ARTICLE IN PRESS

Swarm and Evolutionary Computation ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

journal homepage: www.elsevier.com/locate/swevo

Computing with the collective intelligence of honey bees – A survey

Anguluri Rajasekhar [a], Nandar Lynn [b], Swagatam Das [c,*], P.N. Suganthan [b,*]

[a] Department of Mechanical Engineering, University of California, Riverside, United states
[b] School of Electrical Electronic Engineering, Nanyang Technological University, 639798 Singapore, Republic of Singapore
[c] Electronics and Communication Sciences Unit Indian Statistical Institute, 203 B T Road, Kolkata 700108, India

ARTICLE INFO

Article history:
Received 26 September 2015
Received in revised form
22 April 2016
Accepted 3 June 2016

Keywords:
Nature inspired computing
Swarm intelligence
Honey bees foraging
Particle swarm optimization
Bee colony optimization

ABSTRACT

Over past few decades, families of algorithms based on the intelligent group behaviors of social creatures like ants, birds, fishes, and bacteria have been extensively studied and applied for computer-aided optimization. Recently there has been a surge of interest in developing algorithms for search, optimization, and communication by simulating different aspects of the social life of a very well-known creature: the honey bee. Several articles reporting the success of the heuristics based on swarming, mating, and foraging behaviors of the honey bees are being published on a regular basis. In this paper we provide a brief but comprehensive survey of the entire horizon of research so far undertaken on the algorithms inspired by the honey bees. Starting with the biological perspectives and motivations, we outline the major bees-inspired algorithms, their prospects in the respective problem domains and their similarities and dissimilarities with the other swarm intelligence algorithms. We also provide an account of the engineering applications of these algorithms. Finally we identify some open research issues and promising application areas for the bees-inspired computing techniques.

Contents

1. Introduction . . . 2
2. Honey bees – biological perspectives and motivations . . . 2
 2.1. Honey bee hive . . . 2
  2.1.1. Queen Bee . . . 2
  2.1.2. Drone bees . . . 2
  2.1.3. Worker bees . . . 2
 2.2. Honey bee foraging and communication . . . 3
  2.2.1. Foraging . . . 3
  2.2.2. Communication . . . 4
 2.3. Honey bee swarming and nest-site-selection . . . 4
 2.4. Spatial memory and navigation . . . 4
 2.5. Division of labor . . . 4
3. Computational methods based on Queen Bees, mating, marriage . . . 4
 3.1. Queen Bee Evolution (QBE) . . . 4
 3.2. Marriage in Honey Bees Optimization (MBO) . . . 5
 3.3. Honey Bees Mating Optimization (HBMO) Algorithm . . . 6
4. Methods involving bees foraging and communication . . . 9
 4.1. Foraging involving communication based Methods . . . 9
  4.1.1. Bees System (BS) . . . 9
  4.1.2. Bees Colony Optimization (BCO) . . . 10
  4.1.3. Bees Colony Optimization Algorithm (BCOA) . . . 11
  4.1.4. Artificial Bee Colony Algorithm . . . 11
 4.2. The Basic ABC Algorithm . . . 12

* Corresponding authors.
E-mail addresses: rajasekhar.anguluri@ieee.org (A. Rajasekhar), swagatam.das@ieee.org, swagatam.das@isical.ac.in (S. Das), epnsugan@ntu.edu.sg (P.N. Suganthan).

http://dx.doi.org/10.1016/j.swevo.2016.06.001
2210-6502/

Please cite this article as: A. Rajasekhar, et al., Computing with the collective intelligence of honey bees – A survey, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.06.001

4.3. Improved ABC Variants 12
4.4. Hybrid ABC Variants 15
4.5. ABC for discrete and binary optimization problems 15
4.6. Real world applications 16
4.6.1. Bees Algorithm. 16
4.6.2. BeeAdhoc 17
4.6.3. Bees Swarm Optimization 17
4.6.4. Virtual Bee Algorithm 17
4.6.5. Miscellaneous methods resembling bees foraging in nature 18
4.7. Bee dances and communication 18
5. Methods based on Honey Bee swarming 18
5.1. Decision making and nest-site-selection 18
5.2. Floral and pheromone laying 18
6. Methods based on spatial memory and navigation 19
7. Methods based on division of labor 19
8. Potential future research directions with methods based on honey bees 19
9. Conclusion 20
Appendix A. Supplementary material. 21
References 21

1. Introduction

BEES are four-winged flower-feeding flying insects having similarities with ants and wasps and they belong to family of *Apoidea*. They are well known for their pollination and production of honey and beeswax. There are around 20,000 known species of bees in seven to nine recognized families. Among all the categories of bees, bumblebees and honey bees have a very peculiar life cycle which had attracted the several of researchers to study about the facts associated with them. These studies led to different theories which are highly suitable to meet the needs of engineering applications. In a honey bee swarm, the tasks are dynamically assigned and the bees are adapts to the new environment in a collaborative intelligent manner [S1,S2,S3].

In the recent past, with the computational cost having been almost dramatically reduced, the researchers have drawn the ideas from nature to solve difficult computational problems. Among the population-based models for computer aided problem solving, the insect colonies present a high level of interaction through mutual co-operation and organizational behavior which are necessary for survival in the life process. Computer scientists observed that efficient metaheuristic methods can be developed, based on the honey bees' intelligent behaviors of co-operation, allocating the task force and finding the food source using the mutual interaction as guidance.

The term *swarm* is used for an aggregation of natural creatures like fishes, birds and insects such as ants, termites, and bees. The members in the swarm interact with each other without any central supervision and they move based on the information obtained from their neighborhood. In this paper, a comprehensive survey of the computing algorithms based on the social behaviors of honey bees is presented including different variants of bee algorithms and applications of those variants to real-world optimization problems.

The paper is organized in the following way. Section 2 discusses the biological perspectives of honey bees and the motivations behind the algorithms inspired by different activities of honey bee swarm. Section 3 outlines various algorithms based on the mating and marriage of queen bees. In Section 4 studies based on developments of methods involving foraging and communications between bees are reviewed. Section 5 provides the survey of methods based on honey bee swarming. In Section 6 algorithms based on spatial memory and navigation methods are discussed. Section 7 presents a comprehensive outline of the algorithms based on the division of labor in honeybees. Finally, the paper is concluded in Section 8 with a discussion of the potential future research issues. A summary of the key concepts of some most representative bee-inspired algorithms along with the meaning of various metaphors commonly found in the related literature has been provided in Table 2.

2. Honey bees – biological perspectives and motivations

In nature honey bees create nests called "hives" (consisting up to 20,000 individuals) and they work together in highly structured social order. The brief taxonomy involved in the life cycle of bees is shown in Fig. 1. The following section provides a brief description of the evolution of bees and their role played in collection of nectar.

2.1. Honey bee hive

This is the place where new bees take birth. The hive consists of a number of vertical combs. There are two kinds of comb cells; the *worker* cells and the *drone* cells. In due course of time occasionally the bees build a third type of cells, the *queen cells*, in which the queen bees are reared.

2.1.1. Queen Bee

In general "Queen Bee" refers to an adult mated female. Her sole purpose in life is to lay eggs and to produce more bees. She is the only female in the hive.

2.1.2. Drone bees

Drones are the male honey bees developed from eggs that have not been fertilized. Drones mate with a queen in front of the hive through the phase of *mating flight.*

2.1.3. Worker bees

Workers are the most populated non-mating female bees, constituting over 98% of the colony's population. The worker bees plays major role in collection of nectar and in safe-guarding the hive from potential intruders through social interactions. Several bee colony based algorithms are based on the collective intelligence shown by the worker bees.

Please cite this article as: A. Rajasekhar, et al., Computing with the collective intelligence of honey bees – A survey, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.06.001

2.2. Honey bee foraging and communication

Learning plays a vital role in foraging. Von Frisch and Seeley are considered to be the pioneers in studying the learning capabilities of honey bees. From different experiments and many studies conducted by Von Frisch [S4,S5], Von Frisch and Lindauer [S6], Seeley [S7], Chalifman [S8] and Lindauer [S9], it is observed that a single forager will visit the different flowers (food sources) in the morning, if there were a particular kind of flower which has sufficient attraction and reward, the bee will visits that flower as many times as possible for most of the day. Honey bees are quite skillful in associative learning.

Table 1
Probability Rating for BCOA Algorithm.

Profitability Rating	r_i
$Pf_i < 0.9.Pf_{colony}$	0.60
$0.9.Pf_{colony} < Pf_i < 0.95.Pf_{colony}$	0.20
$0.95.Pf_{colony} < Pf_i < 1.15Pf_{colony}$	0.02
$1.155Pf_{colony} < Pf_i$	0.00

2.2.1. Foraging

Foraging process begins when the foragers (worker bees) leave the hive in search of food. Almost all flowers produce nectar to attract insects, primarily honey bees. A bee uses its cognitive intelligence and also observes the neighboring bees and then determines to which flower (source) she should visit. At the end foragers carry the collected nectar to hive.

Table 2
Summary of some of the most representative honey bee inspired algorithms.

Biological aspect of the Honey Bee Social Life Cycle	Name of Algorithm and Developers	Metaphor Deciphered	Features of the algorithm
Queen bees	Queen Bee Evolution (QBA) by Jung [64]	Queen bee → fittest solution, which is allowed to crossover with other solutions, within the framework of a standard GA with two mutation rates.	Imposes a strictly elitist form of crossover where the decision variables from the best solution trickle down to different offspring solutions through crossover. Basically a modification of GA.
Bees mating and marriage	Marriage in Bees Optimization (MBO) by Abbas [1]	Queen and drone → candidate solutions, workers → local search heuristics for improving offspring solutions (broods)	Involves several heuristic methods and greedy local search in the framework of GA as the new solutions (broods) are generated by using crossover and mutation. Number of control parameters is high and their tuning is not always straightforward.
Bees mating and marriage	Honey Bees Mating Optimization Algorithm (HBMO) by Haddad et al. [50,S28, S29]	Queen and drone → candidate solutions, broods → newly generated candidate solutions, workers → local search heuristics for improving offspring solutions	Very similar to MBO, the five stage algorithm adapts the local search heuristics based on the improvements of the newly generated solutions (broods).
Honey bees foraging	Bee System (BS) by Sato and Hagiwara [151]; alternative formulation by Lucic and Teodorovic [91]	Bees → search agents, searching for food → exploring the search space with low cost mathematical operators by some individuals of the population (scouts), Bee colony → population or sub-population of candidate solutions, Food source → candidate solution, Quality of food source → fitness of a solution.	Sato and Hagiwara's BS generates multiple solutions in the neighborhood of the best solution. In Lucic and Teodorovic's alternative BS (mostly applied for combinatorial optimization) employs additional exploring agents (scouts) for each population of solutions.
Honey bees foraging and communication	Bee Colony Optimization (BCO) by Teodorovic and Dell [170]	Waggle dance → sharing the information about quality of solution obtained by different search agents.(rest are similar to BS)	Bees create partial solutions to the combinatorial optimization problem and go on balancing between explorative forward pass and exploitative backward pass phases of the algorithm.
Honey bees foraging and communication	Bee Colony Optimization Algorithm (BCO) by Chong et al. [28]	Bees/foragers → search agents, Dance → sharing information about fitness of current solution, Recruitment → fresh start of building a solution, Following → continuing with the older partial solution, Nectar → destination state	Mostly applicable to combinatorial optimization problems, the foraging or nectar exploration part uses transition probabilities similar to the ant colony optimization algorithm.
Honey bees foraging and communication	Artificial Bee Colony (ABC) by Karaboga [69]	Bees → search agents, Food sources → candidate solutions, Nectar amount → fitness value of a solution, Employed bees → generate solutions in the neighborhood of existing ones, Onlooker bees → share information from employed bees and randomly choose some solutions for further exploration, Scout bees → neighborhood exploration of newly initialized solutions that replace older ones (which could not be improved by neighborhood explorations).	Most popular bees foraging based algorithm for continuous optimization (later extended for discrete and binary optimization problems). The explorative coverage of search space, especially form multi-modal functional landscapes can be poor. The neighborhood exploration in onlooker bees phase is by forming an arithmetic recombination of a randomly selected dimension of the current solution with the corresponding dimension of another solution available.
Honey bees foraging and communication	Bees Algorithm (BA) by Pham et al. [130]	Flower/food source/site → candidate solution, Bee colony → population of search agents, Profitability → fitness of a solution, Flower patch → neighborhood of a solution, Recruitment procedure → to search further the neighborhoods of the most promising solutions, site abandonment procedure→ re-initialize some of the poorer solutions.	The algorithm combines a neighborhood search strategy with global search and can be used for solving both continuous and combinatorial optimization problems. However, the exploration of the search space is not somewhat intelligently guided and the neighborhood search may waste FEs considerably.
Bees' communication	BeeAdHoc by Wedde et al. [185,S185]	Scout → agents discovering new routes to destination, Forager → transport data packets as well as evaluate the quality of the routes simultaneously.	An efficient routing algorithm for mobile ad hoc networks.

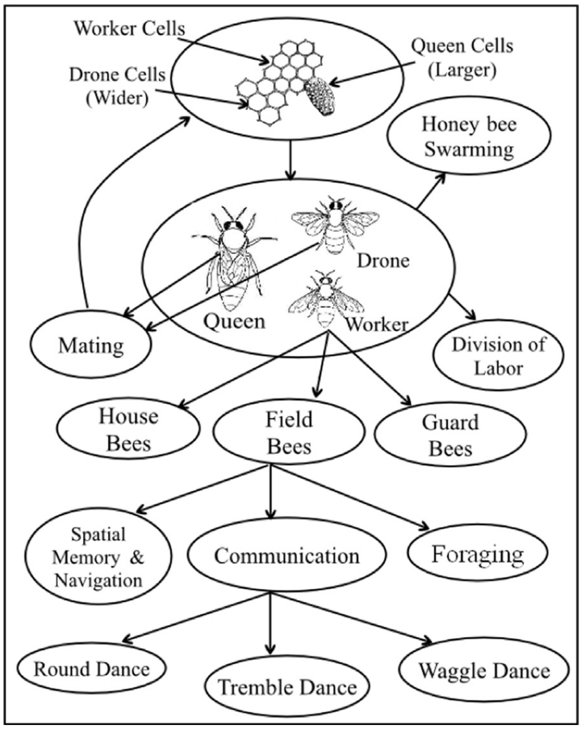

Fig. 1. The taxonomy involved in the life cycle of honey bees.

