optimization Problems", *International Journal of Computational Intelligence Systems*, Vol. 16, No. 1, 2023, doi: 10.1007/s44196-023-00279-6.

[6] Y. H. Choo, Z. Cai, V. Le, M. Johnstone, D. Creighton, and C. P. Lim, "Enhancing the Harris' Hawk optimiser for single- and multi-objective optimisation", *Soft comput*, Vol. 27, No. 22, pp. 16675–16715, 2023, doi: 10.1007/s00500-023-08952-w.

[7] D. H. Wolpert and W. G. Macready, "No Free Lunch Theorems for Optimization", 1997.

[8] P. D. Kusuma and A. L. Prasasti, "Guided Pelican Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 6, pp. 179–190, 2022, doi: 10.22266/ijies2022.1231.18.

[9] M. Dehghani, Z. Montazeri, E. Trojovská, and P. Trojovský, "Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems", *Knowl Based Syst*, Vol. 259, 2023, doi: 10.1016/j.knosys.2022.110011.

[10] P. D. Kusuma and A. Dinimaharawati, "Extended Stochastic Coati Optimizer", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 3, pp. 482–494, 2023, doi: 10.22266/ijies2023.0630.38.

[11] M. Noroozi, H. Mohammadi, E. Efatinasab, A. Lashgari, M. Eslami, and B. Khan, "Golden Search Optimization Algorithm", *IEEE Access*, Vol. 10, pp. 37515–37532, 2022, doi: 10.1109/ACCESS.2022.3162853.

[12] M. Dehghani, Š. Hubálovský, and P. Trojovský, "A new optimization algorithm based on average and subtraction of the best and worst members of the population for solving various optimization problems", doi: 10.7717/peerj.

[13] F. A. Zeidabadi and M. Dehghani, "POA: Puzzle Optimization Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 1, pp. 273–281, 2022, doi: 10.22266/IJIES2022.0228.25.

[14] R. Moritz and R. Crewe, "The Dark Side of the Hive", *the United States of America Press 198 Madison Avenue*, New York, NY 10016, United States of America., 2018.

[15] L. Insolia, R. Molinari, S. R. Rogers, G. R. Williams, F. Chiaromonte, and M. Calovi, "Honey bee colony loss linked to parasites, pesticides and extreme weather across the United States", *Sci Rep*, Vol. 12, No. 1, 2022, doi: 10.1038/s41598-022-24946-4.

[16] K. V. Frisch, *The Dancing Bees: ACCOUNT OF THE LIFE AND SENSES OF THE HONEY BEE. Springer Vienna*, 1954, doi: 10.1007/978-3-7091-4697-2.

[17] T. D. Seeley, *Honeybee democracy. Princeton University Press*, 2010.

[18] T. D. Seeley, *The Wisdom of the Hive: The Social Physiology of Honey Bee Colonies*, London, England, 1995.

[19] B. Moisset and S. Buchmann, *Bee Basics An Introduction to Our Native Bees A USDA Forest Service and Pollinator Partnership Publication. A USDA Forest Service and Pollinator Partnership Publication*, 2011.

[20] E. Mader, M. Spivak, and E. Evans, "Managing Alternative Pollinators: A Handbook for Beekeepers, Growers, and Conservationists", *Sustainable Agriculture Research and Education (SARE) Natural Resource, Agriculture, and Engineering Service (NRAES)*, Vol. 11, 2010, [Online]. Available: www.nraes.org

[21] S. Krishnan, G. W. Guerra, D. Bertrand, S. W. Kanounnikoff, and C. Kettle, "The pollination services of forests: A Review of Forest And

Landscape Interventions To Enhance Their Cross-Sectoral Benefits", *FAO and Bioversity International*, 2020, doi: 10.4060/ca9433en.

[22] C. Silva, J. N. Radaeski, M. Arena, and S. Bauermann, "Atlas of pollen and plants used by bees", *First edition. Consultoria Inteligente em Servicos Ecossistemicos (CISE)*, 2020.

[23] L. A. Garibaldi, M. Dondo, J. Hipólito, N. Azzu, B. F. Viana, and M. Kasina, "A quantitative approach to the socio-economic valuation of pollinator-friendly practices : a protocol for its use", *Food and Agriculture Organization of the United Nations (FAO), United Nations Environment Programme (UNEP)*, 2016.

[24] Diana Sammataro and Jay A. Yoder, *Honey Bee Colony Health, Challenges and Sustainable Solutions*, 2012.

[25] H. R. Hepburn, C. W. W. Pirk, and O. Duangphakdee, *Honeybee Nests Composition, Structure, Function*, Springer Nature, 2014.

[26] B. Gemmill-Herren, N. Azzu, A. Bicksler, and A. Guidotti, "Towards sustainable crop pollination services", *Rome: FAO*, 2020. doi: 10.4060/ca8965en.

[27] C. Li and M. Research Fellow, "UNDERSTANDING, CONSERVATION AND PROTECTION OF PRECIOUS NATURAL RESOURCES UNDERSTANDING, CONSERVATION AND PROTECTION OF PRECIOUS NATURAL RESOURCES-BEES", In: *Proc. of Technology, Engineering, Arts, and Mathematics (STEAM)*, Lenox Institute Press, 2019.

[28] N. Bradbear, "Bees and their role in forest livelihoods", *Rome: Food And Agriculture Organization of The United Nations*, 2009.

[29] T. D. Seeley, "Honeybee democracy", *Princeton University Press*, 2010.

[30] "Beekeeping calendar for the Northeast."

[31] M. Dehghani and P. Trojovský, "Hybrid leader based optimization: a new stochastic optimization algorithm for solving optimization applications", *Sci Rep*, Vol. 12, No. 1, 2022, doi: 10.1038/s41598-022-09514-0.

[32] K. Hussain, M. N. M. Salleh, S. Cheng, and R. Naseem, "Common benchmark functions for metaheuristic evaluation: A review", *International Journal on Informatics Visualization*, Vol. 1, No. 4–2. Politeknik Negeri Padang, pp. 218–223, 2017, doi: 10.30630/joiv.1.4-2.65.

[33] M. Jamil, "A literature survey of benchmark functions for global optimisation problems Xin-She Yang", *arXiv:1308.4008*, 2013, [Online]. Available: http://www.geatbx.com/

[34] G. Wu, R. Mallipeddi, and P. N. Suganthan, "Problem Definitions and Evaluation Criteria for the CEC 2017 Competition and Special Session on Constrained Single Objective Real-Parameter Optimization", *Technical Report*, 2017, [Online]. Available: https://www.researchgate.net/publication/317228117

[35] J. Liang, P. N. Suganthan, B. Y. Qu, D. W. Gong and C. T. Yue, "Problem Definitions and Evaluation Criteria for the CEC 2020 Special Session on Multimodal Multiobjective Optimization", *Technical Report*, 2019, doi: 10.13140/RG.2.2.31746.02247.