2.2.2. Communication

A bee returned to the hive from a good food source performs a kind of 'dance' on the vertical comb. In general these are given particular name based on the profitability and distance of the flower (source). For instance, if the distance is beyond 150 m the bee performs *waggle dance*, a particular figure eight dance to share the information regarding the direction and distance of the food sources. Other workers follow the dancing bee and then learn the direction and distance of the food source based upon the dancing behavior. A waggle dance with a very short waggle run used to be characterized as a round recruitment dance. Forager honey bees of the species *Apismellifera* can perform tremble dance to recruit more receiver honey bees to collect nectar from the workers. tremble dance is similar to the waggle dance, but it is used by a forager when the foraging bee perceives a long delay in dumping its nectar or a shortage of receiver bees.

2.3. Honey bee swarming and nest-site-selection

Swarming is often considered to be the process in which bees abandon their hives and collectively move to perform various life-supporting activities. One situation includes rapid reproduction of honey bees in the colony resulting in the increase of population. The reasons may include abundant availability of nectar and favorable environmental conditions. On other time it may happen that the honey flow has been restricted or ailments which forces the worker bees to leave the hive. As soon as they emerge as a swarm they aim to construct a new site after deciding the most suitable site. The most experienced scout bees search a large area for a suitable new site. Bees that have found good sites return to the cluster and the strength of their waggle dance becomes proportional (in some sense) to the quality of the nest-site [S10,S11,S12].

2.4. Spatial memory and navigation

Often foragers (worker bees) come across a situation where finding a way between food source and hive becomes extremely difficult due to fractal oriented landscapes and longer distances. Honey bees usually overcome this problem with the help of landmarks and celestial cues. According to Menzel et al. [S13], foragers use a map oriented organization regarding the spatial memory, which is based on the computations of land marks and celestial cues [S14,S15].

2.5. Division of labor

Division of labor refers to biases in the inclination of individuals to perform different tasks within a group [S16,S17]. According to Beekman et al. [S18], the bee colony should divide its workforce and allocate the appropriate number of individuals for each of the many tasks. Even though each individual bee can perform every task in a hive, each bee can also be specialized for a particular task.

The aforementioned taxonomy is a general representation of the life-cycle of bees and particular emphasis was provided only on the phases, which served as a source of inspiration for bee inspired computational methods. Literature study reveals that each phase has played a very important role in developing model that suited for many challenging applications. In the following sections we provide a comprehensive account of the algorithms/methods/systems that are developed based on the behaviors mentioned in the above taxonomy.

3. Computational methods based on Queen Bees, mating, marriage

3.1. Queen Bee Evolution (QBE)

QBE strategy was first proposed by Jung [64] to enhance the performance of Genetic Algorithms (GAs). According to Jung QBE makes GA quickly approach the global optimum by improving the exploitation and exploration capabilities. The basic inspiration is as follows. In a bee colony, *Queen* is considered as a mother for all the bees and given hierarchal preference in every task. In QBE, the fittest bee (representing the best solution) is considered as the queen in each generation, and is allowed to crossover (mate) with other bees selected by the parent (candidate solutions) through a selection rule. This increases the exploitation of GA at the same time this may lead to premature convergence. To overcome this problem and to decrease the premature convergence rate some individuals in the QBE framework are frequently mutated resulting in improved exploration quality of the GA.

The two key differences between normal Evolutionary Algorithm (EA) and QBE are denoted with # symbol in the pseudo-code of QBE provided as Algorithm 1. Usually in GA the parents $P(t)$of NP individuals are selected by a selection mechanism like roulette wheel or tournament selection. On the other hand, in QBE, parents $P(t)$ consists of half of the NP couples of queen bee $I_q(t-1)$ ($q=argmax\ f_i(t-1)$, $1<i<NP$) and each selected bee $I_m(t-1)$ ($1<m<2/n$). The other difference is that in EAs all the individuals are mutated with problem dependent mutation probability P_m, while in QBE, some individuals are mutated with normal probability $P_m(<0.1)$ and others are with the strong mutation probability P_m' ($P_m' > P_m$).

Algorithm 1. Queen Bee Evolution.

// t=time; NP=population size; P=individuals; ξ=Normal mutation rate; P_m=Normal mutation probability; P_m'=Strong mutation probability; I_q, I_m = Queen and selected bee//

$t=0$
Initialize and evaluate $P(t)$
While (termination criterion is not satisfied)
Do

Please cite this article as: A. Rajasekhar, et al., Computing with the collective intelligence of honey bees – A survey, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.06.001

t=t+1
Select $P(t)$ from $P(t-1)$ (#)
$P(t)=((I_q(t\text{-}1), I_m(t-1)))$
Recombine $P(t)$
5. **Do** crossover
6. **Do** Mutation (#)
7. **For** $i = 1$ to NP
If $(i <= (\xi.NP))$
Do mutation with P_m
Else
Do mutation with $P_m^{'}$
End if
End For
8. Evaluate $P(t)$
9. **End**

Many QBE variants were proposed for solving optimization problems and also were used in conjunction with various engineering applications. Some noteworthy variants and applications include economic power dispatch problem using QBE by Qin et al. [139]; design of fuzzy knowledge base controller using QBE with weighted crossover operator [12,13,77]; parameter extraction of organic thin film transistors compact model using QBE by Moreno et al. [109]; designing DNA sequences that satisfy combinatorial and thermodynamic constraints using Bee Swarm Genetic Algorithm (BSGA) [195]; GA based on Multi-bee Population Evolutionary (BMGA) by Lu and Zhou [90]; Queen-Bee crossover in GA for Label-constrained minimum spanning tree problem by Xiongm et al. [192];Tuning the parameters of Boost Converter using Queen-Bee-Assisted GA by Sundareswaran et al. [162]; Tuning the scaling factors of fuzzy logic control using modified queen-bee-based genetic algorithm (MQBGA) by Mohammad et al. [S258]; Designing fuzzy controller using artificial DNA assisted queen bee genetic algorithm (DNA+QBGA) by Ming et al. [S259]. Vakil-Baghmisheh et al. [179] developed a hybrid algorithm that combined QBE algorithm with the Nelder-Mead algorithm to design PID controller gains for a Gryphon robot; Jie-Sheng et al. [181,182] proposed a new Bee Evolutionary Genetic Algorithm (BEGA) to solve the aircraft landing scheduling problem and vehicle routing problems. In BEGA, self-adaptive crossover operator is adopted to increase solution accuracy and to avoid premature convergence. Huang et al. [107] improved the performance of BEGA by incorporating parameters such as self-adaptive selection operator (to limit the size of random population in each generation) and a controlling operator which extends the biodiversity of the swarm to avoid premature convergence.

3.2. Marriage in Honey Bees Optimization (MBO)

According to Rinderer and Collins [S19], a honey bee's behavior is the interaction of (i) genetic potentiality, (ii) physiological and ecological environments, (iii) the social conditions of the colony, and several ongoing interactions among these three. Each individual bee performs sequences of actions unveiled according to environmental, genetic and social regulation. The outcome of each action greatly influences the next subsequent actions of both, a single bee and the rest in the colony. Marriage process is difficult to understand because the queens mate during their mating-flight far way in front of the hive [S20].

Marriage in Bees Optimization (MBO), proposed by Abbas [1] in 2001, mimics the honey bee evolution. The evolution begins with a solitary colony with a single queen without a family and then emerges to an eusocial colony comprising one or more queens with a family. The MBO models the marriage behavior of the honey bees. The artificial bees can be classified into queens, drones and workers. The queens and drones represent the solutions and workers represent the local search heuristic for improving the offspring solutions. The mating process between the queen and the drone encountered probabilistically during the flight, followed by feeding process by the worker. At initialization, bee population is randomly generated and the solutions with better fitness values are selected as queens and the rest as drones. Then, mating process can be probabilistically determined using the following expression:

$$P(Q, D) = e^{-d/s}. \tag{1}$$

Here d is the absolute difference between the fitness of D and Q, and $s$is the speed of the queen Q. $P(Q,D)$ represent the successful mating probability. If the mating is successful, a sperm of the drone is added to the queen's spermatheca. The mating flights are repeated until the energy of the queen is lower than the predefined threshold or the queen's spermatheca is full. After each transition, the queen's speed and energy are decreased according to the following equations:

$$speed(t + 1) = \alpha.\ speed(t), \tag{2}$$

$$energy(t + 1) = energy(t) - step, \tag{3}$$

where α is a scaling factor and lies between 0 and 1,*step* is the amount of energy reduction after each successful mating or transition. Algorithm 2 provides the set of heuristics to be performed in MBO approach. After completing the mating process, the new broods (offspring solutions) are generated using crossover operation of the queen and randomly selected drone solutions. The breeding process is repeated until the number of broods reaches the required value. Afterward, a binary mutation is applied to improve the genotypes of newly-born broods.

Then, feeding of offspring solutions produced by queens are carried out by workers. As the worker action is constrained to only breeding (nursing), worker can be represented the local search heuristic functions. The workers are selected according to their fitness values and then applied to enhance the offspring solutions.

After breeding process, the weaker queens are replaced with the fitter broods until there is no more brood fitter than any queens. The remaining broods with lesser fitness are killed before starting the new mating flights. The procedures are shown in Algorithm 2.

Algorithm 2. Marriage in Honey Bee Mating Optimization.
1. Initialize the workers and randomly generate queens
2. Apply local search (greedy criterion) to get best queen (in terms of fitness)
3. **For** $f = 1$ to Max number of mating flights
For $i = 1$ to Max number of queens
Initialize energy, speed and position
Queen moves between states randomly
Drone is selected probabilistically using Eq. (1)
If (queen selects a drone)
Sperm is deposited in spermatheca of queen
End
Update queen's speed and energy using Eqs. (2) and (3).
End For
Generate new broods by crossover and mutation operations
Employ workers to improve the broods
Update workers fitness
While (best brood > worst queen)// fitness

Replace the queen with best brood
Eliminate the best brood from brood family
End While
4. End For

The MBO model found a profound use in applications such as different satisfiability problems by Abbass [1,S21,S22,S23]; Optimization of partitioning and scheduling by Koudil et al.[81] and solving infinite horizon-discounted cost stochastic dynamic programming problems by Chang [23] (termed as Honey-bees Policy Iteration (HPI)).

Teo and Abbass [168] proposed a hybrid MBO with annealing function. In the proposed method, the queen's mating flight trajectories are determined based on the probabilistic function of the queen's fitness. A conventional annealing function is used in this context to intensify the search. Abbas and Teo used this hybrid method to construct the pool of drones in [S24].

Yang et al.[197] proposed faster marriage in honey Bees optimization named FMBO algorithm with global convergence behavior. In FMBO, the computation became easier and faster by initializing the drones randomly and restricting the conditions of iterations. To improve the performance of MBO, Yang et al. [S25] judiciously merged MBO with the Nelder-Mead method and applied to solve the Traveling Salesperson Problem (TSP). Yang et al. [S26] proposed a hybrid method involving the Wolf Pack based local search known as WPS-MBO for solving a few complex optimization problems and also TSP as well. Vakil-Baghmisheh and Salim [180] proposed a Modified Fast Marriage in honey Bee Optimization (MFMBO) and validated the performance of the proposed approach on various numerical benchmarks. Further the authors compared MFMBO with other bee algorithms like ABC, QBE, and FMBO methods.

Vakil-Baghamisheh et al. [S27] used this MFMBO to predict the structure of the Met-enkphaline using torsion angles representation and ECEPP/3 energy function. Salim and Vakil-Baghmisheh [150] presented three versions of the Discrete Fast Marriage in honey Bee Optimization (DFMBO1, DFMBO2, and DFMBO3) algorithm and discrete ABC algorithm using three logical operators - OR, AND, and XOR. The performance of their proposed methods was validated using four benchmark functions and was also compared with ABC, QBE, and FMBO. Ping et al. [122] used MBO algorithm via support vector regression models for forecasting output of the integrated circuit industry.

Thammano and Poolsamran [174] proposed a new Self-organizing model of MBO (SMBO) to improve the performance of MBO. Unlike the original model, SMBO automatically searched for the proper number of queens. The problem space was partitioned into several colonies with its own queens and later the queens were encouraged to compete for a larger colony. Thereafter a greedy mechanism was used between the newly born brood and the queens. The fuzzy c-means clustering algorithm was employed to assign the drones to the proper colonies. Performance of SMBO was validated on six benchmark functions and compared to the original MBO. The authors also applied SMBO to solve multi-objective optimization problems and its performance was compared with two well-known approaches, the Pareto archived evolution strategy and the non-dominated sorting genetic algorithm (NSGA) [S226].

3.3. Honey Bees Mating Optimization (HBMO) Algorithm

HBMO was introduced by Haddad et al. [50,S28,S29] and it is a meta-heuristic approach similar to MBO in terms of crossover and genotypes. This algorithm has five main stages: mating flight, new brood creation, performing local search on broods via workers, adapting worker's fitness based on the improvement obtained on broods and finally replacing weaker queens by the broods with better fitter values. A pseudo code of this algorithm is provided as Algorithm 3. Similar to MBO and QBE, HBMO had been successfully applied to solve many real-world optimization problems which are listed in the Table 3.

Amiri and Fathian [9] presented a two stage method in which HBMO was integrated with Self-Organizing Feature Map (SOM) neural network to solve the clustering problem. SOFM was used to determine the number of clusters and cluster centroids and then HBMO method based on the *k*-means algorithm ((HBMK) was employed to find the final solution. SOM with HBMK was compared with SOM with *k*-means and SOM with genetic *k*-means via a Monte Carlo study. Fathian and Amiri [S33,S34] applied the HBMOK algorithm to solve an internet bookstore market segmentation based on customer loyalty.

Marinakis et al. [98–100,S36,S37,S38] proposed the HBMO algorithm to handle the Vehicle Routing Problem and termed as HBMOVRP. HBMO algorithm is combined with multiple phase neighborhood search-greedy randomized adaptive search procedure (MPNS-GRASP) and Expanding Neighborhood Search (ENS) algorithm. The algorithm slightly modified the crossover operator and the workers' heuristic nature. This particular hybrid method was successfully applied to various TSP problems.

Algorithm 3. Honey bee Mating Optimization.
1. Start
2. Initialize the model and algorithm input parameters:
3. Randomly generate a set of initial solutions and rank the solutions based on the objective functions keeping the best one as queen and other ones as drones
4. Use simulated Annealing to select the set of solutions from the given search space to make a mating pool for generating new trail solutions and replacing them with previous best solution if they are fit
5. Generate a new set of solutions (breeding process) by employing different predefined crossover operators and heuristic functions (workers) between the best present solution (queen) and trail solutions based on their fitness values obtained
6. Improve the newly generated set of solutions utilizing different heuristic functions and mutation operators according to their fitness.
7. Sort new solutions in accordance with their fitness values, selecting the best new solutions and best new trail solutions for next iteration
8. IF (new best solution better to previous one)
Substitute the best solution
9. **Else**
Keep the previous best solution
10. **If** termination criterion not satisfied go to step 5
11. Else report the best solution and terminate.

In the area of power systems and related, Niknam et al. [117] presented an efficient multi-objective HBMO for multi-objective distribution feeder reconfiguration problem (Niknam [S40], Niknam and Sadeghi [S41]). In MHBMO, several queens are used and the queens are considered as an external repository to save non-dominated solutions found during the search process. The objective functions are based on fuzzy clustering technique, particularly to control the size of the repository within given boundaries.

Niknam also proposed a new Chaotic Improved HBMO (CIHBMO) [115] based on the logistic and tent equations. CIHBMO is also used for solving multi-objective daily Volt/Var control in distribution systems and in non-smooth economic dispatch problem

Table 3
Summary of applications of the bees inspired algorithms.

Sub areas and problem	Types of Bees Algorithms used and References
Hybridization	
Queen Bee	Queen Bee with GA [64], Improved Bee Evolutionary GA (BEGA) [107]
Honey Bee Mating Optimization	HBMO with simulated annealing [S261]
Marriage in Honey Bees (MBO)	MBO with annealing approach [168], MBO based on Nelder Mead [S25], MBO-Wolf pack search method [S26]
Bee Colony Optimization (BCO)	Simulated Bee Colony Optimization [S57]
Artificial Bee Colony	ABC-PSO [S115], Differential ABC [S140],Hybrid Quantum evolutionary algorithm (QEA) with ABC [37], G-best guided ABC [203], Memetic ABC using Nelder Mead [39], Simulated Annealing based ABC [24]
Modification of Bee intelligence Based on Heuristic Methods	
MBO	Fast-MBO (FMBO) [197], Modified FMBO [180]
BCO	BCO enriched with elitism, local optimization and adaptive pruning [190]
ABC	ABC with Banach spaces [140], ABC involving hyper-mutation and a novel crossover operator [88],Co-operative approaches oriented ABC [38,S103], Adaptive control mechanism in ABC [5], Elitist ABC [106], Smart flight and dynamic tolerances in ABC [105],ABC with ripple communication strategy [95], Parallelization of ABC [126], Adapting of onlooker bees [2], Best-so-far selection ABC [15], ABC embedded with Deb's heuristic rules [71], RosenbrockABC [65], Parametric tuning of ABC [S77],ABC with exponentially distributed mechanism (ABC-EDM) [S111], Fast Mutation ABC [S134], Powell method based ABC [44], ABC with Saccadic Flight Strategy
Bees Algorithm (BA)	BA with fuzzy greedy selection criterion [128], BA with pheromone-based recruitment [128], Improvised initialization oriented BA [S171], parallel bees algorithm [93]
Bee Swarm Optimization (BSO)	Parallelized BSO [146]
Electrical Engineering	
Economic Dispatch	Modified QBA [118], HBMO [S42], BCO [16], ABC [S78, S84, S94, S95], BA [82,S177], BA with Pareto Dispatch [S162], Fuzzy HBMO [45], Hybrid BCO and Quadratic Programming [17], Enhanced BSO [S257], Incremental ABC [S240], ABC with Dynamic Population [S241], GbestABC [S247]
Boost Converter & PI/PID Controller, inverter	Queen Bee Assisted GA [162], Hybrid Genetic ABC [S125], Cauchy Mutated ABC [142], BCO [68]
Capacitor Placement, compensator	Queen Bee Assisted GA [S214], Virtual Bee Algorithm [S188], Bees Algorithm [S176]
Distribution System's network configuration	HBMO [S43], ABC [143] [S127], Modified ABC [S91], Priority ordered constrained search along with ABC [S129], Hybrid HBMO and Fuzzy sets [117], Chaotic Improved HBMO [115], Efficient Multi Objective HBMO [S40, S41], fuzzy method based on HBMO [116], A hybrid HBMO and Discrete PSO [114], BCO [S65]
Optimal Power flow	ABC [S215], Hybrid DE-ABC [S250]
Load Profile Clustering	HBMO [S45]
HVDC and FACTS	BA [S168], Multi-objective BA [60]
Fault location	Enhanced HBMO [58]
Solar Cells	Artificial bee swarm [11]
Engineering Design, Control System and Robotics	
Fuzzy Controller in Robotic Systems	Modified queen bee evolution based GA [13], A Novel Parent Selection Operator in GA [12], BA [S154,135]
PI/PID Control	Queen Bee with nelder-mead [179], ABC [S112],
Chaotic and Non Chaotic Systems	ABC [S113, S110, S111]
Automatic Voltage Regular Systems	ABC [S146]
Robot Motion path planning and navigation	ABC [S216], HBMO [S32]
Aircraft control	Chaotic ABC [194], Evolutionary Genetic Algorithm and Clustering Method [182]
Robotic Arm	Bees Algorithm [S151, S165, S167].
Engineering and Mechanical Design	Bees Algorithm [S156, S157, S159, S166, 124, S172], Artificial Bee Colony [4]
Machining and milling Process	ABC [S79, S80, S81, 94, S108]
Container over loading problems	hybridized heuristic filling procedure with BA [35]
Swarm Robots	Distributed BA [S252]
Civil Engineering	
Vehicle Routing	Improved BEGA [181], HBMO algorithm with Multiple Phase Neighborhood Search-Greedy Randomized Adaptive Search Procedure (MPNS-GRASP) and the Expanding Neighborhood Search (ENS) [99,S36], Integrated Bee and Fuzzy System [S52], decentralized multi-agent approach (termed BeeJamA) [S199]
Transportation Engineering	Bee System [91,92,S50], Multi-Agent Systems [S51], BCO [169,170], BCO [S233]
Traffic Mitigation, Ride Matching Problems,	Fuzzy Bee System [S56], BCO [S60]
p-center problem	BCO [172,S63]
Traveling Sales Man Problem (TSP), Probabilistic TSP, Euclidean TSP	BCOA [S69], BCOA with the Fragmentation State Transition Rule (FSTR) [188],An efficient BCO with Frequency Based Pruning [189], BCO with local search [S70], Greedy Subtour Crossover method with ABC (ABC-GSX) [S107], HBMO involving GRASP and ENS [98,S39], Fast Bee Colony [S229], Hybrid Ant Bee Colony [S230]
Optimal Reservoir operation	HBMO [S28, S29,51]
Water Distribution Networks	HBMO [50,156,S30,S31]
Structural Analysis	NelderMead Simplex combined with ABC [66], BA [108]
Design of Truss structures	ABC with Adaptive Penalty function [157], Modified ABC [158,S132], Parallel vector evaluated type, swarm intelligence multi-objective variant of ABC (VEABC) [118]
Multi-zone dispatch systems	BA [S170]
Weighted Ring Arc-Loading Problem	Efficient traffic loading algorithm-ABC [S121]
Pattern Recognition and Image Processing	
Data Clustering, Mining& Feature Selection	HBMO [S33,S34], HBMO along with GRASP [100], ABC [73,202,S99], Chaotic ABC [S143], BA [132,S158], ABC and GRASP [S86], BSO [S255]
Fuzzy Clustering	BA [S161]
Image Compression	HBMO- Linde-Buzo-Gray (LBG) [55]
Adaptive Image Restoration	ABC [S142]
Active Contour Model	HBMO [56], HBMO-SNAKE [S260]

Table 3 (*continued*)

Sub areas and problem	Types of Bees Algorithms used and References
Fractal image compression	Local no-search scheme along with HBMO [3]
Image analysis and pattern recognition	HBMO [S47], ABC [S90]
Image segmentation,	Fuzzy entropy and Bee Colony Algorithm [S58], ABC [96]
Segmentation of MR brain images	FCM improved ABC algorithm [S109]
Target recognition for aircraft	ABC optimized edge potential function (EPF) approach [193]
Image Edge Enhancement	ABC [S88,S89]
Multilevel Image Thresholding	ABC [S100,54]
Blind image de-convolution	BA [S253]
SVM	Dynamic ABC [S239]
Signal Processing	
Communication signals recognition	BA and MLP neural [154]
Blind signal-type classification	BA [48]
Routing and wave length assignment	BCO [S53]
PAPR Reduction of OFDM signals	ABC [183]
Electronic Engineering	
Intrusion detection system	ABC-support vector machine algorithm [58]
CMOS inverter, Multilevel inverter	ABC [45,S104], BA [78]
Multiple Sequence Alignment problems	ABC [17,33,S105,S106]
Multiplier-less non-uniform filter bank trans multiplexer	ABC [97]
Loney's solenoid benchmark problem	Gaussian ABC [29,S136]
Electronic devices and circuits	ABC [145,S146], Queen Bee Evolution [195], improved ABC [S244], modified ABC [S269]
Design of digital filters	ABC [74,S123], BA [131]
Antennas	ABC [S138, S139], ABC with inverse fast Fourier technique [S139], BA [47,48,152]
PCB assembly optimization	BA [S159], BA enhanced with TRIZ operators [S174], ABC [S131]
OFDM	ABC [183], ABC enhanced with Tabu search [53]
Computer Science Engineering	
Software test optimization framework	ABC [63,S117], BCO [S232]
Task allocation problems	HBMO [67], distributed BA [62]
Reliability redundancy allocation	Penalty guided ABC [199]
mobile and ad hoc networks (MANETs)	peer-to-peer file searching with BA [S178], Bee Adhoc [185], BeeAIS with Dendrite cell approach [S184]
Routing protocols	BeeSec [S182], BeeAIS [101], BeeSensor [148], Bee Adhoc [149,S183]
Exploring of information	Bee Adhoc [S181], BSO [S187], Approach based on information sharing and processing models of honeybees [S193], BeeHive Metaphor [S200, S201,112]
Max weight satisfiablity	Cooperative BSO [36]
Satisfiability Problems (SAT)	MBO with GSAT and random walk [S21], A single queen bee based MBO [S22], MBO with annealing approach [S23], Multilevel BSO [S256]
Knapsack problem, Quadratic Knapsack problem, Multi-dimensional knapsack	ABC [S82,S83,S102]
Set Covering Problem (SCP)	ABC with local search [161]
Graph coloring problem	Hybrid ABC [40]
Server allocation in internet hosting centers	Division of labor phenomenon [110,111,S211]
Improving efficiency of software agents	Honey Bee Team work architecture [147,S213]
Stochastic dynamic programming	Honey-Bees Policy Iteration [23]
Spanning Tree, Quadratic Spanning Tree	QBE crossover with GA [192], ABC [155,160,S137]
Sensor Networks	ABC [S85, S86, S96, S97, S98, S249]
Scheduling	BCO [33,163], Reserved BCO [S231]
Sequence Planning	HBMO [S227]
GPU	Parallel Bees Algorithm [93]
Market Segmentation and Scheduling	
Market Segmentation	Integration of self-organizing feature maps and HBMO [9], HBMO-PSO [26]
Environmental integrated closed loop logistics model	ABC [S120]
Scheduling	BCO [32,S61], Parallelized BCO [S62], BA [S153], BCO algorithm with idle-time-based filtering [59], Discrete ABC [123,167], [S135], BCOA [27,28], BCOA with big valley land scape [191], ABC [81,S116], ABC with Random Key [S122], BA [204], two-stage ABC [42], Effective ABC [125], discrete ABC [124], Hybrid ABC [83]
Bio Informatics	
DNA sequences	Multiobjective ABC [S141], BA [S152]
Protein related and structure	ABC [S92,S93], Parallel ABC [S126], Modified MBO [S27], BCO [S228]
Protein conformational search	BA [S160], BCO [41]
Artificial Neural Networks	
Training of Feed-forward ANNs	ABC: [70,S73], BA [136], Complex Neural Fuzzy System [S142]
Medical Pattern classification problems	ABC [S74,120]
Evapotranspiration problem	ANN-ABC [119]
S-system models	ABC-NN [200]
Wavelet Latin Hypercube Sampling	Bee Recurrent Neural Network (BRNN) optimized by ABC algorithm with Monte Carlo Simulation [201]
Bottom hole pressure prediction	hybrid ABC-back propagation strategy (ABC-BP) [61]
Wood defects	BA [S148], BA with SVM [S155], Hybrid PSO-Bees Algorithm [134]
Control chart pattern recognition	BA [S149, S150]
Forecasting output of IC industry	Marriage in honey-bees optimization algorithms [122]
Financial forecasting problems	HBMO [S39], ABC-Recurrent Neural Network [57]
Nuclear Engineering	
Fusion reaction	ABC enhanced with grid type computing environment [S119], Distributed and Asynchronous BA [S179]
Fuel Management Optimization	ABC with Random Keys (ABCRK) [34]

Please cite this article as: A. Rajasekhar, et al., Computing with the collective intelligence of honey bees – A survey, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.06.001

Table 3 (*continued*)

Sub areas and problem	Types of Bees Algorithms used and References
Others	
Multi-objective problems	Multi-objective HBMO [117], Chaotic Improved HBMO (CIHBMO) [115,S42], CIHBMO with local search [116], Pareto BCO [S55], multi-objective BCO [89], Multi-objective ABC [124,S79, S80, S81, S95 S118,S141], Pareto-based ABC [85],Vector Evaluated ABC [118],Multi-objective BA [19,60,82,129,152], Binary Bees Algorithm [196], BA with Pareto Dispatch [S162], Pareto based BA [S167], Modified MBO [S226s]
Constrained Optimization	BEGA [182], MBO [1,S21, S22, S23], HBMO [S31] ABC [4,5,105,106,155,S72,S122,S129], modified ABC [71], BA [204], Face Constrained Problem: hybrid ABC and BA [176]
Multimodal & Dynamic Optimization	ABC [S75, S140], BA [S123, S170], Labor division [147], Honey bee behavior [186], Honey-Bees Policy Iteration [156], Differential-ABC [S140], BeesAdHoc [101], Honey bee foraging [14,S68,S191]
Stochastic optimization problems	BA [S163]
Multivariate	Improved ABC [140]
High dimensional and large-scale	HBMO [99], ABC [4,S75,S76], Bees navigation [20]
Multidimensional	ABC: Numerical Function Optimization [72], Engineering Design Problems [4] BA: Numerical Function Optimization [S147], Environmental/Economic power dispatch [82], orthogonal frequency division multiplexing [152], optimal allocation of FACTS [60]
Miscellaneous	Portfolio optimization: BCO [S54], ABC [S124], Two level distribution problem: Bio inspired Bees Algorithm [196], Direction Finding of ML Algorithm ABC [S113], Sudoku and NP Complete Problems: ABC [S87] Direction Finding: ABC [S114], Inverse Heat Conduction: ABC [S130], Non-Traditional Machining: ABC [S145], Coordination of overcurrent relays: BA [S173], Nurse Rostering: BCO [175]; timetabling problems: Self-Adaptive BA [S254], Plant Cycle Location Problem [S35], Power System State Estimation [S43,113]

in the field of power systems [S42]. Niknam et al. [116] presented a new interactive fuzzy method using modified HBMO for solving the Multi-objective Optimal Operation Management (MOOM) problem. In this method, the modified HBMO is combined with chaotic local search method to prevent premature convergence. Niknam [114] also hybridized discrete particle warm optimization with HMBO for solving the multi-objective distribution feeder reconfiguration optimization problem.

Huang [58] proposed Plant Growth Simulation (PGS) algorithm enhanced HBMO to improve the local search capability and overall computation of HBMO. The authors used their approach to estimate the fault section in a power system in practical scenario, and the results indicated that enhanced HBMO was superior to comparative algorithms and further it also avoid premature convergence.

Ghasemi [S230] applied a fuzzified multi-objective interactive HBMO to the environmental/economic dispatch problem. The approach was quite different from Niknam et al.'s HBMO method [116] involving fuzzy clusters. In multi-objective IHBMO, the dominated and non-dominated solutions are sorted by using Pareto dominance and the best solution is extracted using fuzzy set theory. The performance of the multi-objective IHBMO was tested using IEEE-30 and IEEE-180 bus systems. The authors verified the superiority of IHBMO over other multi-objective optimization algorithms in terms of convergence characteristics and Pareto sets obtained.

In image processing area, Horng et al. [56] applied HBMO-based snake algorithm to improve the detection of the concave region connected with control points of active contour. Li proposed improved HBMO-SNAKE algorithm by adjusting the step and replacing the model parameters in [S260]. Afarandeh et al. [3] proposed a new technique which used of no-search scheme and local search with HBMO. The proposed method was applied to fractal image compression and gained superior performance over other compared fractal encoding algorithms.

Bernardino et al. [S44] applied a hybrid version of HBMO to connect a given set of terminals to a given set of concentrators and to minimize the link cost to form a communication network. Chiu et al. [26] proposed a hybrid heuristic which integrates PSO and HBMO methods for market segmentation that segments data into homogenous clusters by using cluster analysis.

Yuan [S227] proposed Hybrid HBMO for assembly sequence planning problem in which multipoint precedence crossover was used to avoid from generating infeasible solutions and to preserve the queen and broods' meaningful characteristics. The author also presented a novel hybrid HBMO algorithm in which a new encoding and decoding schemes were designed to fit the problem and simulated annealing was employed to improve the broods. Shayeghi proposed parallel vector evaluated improved HBMO (VEIHBMO) to tune the parameters of Power System Stabilizer (PSS) in [S262]. The experimental results validated that VEIHBMO algorithm performs superior to bacteria foraging algorithm and cultural algorithms in PSS tuning optimization problem.

4. Methods involving bees foraging and communication

4.1. Foraging involving communication based Methods

Stumper and Broomhead [S48] laid the first steps to describe the nectar foraging of honey bees algebraically. Their work later played a key role in the development of computational methods for solving different problems based on the characteristics of bees foraging. This idea also led to the origination of various optimization algorithms, inspired by the honey bees' foraging behavior. During the last decade there have been several algorithms of this kind. Some of the well-known heuristics based on foraging and communication of bees that played a vital role in designing or optimizing the real world applications are summarized in this section.

Methods Based on Foraging.

1. Bees System (BS)
2. Bee Colony Optimization
3. Bee Colony Optimization Algorithm
4. Artificial Bee Colony
5. Bees Algorithm
6. *BeeAdhoc*
7. Bee Swarm Optimization
8. Virtual Bee Algorithm
9. Miscellaneous Methods resembling Bee's Foraging nature

4.1.1. Bees System (BS)

In 1997 Sato and Hagiwara [151] proposed a first computational algorithm based on bees' foraging behavior and called it *Bee*

System (BS). In this approach each bee is considered to be a chromosome and is employed to find the solution. Some chromosomes having considerable fitness are regarded as superior chromosomes and others try to find solution around the vicinity of the superior ones by generating multiple solutions.

In an alternate development, Lucic and Teodorovic [91] explored bees' behavior and developed the models to tackle the combinatorial optimization problems. In this BS, (different from that of Sato and Haigwara [151]), every bee colony has scouts and they are the bee colony's explorers. There is no guidance for the explorers when searching for food. Therefore, the scouts/explorers are characterized by low search costs and a low average in food source quality. However, in course of time, the scouts can accidently discover the rich, entirely unknown food sources. In case of solving the difficult combinatorial optimization problems, the bees discovered the groups of the feasible solutions quickly. Among these feasible solutions, some of them could be proved as the very good quality solutions. The survival and progress of the bee colony depends on the rapid discovery and efficient utilization of the best food sources. The BS and its variants were mainly applied to traveling salesman problem in [92,S49–S52].

4.1.2. Bees Colony Optimization (BCO)

Inspired by the honey bees foraging behavior, Teodorovic and Dell [170] proposed a powerful metaheuristic algorithm, called bee colony optimization (BCO). BCO employs *artificial bees* as agents, which collaboratively solve the complex combinatorial optimization problems. BCO [169,S59,S64] is a more generalized and improved version of the BS algorithm [91] and hence in this context we consider both of the algorithms more or less similar.

The BCO algorithm is mainly divided into the two processes: *forward pass* and *backward pass*. The main idea behind the algorithm is to exploit and explore the search space so that optimal solution can be found in predefined steps. In each *forward pass* phase, each agent explores the search space with a predefined number of moves which is applied to construct and/or improve the solution. Once the bees are acquainted with new partial solutions, they go back to the hive and start the *backward pass* phase.

In backward pass, the artificial agents share information about the quality of their solutions via waggle dances. With a certain amount of probability value, the bees decide whether to abandon the created partial solution to become a uncommitted follower or to dance and recruit the remaining nest mates before they return to the patches or create partial solution. This makes the bees with high fitness value to continue exploration once again in the same area. Thus, once again the bees start to forage. Subsequently the forward pass phase takes place and the bees expand the previously created partial solutions. Then they perform backward pass and this process continues in an iterative fashion until a predefined termination criterion is satisfied. Every follower chooses a new solution from the recruiters using the roulette wheel selection. A brief pseudo-code of BCO has been presented in Algorithm 4.

BCO applications to numerous real world and computationally complex problems are listed in Table 2.

Algorithm 4. Bee Colony Optimization.

//*B* – Total number of bees in the hive/
//*NC* – The number of constructive moves during each forward pass
1. Initialization
2. **For** *I* iterations **do**
3. **For** all *B* bees do
   i. **Set** $i=1$ // counter for number of constructive moves in the forward pass
   ii. Evaluate all possible constructive moves;
   iii. According to the fitness obtained choose one move by using the roulette wheel selection.
   iv. $i=i+1$ if $i<=NC$ **Go** to step (ii)
4. **End For**
5. All bees return back to the hive
6. Sort the bees using their fitness values obtained
7. **For** all bees *B* **do**
   i. Backward pass
   ii. Every bee decides randomly whether to become a recruiter or to become a follower via dances and fitness sharing
   iii. For every follower choose a new solution by roulette wheel basis
8. **End For**
9. If best solution obtained in iteration is global best, Update best-known solution
10. **End For**
11. Display the best result.

Contrasting to several attempts made by contemporary researchers in developing various enhancement strategies to BCO, Tereshko, Lee and Loengarov [173,S66,S67] developed a honey bee foraging model based on the reaction-diffusion equation. The authors investigated the bee communication behavior and analyzed the mapping of information about the explored environment. The empirical study of BCO was performed in [S234] and the experiment was conducted using several numerical test functions. The experimental results were compared against particle swarm optimization, GA, differential evolution and artificial bee colony algorithms. The study confirmed that BCO is able to find high quality solutions within reasonable CPU times.

BCO was mainly used in scheduling problems. Few notable approaches are as follows. Banarjee et al. [S55] combined BCO with rough set based approach for modeling process and supply chain scheduling which had multiple conflicting criteria. Teodorovic et al. [171] proposed a new metaheuristic termed as fuzzy bee system (FBS) in which the artificial bees communicated each other using the approximate reasoning and fuzzy logic rules. FBS was applied to solve the combinatorial problems. Teodorovic and Dell'orco [S56] used this FBS system as travel demand management technique for solving the ride matching problem.

McCaffrey [S57] used a simulated BCO in generation of pairwise test sets with a minimal size. Yonghao et al. [S58] used BCO along with fuzzy entropy in image segmentation. Davidovic et al. [32,S61] applied BCO to the static scheduling problem. The goal of BCO was to find an optimal schedule of the independent tasks of identical machines with a minimum completion time for all tasks. Davidovic et al. [S62] proposed parallelization strategies of BCO and validated on identical machines problem. Yueh-Min Huang et al. [59] proposed a new BCO with idle-time based filtering scheme for solving the open shop scheduling problems, to reduce the problem solving time. Taheri et al. [163] developed a novel Job Data Scheduling (JDS) algorithm using Bee Colony (BC) called JDS-BC to apply to simultaneous job scheduling and data replication in grid environments. Three different sizes of test-grid were used to test the performance of JDS-BC.JDS-BC provided invaluable insights into data-centric job scheduling for the grid environments. In [S231], reserved BCO was applied to grid scheduling to allocate the resources properly. BCO was also used for scheduling the independent tasks on homogeneous multiprocessor systems in [33].

In [S229], the authors modified BCO by pattern reduction (PR) and in [S230] by hybridizing with ACO to reduce the computational time of solving TSP. In the parallel lines, Teodorovic [172] and Davidovic et al. [S63] used a new improved BCO variant (BCOi) for the *p*-center problem (Transpiration Engineering) and showed that BCOi provided better results in less computational time.

Forsati proposed improved BCO (IBCO) algorithm to address the document clustering problem in [S263]. In IBCO, the cloning and fairness concepts were introduced to increase the explorative power of BCO algorithm. The IBCO algorithm was applied to data clustering and introduced IBCOCLUST. The performance of IBCOCLUST algorithm was tested using large and high dimensional sets and compared with original BCO, *k*-means, GA and PSO based clustering algorithms to show the superiority of IBCO over the competitors.

To enhance the exploitation ability, Moayedikia introduced weighted BCO (*w*BCO) algorithm in [S264]. In *w*BCO, global and local weights were considered, which allowed the bees to search deliberately through the solution space. A new recruiter selection was also adopted to preserve the population diversity. In [S264], *w*BCO was employed to tackle feature selection (FS) problem. The performance of *w*BCO and FS-*w*BCO were evaluated using several benchmark functions and datasets and compared with other BCO algorithms.

BCO had been used for various applications such as Genetic Clustering for Flexible Protein-Ligand [S228], automated Software Testing [S232], Transit Network Design [S233], Nurse Rostering problem [175], Optimizing Type-1 and Type-2 Fuzzy controller Design [S265], Sequential Ordering Problem [S266], the satisfiability problem in probabilistic logic [S267] and was also adapted to current control strategy of power factor (PF) correction for a Sheppard-Taylor converter [68].

4.1.3. Bees Colony Optimization Algorithm (BCOA)

Similar to BCO, in 2006, Chong et al. [28] introduced the Bee Colony Optimization Algorithm (BCOA) based on honey bee foraging and waggle dance behaviors. Although the basic framework of this algorithm is same as that of BCO, the search heuristics involved are completely different and it is as follows. Typically BCOA consists of a total of n foragers. In the course of iterations each forager f_i constructs a solution and promote their solutions among the foragers. Then according to the quality of its own solution, a forager decides whether to retain its previous solution or to discard it and adopt other forager's solution. Decision is solely left to the female bee and after the decision is made, a new solution is created based on its current solution.

According to Chong et al. [28], bees search their food sources by foraging and waggle dance. Briefly, in BCOA, there are two phases: a dancing phase and a foraging phase. During an iteration, each forager build ups a solution for the given problem. Then with probability p, each forager f_i ($i \in [1, n]$)on returning to the hive from nectar exploration attempts to perform waggle dance within duration $D = d_i A$, where d_i is based on profitability rating and A is the waggle dance scaling factor. In this approach, the profitability rating is defined as *Pfi* and is related to the objective function. *Pfi* can be obtained by the following equation:

$$Pf_i = \frac{1}{C^i_{\max}}, \tag{4}$$

where C^i_{max} represents the fitness value of f_i's current solution. The average profitability rating of all the dancing foragers is given by:

$$Pf_{colony} = \frac{1}{n_d} \sum_{i=1}^{n_d} Pf_i, \tag{5}$$

where n_d corresponds to the number of dancing bees while computing the profitability rating. The waggle dance duration f_i is given by:

$$d_i = Pf_i / Pf_{colony}. \tag{6}$$

Based on the probability r_i, each forager attempts to follow a randomly selected dance of another forager. The probability r_i is dynamic and changes according to the probability rating as shown in Table 1. If the probability ratio is low compared to the colonies, a forager is more likely to randomly observe and follow a waggle dance on the floor. A successful combination of waggle dance and foraging constitutes iteration. The best solution during the iterative process is obtained at the end of iterations. The pseudo-code of BCOA has been shown as Algorithm 4a.

Algorithm 4a. BCOA

1. **For** $i=1$ *to* $Iter_{max}$ //Total number of iterations//
2. **For** $j=1$ *to i* // for every foraging bee //
a. **Forage**
b. **Save the best Solution**
c. **Perform Waggle Dance**
3. **End**
4. **End**

In the above approach the foraging part is not discussed to make the algorithmic description more general and left to the readers' interest which can be found in Chong et al. [28]. Wong et al. [S69] applied BCOA to TSP and compared it with six other optimization methods, thus demonstrating the superiority of BCOA in solving TSP. Wong et al. [191] applied an improved BCOA with big valley landscape exploitation to solve a job shop scheduling problem. Chong et al. [27] proposed a new BCOA algorithm which used an efficient neighborhood structure to find the feasible solutions and iteratively improved on prior solutions. The proposed algorithm was applied for solving job shop scheduling problem [27] and TSP instances [S70]. In this approach BCOA model is constructed similarly to that of the original version. To improve the prior solutions generated by BCOA, the algorithm is integrated with a 2-opt heuristic and the obtained results are tested on a set of benchmarks. The comparative study showcased the superiority of the integrated approach. 2-Opt algorithm was originally designed for finding routes in the traveling salesperson problem, and was later applied to genetic algorithm, simulated annealing, etc. The original 2-Opt local search prohibits routes from self-crossing by reordering them. This provides a good chance of escaping from local optima.

To further improve the performance of BCOA with 2-opt, Wong et al. [189] proposed two mechanisms: Frequency-Based Pruning Strategy (FBPS) and Fixed-Radius Near Neighbor (FRNN) 2-opt. Based on the accumulated frequency of its building blocks recorded in a matrix, FBPS produces a subset of promising solutions to perform 2-opt whereas FRNN 2-opt exploits the geometric structure in a permutation of TSP sequence and the performance is validated on TSP. Wong et al. [188] proposed a hybrid version of BCOA and called it BCOA with the Fragmentation State Transition Rule (FSTR). The FSTR helps BCOA to produce a tour by combining a few fragments of cities. BCOA with FSTR was tested on 84 benchmark problems with different dimensions ranging from 14 to 1379. Wong et al. [190] developed a bee colony optimization framework by including local optimization and adaptive pruning (besides elitism) in the BCO framework. The algorithm was tested on TSP and quadratic assignment problems. Low et al.[89] integrated a multi-objective BCOA into automated red teaming process to find a set of non-dominated solutions effectively from a large search space.

4.1.4. Artificial Bee Colony Algorithm

In 2005, Karaboga proposed a simple algorithm for multivariate and multi-modal continuous optimization problems, known as the artificial bee colony (ABC) [69]. Since its inception, ABC optimization algorithm has received huge attention from both practitioners and researchers on intelligent optimization. In this sub section we begin with the basic version of the algorithm. We then shift our focus on the works that made significant improvements in the algorithm and also on several practical applications of ABC.

4.2. The Basic ABC Algorithm

ABC classifies the foraging artificial bees into three groups namely *employed bees, onlooker bees,* and *scout bees.* The first half of the colony consists of the *employed bees* and the second half consists of the *onlooker bees*. The bees searching for food around the food sources are called *employed bees*. The bees which make decision to choose the food sources found by employed bees are called *onlooker bees*. The employed bees of abandoned food sources become *scout bees*. In ABC, each food source represents a candidate solution and the nectar amount of the associated food sources corresponds to the fitness value of the solution.

The employed bees are randomly initialized in the search space. The employed bees are associated with each food source and find the new food sources by using the search strategy shown in (9). If the quality (fitness) of the new food source is better than that of the old food source, the old food source is replaced by new one. Otherwise, the old food source is retained. In a D-dimensional search space, the position of the i^{th} food source is represented as $X_i = (x_{i1}, x_{i2},\ldots,x_{iD})$. After the information is shared by the employed bees, each of the onlookers selects a food source according to the probability given by:

$$P_i = fit_i \Big/ \sum_{k=1}^{FS} fit_k, \tag{7}$$

where FS is the total number of food sources. The fitness value fit_i is calculated as:

$$fit_i = \frac{1}{1 + f(X_i)}, \tag{8}$$

where $f(X_i)$ is the objective function to be minimized/maximized. The onlooker finds new food source using the following Eq.:

$$x_{new,j} = x_{i,j} + r.\ (x_{i,j} - x_{k,j}), \tag{9}$$

where $k \in (1,2, \ldots, FS)$ such that $k \notin i$ and $j \in (1,2,\ldots,D)$ are randomly chosen indices, r is a uniformly distributed random number within $[-1, 1]$. This is like replacing one random component of the i-th solution with an arithmetic recombination of that component and the corresponding component from another solution. The process conceptually resembles the binomial crossover of another very powerful and elegant continuous parameter optimizer, called Differential Evolution (DE) with crossover rate $Cr = 0$. Similar to the employed bee phase, the greedy selection is applied between the old and new food sources. In this way, onlooker bees select good food sources. In both phases, each bee searches for a better food source until a certain number of cycles called *limit* is exceeded. If the fitness value is not improved until *limit*, that particular bee becomes a *scout* bee. A food source is reinitialized randomly in the scout bee phase and the search process continues until the stopping criterion is satisfied. The pseudo code is provided in Algorithm 5.

Algorithm 5. Artificial Bee Colony Algorithm.

1. Initialization (randomly)
2. Move the employed bees onto their food sources and evaluate their nectar amounts.
3. Place the onlookers depending upon the nectar amounts obtained from employed bees.
4. Send the scouts for exploiting new food sources.
5. Memorize the best food sources obtained so far.
6. If a termination criterion is not satisfied, go to step 2; otherwise stop the procedure and display the best food source obtained so far.

Two evaluations of the fitness function are performed in each generation of the ABC algorithm together with the fitness Function Evaluations (FEs) performed by send scout phase (line 4 in Algorithm 5). This slightly complex way of calculating the FEs (in one generation we have 2^*NP+s FEs, where s is the number of scout bees re-initialization and NP is the total population size) triggered some misconceptions by reporting the quality of results in swarm intelligence based research community. Recently Mernik et al. [103] pointed out these facts for the potential users of this algorithm in their survey article.

4.3. Improved ABC Variants

Tsai et al. [177] was one among the first to make improvements in the performance of the ABC algorithm. As the onlooker bees' movements are limited to the selected nectar source and randomly selected source, the authors proposed an interactive ABC (IABC) algorithm. The main idea of this approach is to improve the relationship of the employed and the onlooker bees. The IABC algorithm used Newtonian law of universal gravitation in the onlooker phase and the selection of onlookers was done using the roulette wheel selection. Baykasoglu [18] proposed a new modification to ABC which uses shift neighborhood searches, double neighborhood searches and greedy randomized adaptive search heuristics. The modified ABC was applied to solve the generalized assignment problems and used penalty function approach for handing the constraints.

Banharnaskun et al. [S101] proposed distributed parallel ABC in which the bee colony was decomposed into several subgroups and performed optimization by using the message passing technique. Wenping et al. [S103] and El-Abd [38] suggested cooperative approaches to ABC for solving complex optimization problems. In order to enhance ABC convergence rate in solving the constrained, composite and non-separable functions, Akay and Karaboga [5] provided more control parameters rather than the single parameter *'limit'* in basic ABC. The parameters are as follows: Modification Rate (MR) to determine number of variables to be perturbed, and Acceleration Factor (AF) which guides the uniform random number generation in search equation. However, this made the search strategy of ABC more similar to the binomial crossover of DE and MR became closer, in spirit, to the crossover rate Cr in DE.

Ruhai et al. [95] integrated a communication strategy for parallelized ABC optimization for solving numerical optimization problems. In this approach, artificial agents are split into several independent subpopulations and the agents in different subpopulations communicate with each other using the proposed communication protocols. Parpinelli [126] proposed a sequential version of ABC enhanced with the local search and investigated the parallelization of ABC algorithm using three parallel modes: master-slave, multi-hive with migrations and hybrid hierarchical on the 3 numerical benchmark functions with a high number of variables.

Alam et al.[7] introduced Exponentially Distributed Mutation (EDM) into ABC. In ABC-EDM, the exponential distribution is incorporated to produce mutation steps with varying lengths and to adjust the current step length suitably. The results verified that their method is better than other methods like PSO and ABC. Aderhold et al. [2] studied the influence of the population size on ABC and investigated the role of onlookers (when it is useful). They also proposed two ABC variants for the position update rule. From their experimentations it is clear that use of onlookers depends on hardness of optimization goal and also their new variants outperformed the standard ABC on the test functions significantly.

Xiaojun and Yanjiao [S134] presented a Fast Mutation ABC (FMABC) algorithm. In this algorithm when choosing food sources, the onlooker uses the pheromone and the sensitivity model in Free Search algorithm to that of traditional roulette wheel selection. Raziuddin et al. [S140] introduced Differential-ABC (DABC) for solving the complex multimodal and dynamic optimization problems. In DABC, bees update strategy is enhanced for improving the solution quality. Banharnasakun et al. [15] modified the solution update rule of onlooker bees, in which the best feasible solutions found so far are shared globally among the entire population. They biased the solution direction toward the best-so-far position and also adapted the radius of search for new candidates as the search progresses towards convergence. Kang et al. [65] presented a Rosenbrock Artificial Bee Colony (RABC) algorithm which combines Rosenbrock's rotational direction with ABC. In RABC, the exploration phase is realized by ABC and the exploitation phase is done by rotational direction method. Their numerical results showed that RABC performed better than the peer algorithms in terms of convergence speed, success rate, and solution accuracy.

Mezura-Montes el al. [104] tried to adapt ABC algorithm to solve the constrained optimization problems by modifying selection mechanism, the scout bee operator and the equality and boundary constraints. In [94], Luo et al. developed Convergence Onlookers ABC (COABC) algorithm in which a new search strategy with the best solution from the previous iteration was used in the onlooker stage to improve the exploitation. The roulette wheel selection mechanism was performed based on nectar amounts of each food source. The experimental results proved that COABC outperformed ABC in the solution quality and convergence rate. Karaboga and Gorkemli [S237] proposed quick ABC (qABC) algorithm in which the local search ability of ABC was improved by modeling the foraging behavior more accurately. Another Improved ABC (IABC) algorithm was developed by Li et al. [S238] by modifying the search process with the best-so-far selection, inertial weight and acceleration coefficients which were borrowed from the realm of Particle Swarm Optimization (PSO). Fister et al. [39] was the first to develop memetic version of ABC algorithm by hybridizing ABC with two local search heuristics: the Nelder-Mead algorithm (NMA) for the exploration and the random walk with direction exploitation (RWDE) for exploitation. The stochastic adaptation method was employed to balance the diversification and intensification of the search.

Fitness learning based ABC with proximity stimuli (FlABCps) was developed to enhance ABC's optimizing ability [30]. The fitness learning with weight selection and proximity stimuli helped to keep the balance between the exploration and exploitation search processing ABC. The fitness learning based ABC algorithm was recently used for solving the protein ligand docking problem [178]. Quantum-inspired ABC (QABC) algorithm was developed to enhance the diversity and computing capability of original ABC [S242]. The quantum concepts such as quantum bit, states superposition and quantum interference were used and tested on the benchmark functions to prove that QABC was competitive to quantum swarm and evolutionary approaches. In order to improve the poor exploitative search of ABC, Powell method was used in [44] as the local search and the new search equation with the best individual from the current population. Powell ABC algorithm was evaluated using the constraint and unconstraint problems and compared to other ABC and recent state-of-the-art algorithms. Its results showed highest quality solution, fastest convergence and strongest robustness. ABC based on orthogonal learning was also presented to address the issue of the poor exploitation of ABC algorithm in [43]. The Orthogonal Experiment Design (OED) was used to develop the Orthogonal Learning (OL) strategy and the search equation was modified like cross operator of GA. The variant ABCs with OL strategy were studied for the search experience and OCABC offered the best in the solution quality and the fast in global convergence rate.

To increase the convergence speed, a combinatorial solution search equation was introduced and to avoid getting stuck in the local optima, chaotic search technique was employed on scout bee phase in the ERABC algorithm. Reverse selection based on roulette wheel was used to preserve the diversity. ERABC algorithm was tested on 23 benchmark functions and it exhibited good performance. The analytical study of various diversity measures in ABC algorithm was also conducted in [S246]. In [8], a two-fold modification of the original ABC algorithm was suggested: the ratio of employed and onlooker bees and number of changes of dimensions' value in each cycle. Various ratios of employed and onlooker bees were examined. The ratio with more number of onlooker bees obtained better results. Two or three changes were made for the dimensions' value in each cycle. The modified ABC algorithm was tested on well-known benchmark functions and two modifications effected on the performance of ABC algorithm in terms of solution quality and convergence rate. In order to find the new search direction, the onlooker bee foraging process was modified by combining the information of the best food sources (based on fitness/nectar value) and the information of the current food source's location in [S248]. The new ABC algorithm was tested using the well-known benchmark functions to show the superior performance and efficiency of the proposed ABC over original ABC algorithm.

Mezura-Montes and Velez-Koeppel [106] developed a novel algorithm called elitist ABC. In elitist ABC, all types of employed, onlooker and scout bees are modified to generate more diverse solutions. Mezura-Montes et al. [105] presented an adaptive variant of ABC based on smart flight and dynamic tolerances. The elitist and adaptive ABCs are applied to solve the constrained optimization problems and verified in [105,106].

To overcome the drawbacks due to single variable perturbation, one-position inheritance scheme [S243] and opposite directional search were introduced to ABC algorithm in [S244]. The performance of the improved ABC algorithm was tested using the numerical benchmark functions. Other works which concentrated majorly in improving the performance of ABC include Efficient and Robust ABC algorithm by Xiang and An [S245], chaotic systems and opposition-base learning method were used in [S235,S236], hybridizing with cat swarm optimization (CSO) by Tsai et al. [S128] for numerical function optimization,hybridizing with simulated annealing (SA) for solving global optimization problem [24] and hybridizing with PSO for solving complex numerical optimization problems [S251].

Rajasekhar et al. [142] presented Cauchy mutation ABC (C-ABC) algorithm. They first validated C-ABC on numerical benchmarks and then considered a real world application of tuning Proportional-Integral speed controller for a permanent magnet synchronous motor (PMSM) drive. Ravi Kumar and Rajasekhar [S125] also developed a new hybrid of ABC and GA based on pipelining method of hybridization and applied for obtaining controller parametric gains in PMSM drive. Similar to the Cauchy mutation proposed by Rajasekhar, Coelho and Alotto [29,S136] modified original ABC and proposed Gaussian ABC algorithm and applied to Loney's Solenoid Benchmark Problem. The authors showed the suitability of ABC in problems relating to electromagnetic optimization.

In order to solve the multi-parameter optimization, dynamic ABC (D-ABC) algorithm was proposed in which a dynamic 'activity' factor was introduced to increase the convergence speed and improve the solution quality of original ABC algorithm [S239]. The D-ABC was applied to multi-parameters optimization of support vector machine (SVM) and better SVM parameters were resulted. The new ABC algorithm with dynamic population size called ABCDP was also introduced in which the population size was dynamically changed related to each iteration result and the algorithm was applied to tackle the economic and emission dispatch problem in the power system [S241]. *gbest*-guided ABC algorithm was improved by using the fitness value of global best solution while calculating probability values associated with the food sources [S247]. The improved GABC algorithm was tested on IEEE 3 bust test system and applied to economic and emission dispatch problem of the wind–thermal power system.ABC algorithm was also hybridized with DE to solve the optimal reactive power flow problem in the power system [S250].

Sonmez [157] introduced ABC with adaptive penalty function approach (ABC-AP) to find the solution for truss structural optimization problem and validated the performance over five truss examples. Li et al. [85] came up with a new hybrid Pareto-based discrete ABC algorithm for solving the multi-objective flexible job shop scheduling problem. Each food source composes two components: routing and scheduling filled with discrete values. The author also introduced a new crossover operator which allows the bees to learn valuable information from each other.

Vargas Benitez and Lopes [S126] proposed two parallel approaches for ABC: master-slave and hybrid-hierarchical for solving protein structure prediction problem. Haris et al.[53] proposed a computationally efficient meta-heuristic algorithm based on ABC and Tabu Search (TS) concepts. Oliveira and Schirru [34] introduced ABC with Random Keys (ABCRK) for solving a combinatorial problem in Nuclear Engineering.

Omkar et al. [118] presented vector evaluated ABC (VEABC) algorithm to solve multi-objective optimization problem (optimizing laminated composite components). VEABC is a parallel vector evaluated type, swarm intelligence multi-objective variant of ABC. The performance of VEABC is validated using different loading configurations like uniaxial, biaxial and bending loads.

Xiu et al. [S269] presented modified ABC (MABC) algorithm in which one position inheritance scheme was introduced to learn from the previous experience and the best-so-far-searched solution was used as the guidance to increase the convergence speed. The MABC algorithm was tested on two mathematical functions and loudspeaker design problem. Compared to ABC, GA, PSO and DE algorithms, MABC converged faster as well as obtained a better solution in [S269].

Wei-Feng et al. [S270] developed a novel ILABC algorithm which was based on information learning. In ILABC, subpopulations were divided by clustering partition. The subpopulation size was dynamically adjusted according to the last search experience. The two search mechanisms were designed for the communication in each subpopulation and between the subpopulations. ILABC algorithm was evaluated using the CEC 2005 test functions in [S270]. ILABC obtained significantly better performance than compared ABCs and EAs in most of the test problems.

A parallel ABC (P-ABC) with a new search method was introduced in [S271]. In P-ABC, the best solution was generated using the mutation operator to increase the exploration and parallel search structure was used to promote the exploration. P-ABC was tested on peak-to-average power ratio reduction problem. P-ABC obtained good performances with low computational complexity.

Horng proposed ABCOO algorithm for solving stochastic economic lot scheduling problem in [S272]. In ABCOO, ABC algorithm was combined with Ordinal Optimization (OO) theory which relaxed the optimization goal from searching the best solution to seeking the solution with high probability of success. The author verified the superiority of ABCOO by comparing with (μ, λ)-ES, GA, and PSO in [S272].

Mustafa et al. [S276] presented ABC with various search strategies called ABCVSS algorithm. In ABCVSS, five search strategies were proposed to search the solution and selected according to the characteristics of the problem to be optimized. The constant counter contents were set for each strategy during initialization and used to determine the strategy during the search process. The ABCVSS was evaluated using the numerical benchmark functions and compared with EAs, DEs and PSOs in [S276].

In [S277], Shayeghi introduced chaos theory into ABC algorithm and proposed chaotic interactive ABC (CIABC) algorithm for solving economic dispatch problem. CIABC uses chaotic local search operator to improve the local search ability. Newton's law of universal gravitation is employed as an interactive law between the employed and onlooker phase to enhance the exploration. Hence, CIABC benefits from the advantages of original ABC, universal law and CLS operators. The efficiency of CIABC algorithm was demonstrated by testing on solving ED of the small and large scale power systems and comparing with other recent ABC algorithms.

Michiharu et al. [S278] presented ABC with reduction (ABCR) to improve the solution search accuracy and to avoid premature convergence. The number of bees is sequentially reduced to the predetermined number based on the fitness function evaluation count. The bees with good fitness value are selected and the rest with the worst evaluation values are eliminated. The effectiveness of ABCR was studied using the two-dimensional six test functions and compared with real-coded GA, DE, PSO and original ABC algorithm.

Ismail et al. [S279] proposed ABC with distributed-based update rule called *dist*ABC algorithm. Instead of differential solution update rule, the distributed-based solution update rule is used to prevent the stagnation behavior. The *dist*ABC algorithm uses the normal distribution with the mean and standard deviation of selected two food sources to generate the new candidate solution. The author evaluated the distABC algorithm using 18 benchmark problems.The distABC performed better than or at least equally to other compared ABC algorithms in [S279].

Xiangtao et al. [S280] presented a self-adaptive constrained ABC (SACABC) algorithm for solving the constraint optimization problems. As the constraint handling methods, the feasible rule is used in the employed bee phase and multi-objective optimization method is used in the onlooker bee phase. In both phases, a new search mechanism is introduced, which leads the new candidate solutions to search around the random solutions of the previous iterations. The modification rate (MR) [5] is used to enhance the convergence speed of SACABC algorithm. Ivona et al. [S281] also presented crossover-based ABC (CB-ABC) to solve the constraint optimization problem. In CA-ABC, boundary constraint handling method and dynamic tolerance were used for handling the constraints. The solution update rules of employed and onlooker bee phases are modified and crossover operator is used in scout bee phase. The performances of both SACABC and CA-ABC algorithms were evaluated using CEC 2006 benchmark suite for constrained problems and compared with other ABC algorithms. The results showed that both constraint optimization algorithms are better or at least comparable to other compared algorithms. However, CA-ABC algorithm outperformed SACABC on solving the constraint optimization problem.

Jadon proposed accelerating ABC with an adaptive local search (AABCLS) in [S282]. In AABCLS, the original ABC algorithm is modified in two ways. The first modification is the solution update rule which forces the low quality solutions to follow the best-so-far solution and encourage the high quality solutions to follow the best as well as the randomly selected solution. The second is the self-adaptive local search strategy in which the individual's search is self-adaptively adjusted to exploit around the neighborhood of the best solution. The AABCLS algorithm was tested on 30 benchmark problems and obtained superior performance than original ABC and recent ABC variants [S282].

In order to address the poor exploitation problem of original ABC, a Gaussian bare-bones ABC (GABC) algorithm was proposed in [S283]. In GABC, Gaussian bare-bones search equation is designed to generate candidate solutions in the onlooker bee phase. The search equation uses the Gaussian distribution with dynamic mean and variance values and samples the search between the current solution and the best solution of current population. The generalized opposition-based learning strategy is employed to generate new food sources in the scout bee phase. The GABC performance is verified using shifted rotated benchmark problems and compared with other ABCs and EAs algorithms. GABC performed significantly better in terms of solution accuracy and convergence speed [S283].

4.4. Hybrid ABC Variants

Xiaohu et al. [S115] proposed a hybrid of ABC and PSO in which the valuable information are mutually shared between the particle swarm and the bee colony. Kang et al. [66] integrated a popular local search algorithm called Nelder-Mead simplex method into ABC and applied to identify the parameters of concrete dam-foundation systems. Marinakis et al. [S86] presented a new hybrid ABC algorithm with GRASP concept for clustering problems. They also proposed a discrete ABC for feature selection (clustering applications) and validated performance of this new heuristic on various clustering datasets. Zhang et al. [S143] developed a chaotic ABC algorithm based on Rossler attractor to solve the partitive clustering problem.

Duan et al. [37] proposed a hybrid method combining Quantum Evolutionary Algorithm (QEA) and ABC to overcome the limitations of stagnation. In their method, ABC is responsible for enhancing the local search capability as well as randomness of populations, which in turn will help QEA to avoid the local optimum. Liu and Cai [88] developed the ABC Programming (ABCP) algorithm based on the randomized distribution, the bit hyper-mutation (similar to differential mutation employed in DE) and a novel crossover operator. The authors claimed the superiority of ABCP by comparing with original ABC. Guopu and Sam [203] were the first to point out that ABC search moves have an inclination towards exploration than exploitation. To overcome this deficiency, the authors proposed *gbest*-guided ABC in which the information of the best solution in the entire bee population was used to guide the search performed by other members, thereby enhancing the exploitation. This was close in spirit to the particle swarm optimization method.

Pulikanti and Singh [S82] introduced a new hybrid approach which combined ABC with a greedy heuristic and a local search. The proposed approach is applied to solve the quadratic knapsack problem in [S82]. Sundar and Sing [161] also presented a similar hybrid approach combining ABC and the local search to solve the non-unicost set covering problem (SCP). Banharnsakun et al. [S107] extended ABC with Greedy Sub-tour Crossover to enhance the precision and applied it to TSP. Jiao et al. [S144] modified ABC and applied in migration of mobile agent problem and compared with other EA methods. Yeh and Hsieh [199] presented a penalty guided ABC to solve the reliability redundancy allocation problem (RAP).

In [201], a bee recurrent neural network is optimized by the ABC algorithm and then combined with Monte Carlo simulation (MCS) to generate the network reliability prediction model. Hsieh et al. [57] presented an integrated system which combines wavelet transforms and ABC-RNN for stock price forecasting and the performance of their approach is verified by predicting several international stock markets. Wang et al. [S133] proposed ABC with Support Vector Machine (ABC-SVM) model to determine the parameters of SVM and feature selection in building intrusion detection system. Irani and Nasimi [61] proposed a hybrid ABC-back propagation strategy (ABC-BP). In ABC-BP, the local search ability of gradient-based back-propagation (BP) strategy is combined with the global search ability of ABC. ABC-BP is applied to bottom hole pressure prediction in underbalanced drilling a two phase flow through annulus.

4.5. ABC for discrete and binary optimization problems

The discrete variant of the ABC algorithm (DABC) was presented for the first time by Pan et al. [123] for solving lot-streaming flowshop scheduling problem. Tasgetiern et al. [167;S135] proposed modified discrete version of ABC by hybridizing with iterated greedy algorithms to find the smallest total flow time for the permutation flow shop scheduling problem. Afterwards, Several DABC algorithms are applied to permutation flowshop scheduling problem in [165,167]; to flexible job shop scheduling problem in [84,86,87]; to hybrid flowshop scheduling problem in [124,125]; to blocking flowshop scheduling problem in [46,52,144]; to economic lot scheduling problem in [21,22,164,166]; to parallel machine scheduling problem in [79]; and to some traveling salesman problem variants in [75,76]. Recently Lia and Pan [83] proposed a hybrid algorithm called TABC that synergises ABC with Tabu Search (TS) for solving the large-scale hybrid flowshop scheduling problems. Unlike the original ABC algorithm, in TABC, each food source represents a string of job numbers. The authors used a novel decoding method to tackle the limited buffer constraints in the schedules generated. Four neighborhood structures were utilized to balance the exploitation and exploration abilities of the algorithm. A TS-based self-adaptive neighborhood strategy was adopted to impart to the TABC algorithm a learning ability for exploring promising regions of the search space by using the neighboring solutions.

Effective discrete ABC (DABC) algorithm was proposed with job-permutation-based representation for solving the real-world hybrid flowshop scheduling problem from a steelmaking process [125]. In the DABC algorithm, neighboring solution generation operator called multi swap was used for addressing the exploration and exploitation search. Besides, two enhanced strategies were introduced to preserve the diversity: replacing the worst one in population with the new solution in the onlooker phase and producing new food source by performing several insertion moves to abandoned food source in the scout bee phase.

Ozturk [S273] presented a novel binary version of ABC. The improved binary ABC model was based on genetic operators and named as GA-ABC algorithm. In GA-ABC, two-point crossover and swap operators were integrated as a genetic neighborhood generator to improve the global and local search abilities. The GB-ABC was tested on dynamic image clustering and 0–1knapsack binary optimization problems. The results validated that GB-ABC is a suitable optimization algorithm for binary optimization, compared to other binary optimization algorithms.

Kiran proposed an adapted binary version of ABC called ABC_{bin} in [S274]. To solve the binary optimization problems, the solutions are converted into binary values before the objective function specified for the problem is evaluated. ABC_{bin} was tested on 15 test problems and compared with binary versions of PSO and ABC variants. The performance comparison showed that ABC_{bin} is competitive and able to offer better solution quality.

Dongli presented binary ABC (BitABC) algorithm in [S275]. The framework of BitABC algorithm is similar to ABC algorithm.

However, the arithmetic mathematical operator is replaced by the binary bitwise operator in the employed and onlooker bee phases. BitABC was compared with other three binary ABC variants and GA in [S275]. The comparison was carried outing using 13 benchmark problems and BitABC performed better than other binary algorithms in terms of solution accuracy, convergence speed and robustness.

Li [S268] introduced a discrete ABC (DABC) algorithm for solving the molten iron scheduling problem. The author introduced problem specific heuristics and presented population initialization method and neighborhood structures for balancing the exploration and exploitation ability. The algorithm was compared with GA, ABC and PSO algorithms and obtained superior performance in solving hybrid flexible flow shop problem in the molten iron systems.

4.6. Real world applications

While some of the researchers tried to focus on development of convergence and quality of solutions of ABC, many others tried to adapt the basic version to solve challenging real world engineering problems. In Table 3, we include the representative references in which ABC was used to solve various practical problems. However, a few of noteworthy approaches are discussed here to provide a basic insight to the readers. ABC had also been used in a wide variety of applications such as the power systems [S284,S285], image processing [S286–S288], aerospace engineering [S289], signal processing [S290], communication [S291], fuzzy systems [S292] and so on.

4.6.1. Bees Algorithm

Bees Algorithm (BA) was first developed in 2005 by Pham et al. [130] to solve the continuous and combinatorial optimization problems. BA is inspired by the honey bee food foraging behavior. In BA, a neighborhood search is combined with the random search and the bee population is divided into two groups: scouts and recruits. The scout bees are responsible for exploring the search space and the recruit bees are for exploiting the solution found by the scout bees. The algorithmic parameters of BA are the number of scout bees (n),the number of sites selected out of n visited sites (m), the number of best sites out of m selected sites(e), the number of bees recruited for best e sites (nep), the number of bees recruited for the other (m-e) selected sites (nsp) and the initial size of patches (ngh). The site and its neighborhood and stopping criterion are defined during initialization.

In the first step, the BA starts with initializing the n scout bees randomly in the search space. Secondly, the sites visited by the scouts are evaluated and the fitness value corresponds to the quality of the solution. The algorithm proceeds until the termination criterion is satisfied. In the sub-step (a), bees with good quality solutions are chosen as “Selected bees” and the sites visited by them are selected for the neighborhood search. The best $m <= n$ scout bees are selected and the rest are discarded. The selected bees are further divided according to their fitness values into e elite selected bees and the m-e non-elite selected bees. Alternatively, the probability to be selected is determined by the fitness values. In sub-step (b), more bees are recruited to follow the elite selected bees and search in the best site neighborhood. The number of recruit bees assigned to each elite selected bees depends on the quality of the solution, neq recruits for each elite bee and nsp recruits fir each non-elite bee. At the assigned neighborhood, each recruit bee performs a local search using the following heuristic step:

$$x_j^* = (x_j - ngh) + (rand(0,1).\ ngh.2) \tag{10}$$

If the solution found by the recruit bees is improved over the best selected bee's solution, the best will be replaced by improved solution. Otherwise, the previous best will be kept. In the end of substep (c) and substep (d), the scout population is filled up with these m solutions and the rest of bees i.e., n-e bees are initialized with random solutions. In this way algorithm proceeds till it reaches the global solution or meets the predefined criterion. Algorithm 6 summarizes steps involved in BA.

Algorithm 6. Bees Algorithm.
1. Initialize the bees randomly in the search space
2. Evaluate the fitness
3. **While** (! = termination criterion)
   (—————New population is being formed—————)
   a. Select the sites (solutions)
   b. Recruit bees to perform local search on selected solutions and evaluate fitness
   c. Out of the solutions obtained (patch) select the best improvement
   d. Assign remaining bees to search randomly (scouts) and evaluate fitness
4. **End While**

Similar to well established approaches like ABC, BA has also showed its advantages in many computationally complex problems that are highlighted in Table 2.

The early development and improvements of the BA are carried out by Pham and Darwish [128,S164]. The authors described an enhanced version of BA in which the fuzzy greedy selection criterion is used to search local sites. Pham et al. [133] proposed an improved BA to handle the dynamic optimization problem. In the proposed BA, the new search operators are included and the newly formed individuals' survival probabilities are enhanced by the new selection procedure. Packianather et al. [121] presented a new version of the BA which uses pheromone to attract explore the promising regions of the search space and compared to the original BA and showed that new version of BA gave an average improvement of 41% in convergence speed. Chen and Lien et al. [25] proposed a new optimization hybrid swarm algorithm by integrating the BA with the PSP termed as PBA. They also introduced a neighborhood-windows technique to enhance the search efficiency and a self-parameter updating technique to avoid getting trapped into a local optimum in solving the high-dimensional problems. The proposed technique is used to solve facility layout problem (FL) and compared the performance to that of BA and PSO. Tian et al. [S175] developed a chaos quantum honey bee algorithm by combing chaos optimization and quantum computation to tackle the uncertain complex optimization problems such as transmission system expansion planning based on random fuzzy chance-constrained programming.

Xiaojing et al. [S180] proposed to use an adapted BA to identify fuzzy measures from sample data and the adaption is based on needs of fuzzy measure extraction problems. BA was also hybridized with ABC to solve the constraint optimization problem in [176].

Huanzhe et al. [S171] proposed an improved bees (IBA) algorithm for solving large-scale layout optimization problem without constraints, which differs from the original BA in the initialization and implementation of neighborhood search. Dereli and Das [35] presented a hybrid algorithm by hybridizing heuristic filling procedure with BA to solve container loading problems a NP-hard optimization problem. Abdullah and Alzaqebah [S254] proposed hybrid self-adaptive bees algorithm to solve examination timetabling problem, in which BA was hybridized with SA and late acceptance hill climbing algorithm to increase the convergence speed. The disruptive, tournament and rank selection strategies were used to select the sites to preserve population diversity.

Apart from numerical function optimization problems (that includes NP-hard, combinatorial problems etc) BA has also been widely used in real world engineering problems. Fonseca et al.[41] devised the new variant of BA, which has the local search ability and the abilities to generate scout locations and perform the waggle dance. The authors used this approach to solve the protein structure prediction problem in molecular biology. Shermeh and Azimi et al.[153] presented a novel hybrid intelligent system that automatically recognizes a variety of digital signals. In this recognizer they proposed a multilayer perception neural network with resilient back propagation learning algorithm as classifier and optimized the classifier design by BA. Shrme [154] investigated MLP neural network classifier with quick prop (QP) learning algorithm, extended delta-bar-delta (EDBD), super self-adaptive back propagation (SuperSAB) and conjugate gradient (CG). The best features are selected by BA to be fed to the classifier. Pham and Sholedolu [134] hybridized PSO with BA to train Neural Networks for classifying wood boards according different type of defects.

In the area of electronics and communications, Pham et al.[131] used BA in designing a two-dimensional recursive filters, studied the adaption of BA optimization technique in parallel and distributed computing [S169]. Ang et al. [S174] described an application of BA with new operators excerpted from TRIZ methodology. BA is employed to optimize the assemble sequence problem in a printed circuit board (PCB) machine. They also made a case study taking various constraints into consideration and claim that BA enhanced with TRIZ operators is performing well mainly in the PCB assembly using a moving-board-with-time-delay type machine. Dhurandher et al. [S178] presented a peer-to-peer file searching bee (P2PBA) algorithm. In P2PBA, BA is used to provide an efficient peer-to-peer file search in the mobile and ad hoc networks (MANETs).

Ang et al. [S167] presented Pareto-based multi-objective BA to determine the minimum traveling time for a SCARA-type robot arm, considering the trajectory smoothness. The authors compared the results with variant of GA, Nelder-Mead flexible polyhedron search etc and claimed the performance of their proposed approach. Shuo et al. [196] proposed a multi-objective Binary Bees Algorithm (BBA) to solve two-level distribution optimization in the robot industry. The problem includes assigning missions to the robots and allocates the robots to home stations. Bhasaputra et al. [19] proposed another multi-objective Bees Algorithm (MOBA) based on principles on multi-objective optimization. In MOBA, a clustering algorithm is included in order to determine Pareto-optimal Set step size and applied to solve multi-objective optimal power flow problem which is validated on standard IEEE 30-bus system.

Other miscellaneous yet notable approaches include, distributed and asynchronous bees (DAB) grid-based approach, proposed by Go et al. [S179] to optimize the magnetic configuration in order to reduce the neoclassical transport of particles in a nuclear fusion device. Moradi et al. [108] investigated the application of BA in the finite element model updating. BA was applied on a piping system to update several physical parameters of its FE model. Then, the numerical model is tested and the parameters obtained are compared with the experimental ones.

4.6.2. BeeAdhoc

Walker [S181] emulated the honey bees foraging behavior to assist customized internet routing and congestion avoidance. This exploration model constructed by Walker has been used by web search engine that builds the document warehouse for the indexers of web page. Wedde et al. [185,S185] presented a new efficient routing algorithm called *BeeAdHoc*, inspired by the honey bees' foraging basics. In the mobile ad hoc networks, this routing algorithm offers better performance, compared to the existing state-of-art algorithms like DSR, AODV and DSDV. In *BeeAdHoc*, two types of agents are mainly used for routing: scouts and forages. Scouts discover the new routings to the destinations. Foragers transport data packets while evaluating the quality of the routes simultaneously. Mazhar and Farooq [101] made a prime step in developing the secure routing model for bio-inspired MANET routing protocols and proposed a security Beesec model for *BeeAdHoc*.

Artificial immune system (AIS) isa promising potential solution for solving MANET security problem. To challenge MANET security issue, Mazhar and Farooq [101] developed the AIS-based security model to detect the misbehaviors in *BeeAdHoc* and *BeeAIS*. Due to various reasons such as dynamic topology, diverse flooding patterns, contention at the medium access control (MAC) layer, it is difficult to model MANET routing protocols. To overcome this shortcoming, Saleem et al. [149] introduced two key performance metrics for *BeeAdHoc* protocol: routing overhead and route optimality. The metrics provide the unbiased analysis of *BeeAdHoc* protocol and its interesting behaviors.

Mazhar and Farooq [102] developed a dendritic cell based distributed misbehavior detection system called BeeAIS-DC for *BeeAdHoc*, to overcome the drawbacks caused due to AIS. Saleem and Farooq [148] proposed a bee inspired power aware routing protocol called *Beesensor*. The simple bee agent model is used and only little processing and network resources are required. Compared to other ad hoc routing algorithms, *BeeHive* obtains better performance in the fixed networks while *BeeAdHoc* offers similar or better performance but at least energy cost.

4.6.3. Bees Swarm Optimization

To overcome the maximum weighted satisfiability (MAX-W-SAT) problem, Drias et al.[36] introduced a metaheuristic called Bees Swarm Optimization (BSO), based on the foraging behavior of real bees. Precisely, the main framework lies in the bees' strategy, where a colony concentrates its effort of harvest on a small number of areas such as the richest and easiest of access, out of all the potential exploitation areas visited by them. Sadeg and Drias [146] presented a parallel version of the BSO metaheuristic and compared the performances of the sequential and the parallel algorithms in solving instances of the weighted maximum satisfiability problem. Akbari et al. [S186] proposed the so called Powerful BSO which provides different patterns that are used by the bees to adjust their flying trajectories.

In order to handle the stagnation problem, Akbari et al. [6] extended BSO to improve its performance using repulsion factor and penalizing fitness. The time-varying weights (TVS) are introduced to obtain the trade-off between exploration and exploitation. They compared this BSO and extended approach with existing algorithms on different test suites and claim that their method produced excellent results. Later, Drias and Mosteghanemi [S187] designed BSO based web information retrieval to explore the prohibitive number of documents to find the information needed by the user.

Askarzadeh and Rezazadeh [11] proposed artificial bee swarm optimization algorithm (ABSO) to identify the parameters of the solar cell model. Following the honey bees' intelligent collection and nectar processing behaviors, ABSO algorithm identified the parameters of a 57 mm diameter commercial silicon solar cell in [11]. The results demonstrated ABSO algorithm's superiority to other methods.

4.6.4. Virtual Bee Algorithm

Yang [198] proposed Virtual Bee algorithm (VBA) based on honey bee foraging behavior. In VBA, the virtual bees are randomly initialized in the search space. Each bee's position is represented each virtual food source. The nectar amount corresponds to the fitness function value of each position. The virtual bees explore and exploit the search space continuously. In due course of time they interact with other bees and pass the information of the food patches to other bees. The bee that receives information will start foraging with these guide lines for faster foraging. In this way, the solutions are found by the interactions between the bees. In 2010 khan et al. [S188] implemented this VBA in designing supplementary damping controller for thyristor controlled series compensator (TCSC) for improving the stability of power system.

4.6.5. Miscellaneous methods resembling bees foraging in nature

Quijano and Passino [S189,S190,141] developed an algorithm for solving the optimal resource allocation problems. The algorithm is based on honey bee social foraging. The authors proved that if several algorithms (of similar kind based on foraging) compete each other in the same problem domain, their algorithms' strategy is stable evolutionarily and as per Nash equilibrium. Moreover, the authors showed that the allocation strategy is globally optimal in the single/multiple hives. The proposed theory was successfully applied for solving dynamic voltage allocation problem related to interconnected grid of temperature zones.

Baig and Rashid [14,S191] presented a different Honey Bee Foraging algorithm which performs a swarm based collective foraging for improving its fitness (nectar amount) in promising neighborhoods in combination with individual scouting searches. Although their approach seemed to be bit different it shares some similarities with BA proposed by Pham.

Based on self-organization of honey bee colony Lu and Zhou [S192] developed a Bee Collecting Pollen (BCP) algorithm based on self-organization of honey bee colony. They validated their algorithm on Traveling Salesman benchmark. Ko et al. [80] proposed a self-adaptive grid computing protocol called Honeydews which draws its inspiration from adaptive bee foraging behavior in nature and applied it to grid based applications. They also developed a variant of Honey adapt, called HoneySort, for application to grid parallelized sorting settings using the master-worker model.

4.7. Bee dances and communication

So far models developed based on Bees Communication are heavily used in the applications of mobile networks and internets. This section briefs few significant contributions made by researchers in developing efficient optimization models mainly in communication applications.

Information sharing within the social structure of honeybees results from *Bee-to-Bee* communication between distinct hierarchical levels as well as between members of the same distinct levels. Centered on this supposition, Walker [S193] presented *Tocorime Apicu*, an information sharing model, to combine information sharing and processing of the honeybees. The model is adapted to sync with information fluctuations occurred in internet communication. For similar kind of software application, Gordon et al. [S194] presented a distributed algorithm for generating any arbitrary pattern by anonymous homogenous mobile agents on a grid.

Inspired by the honey bees' dance language and foraging behavior, Wedde et al. [S195] presented a fault-tolerant, adaptive and robust routing protocol and termed it as *BeeHive* algorithm, which does not need any global information and works with the local information. In the proposed protocol, network regions called foraging zones are defined and the bee agents travel through these network zones and share information on the network to update the local routing tables (Wedde and Farooq [S196,184]). Wedde et al. [S197] discussed the security threats of routing protocol. The protocol is extended into the security model to protect the beehive from the threats. The extended protocol is termed as *BeeHiveGuard*. Wedde et al.[187] integrated artificial immune system into *BeeHive* algorithm and named as BeeHiveAIS.BeeHiveAIS and *BeeHiveGuard* are compared using the empirical ratification framework.

Wang et al. [S198] proposed a QoSuni-cast routing scheme based on the beehive algorithm. Wedde et al. [186,S199] introduced a completely decentralized multi-agent approach (termed *BeeJamA*) on multiple layers in which the car/truck routing problems are solving using the beehive adaptive algorithms. Navrat [S200]; Navrat and Kovacik [S201] presented a web search approach based on a beehive metaphor. The beehive metaphor includes a dance floor, an auditorium, and a dispatch room and can be used as the model to represent the web search processes. Navrat et al.[112] used the beehive metaphor model to search for the user's predefined web-pages. The authors reported their experimental results in the paper and stated that in the search, the model select the best routes and rejected the bad ones. Olague and Puente [S202] proposed honey bee search algorithm. As in the honey bee communication system, the 3D points are communicated to achieve an improved sparse reconstruction. This could be used in further visual computing tasks.

5. Methods based on Honey Bee swarming

5.1. Decision making and nest-site-selection

Passino [127] was one among the first who laid steps in establishing an abstract mathematical model for honey bee nest-site selection process. As said prior Nest-site selection is a process of social decision making in which the scout bees via swarming locate several potential nest sites, evaluate them, and then select best sites by means of competitive signaling. Based on above intelligence, Passino developed a model and validated that the model possesses the key features of the bee's decision-making process. Detailed information about these works can be obtained in Seeley et al. [S203], Passino [S204], Passino et al. [S205, S206] and Nevai et al. [S207].

A practical application with respect to *Nest-Site-Selection* can be ascribed to work carried by Gutierrez and Huhns [49], the authors came up with a new idea of introducing the nest site making decision behavior of bee in designing fault free software which is a robust fault tolerance technique via multi agents thereby improving large scale, critical and complex system reliability.

5.2. Floral and pheromone laying

In this context, Ashlock and Oftelie [10] studied to understand the nectar-gathering behavior of population of virtual bees in a field of simulated flowers over several hundred generations. They developed a mathematical model based on the Finite State Automata (FSAs) familiarly called as Finite State Machines. Based on this structure an evolutionary algorithm is set up and the observations are made for different parameters.

Purnamadjaja and Russell [137] reported a paper about the current progress of an investigation into the possibility of using pheromone communication between members of a robot swarm. In this work they modeled the interaction between members of a robot swarm which is one form of pheromone communication in social bees and used in robotic systems to rescue disable robots. According to their analogy, pheromone communication can offer the efficient communication between robots. Purnamadjaja and Russell [138] also investigated queen Bee's pheromone which has a number of crucial functions in a bee colony. In a robotic system, one important application is to allow the robots to follow their robot leader. Following the queen bee's pheromones idea, the

Please cite this article as: A. Rajasekhar, et al., Computing with the collective intelligence of honey bees – A survey, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.06.001

robot leader can release different chemicals to bring out different behaviors from other robot members.

6. Methods based on spatial memory and navigation

Apart from all the intelligent foraging behavior Bees' above, inspired by the bees' navigation behavior, Bianco [20] developed a model frame work to perform large-scale navigation and mapping in robotics. Lemmens et al. [S208, S209] presented a non-pheromone-based algorithm which combines bees' recruitment and navigation strategies. Based on their experimentation they claim that their proposed model i.e., non-pheromone-based algorithm performs more effectively when collecting food. Their simulations revealed that the proposed approach requires less computational effort. Finally, concluding that their method is less adaptive to pheromone-based algorithms in foraging. The proposed approach uses a direct and straight way to reach to the destination and it does not consider the obstacles. Thus, the proposed non-pheromone-based algorithm is less adaptive in the obstructed environments. Later, Lemmens et al.[S210] introduced a new hybrid algorithm extended with inhibition pheromones to enhance the adaptability.

Recently Das et al.[31] proposed a new and first of its kind, optimization algorithm by synchronizing the Artificial Bee Colony algorithm and spatial memory and termed it as Spatially Informative Perturbation-based ABC algorithm with saccadic flight. The crux of algorithm lies in the optic flow of geo-spatial information about surrounding location in the foragers.

7. Methods based on division of labor

Nakrani and Tovey [110] made significant contributions in solving server allocation problem for internet hosting centers based on bees' intelligence. The hosting center host services on a *pay-per-use* and then dynamically allocates servers to maximize their income. But due to various constraints optimizing server allocation has been a serious issue and to overcome this problem Nakrani and Tovey [S211] introduced a new honey bee parallel algorithm. They validated the efficacy of *honey bee parallel algorithm* against omniscient, greedy and optimal-static algorithms using synthetic and real request streams. As the server allocation is similar to honey bee forager allocation, the authors proposed a honeybee biomimetic algorithm [111]. To highlight the superiority and adaptive nature of proposed approach they validated it on a benchmark followed by the comparison between conventional algorithms.

Gupta and Koul [S212] presented new swarm intelligence based architecture named SWAN for IP network management to overcome the drawbacks of the traditional network management. SWAN is inspired by the beehive's structure and organization, which represents as an examplefor “Management” and “Division of Labor”. Therefore, in SWAN, the network is managed as collection of cells with many agents' assigned specific roles within the cell.

Due to limited progress in defining efficient teamwork architecture for task execution mechanism by collaborating the agent groups together and coordinating with each other, Sadik et al. [58] developed agent teamwork architecture to improve the performance and task execution efficiency. This architecture has its frame work inspired from honeybee teamwork strategies and they also made a short comparison between the similarities of Honey bee and Agents team work. Sadik et al. [S213] also used this strategy for software agents, which is deployed to work on distributed machines and named it Honey Bee teamwork architecture.

8. Potential future research directions with methods based on honey bees

Although the literature reported rich amount of research and addressed various contributions made by researchers for strengthening bee's based algorithms such as hybridization, adding new heuristics, and adaptive mechanisms, yet there are new research areas and application fields left to explore the capabilities of bees' intelligence. In the following section, we put forth the important research future directions of Bees intelligent algorithms.

1) As all these methods are stochastic based methods and all of them initiates via random initialization the search would be effective if the random distribution is also considered while choosing the initial population. Only few of contemporary researchers explored the influence of initial population on the performance of the algorithm. The performance of the search technique relies on the random number generation in Bees family algorithms (a few foraging methods like BA, ABC, BCO etc.). Hence, using the uniform distribution can improve the search performance of the algorithms. As most of the methods obtain the random numbers from subroutine build-in functions available in the programming languages which are not distributed more uniformly its worthwhile to make use of quasi random numbers or low discrepancy random numbers to obtain good convergence rates starting from the first iteration to the end while maintaining the quality of solutions. Most of common methods that are rampant in the usage are Vander-Corput, Halton, Faure and Sobol sequences for initializing the swarm. These initialization schemes become handy in solving the global optimization problems because of the variation of random numbers generated in every iteration/generation.
2) Most of the EAs and swarm intelligent algorithms (including bees) use normal mutation scheme normally based on Gaussian distributions. The advancements of remaining distributions which gives a supervisory control over the distribution patterns came in to existence recently and distributions based on Cauchy, Levy, Laplace can be incorporated in to the mutation schemes to control the step length of the bees' algorithms mainly in solving numerical optimization problems. Apart from these types of mutation works based on incorporation of adaptive mutation strategies which updates there mechanism in the due course of iterations may lead to better performance in the algorithms.
3) There are also few instances in which the search technique may struck at local optimum or may reach stagnation so there requires some adaptive mechanism to exploit and explore the search space continuously and also to balance the searching (convergence) speed. So, it is possible to think of adding speeding methods like Rechenberg's 1/5th rules and adaptive mechanisms such as include Isotropic, Non-isotropic and correlated self-adaptive mechanism. Covariance matrix adaption strategies can also be incorporated to determine the probability distribution for mutation. These mechanisms seem mostly helpful in solving multi-modal functions and which are of current interest in fields related to optimization.
4) Another potential research may be ascribed to memetic methods and hybridization of local methods in to the search process of bees to make the method more robust and quick. So, far researchers almost hybridized every method with another method to make the performance more effective. This poses constraint on time of convergence and care should be taken while hybridizing with new search methods so that it should not increase the time complexity of the algorithm. A few of methods that gained wide reputation in hybridizing (which act as local search strategies) with others EAs are Hooke Jeeves

method, Nelder-Mead simplex method, annealing strategy, greedy search mechanism etc. and when coming to algorithms simulating swarming methods the search strategies can be enhanced with help of Particle swarm optimization (gbest and pbest updating). Apart from these there is a probability of adding new heuristics based on the problem and few hybridization strategies can be borrowed from the other techniques enjoying the hybridization.

5) From the introduction of EAs many papers are published with new methodologies, improvements, modifications to improve the search method performance and few of them proposed novel bio-inspired strategies but only few of them made contributions in the theoretical understanding of which few of them are GAs, ES, EP, BFO, HS [S217–S221]. However, analyzing the Bees family algorithms is still remained open in the research, especially in the convergence property of the algorithms.
6) Of all the methods discussed above many papers published are of Artificial Bee Colony (ABC) because of its simple exploitation and exploration structure which is balanced throughout the end of search. Many researchers pointed the weakness in the mechanism and presented their ideas for enhancing the performance yet there is still lot of exploration that can be addressed to see true potential of ABC in solving the challenging real-time problems that always remained as hurdles for state-of-art optimizers. Few of them include, in ABC the onlookers choose their food sources based on roulette wheel selection and hence algorithm may take more computations to select the food sources so selection mechanism like rank-based, tournament etc., selections can be embedded. The limit parameter is been changed based on the problems, yet Karaboga and Basturk [70] said that it will be wise to use limit equal to product of number of onlookers and dimensions, it is best suited for continuous optimization problems and fails for discrete and higher dimensional problems hence adaption of limit with help of some adaption strategies can enhance the global searching of algorithm. Apart from these new heuristics, hybridization discussed above can also be incorporated.
7) There is also almost unexplored area in the family of bees algorithms i.e., integrating the learning strategies of which quite interesting are opposition based learning, Rule based learning (fuzzy approach), Ensemble methods, Reinforce based methodology (obtained from machine learning). All these strategies shown their outstanding capabilities in ameliorating performance of various search techniques and hence these techniques can be also handled bees inspired methodologies. Of which ensemble based learning is quite new, and this comes from the field of machine learning which is the concept of combining classifier to improve overall classification performance. Interested readers are suggested to go through this portal for further information www.ntu.edu.sg/home/epnsugan/.
8) Recently scaling, grouping, adaption, micro population etc., related to number of individuals are drawing attention of researchers to have control over the population size [S222,S223]. Many of these methods have been successfully applied to family of DE algorithms and there is immense chance to apply these techniques to bee family algorithms. The main concept of these to change the size of population in the due course of iterations or to initialize less number of population with some additional operators to mainly save the number of Functional Evaluations and hence to reach the solutions in less number of epoch.
9) Constrained optimization problems are the most complex problems which often play vital role in modeling real world problem. Solving these problems will be a hectic task to any stochastic based approach. Therefore, the utility of Bee inspired methods has yet to prove in solving constrained optimization problem so far. Discrete problems are seldom solved using the Bee methods and we claim that these problems are yet to be addressed.
10) Many-objective optimization problems [S224,S225] include more than three objective functions. The conventional MOEAs use the Pareto optimality as a ranking metric and thus may perform poorly on these many-objective functions. Hence, extending BA, BCO, ABC algorithms to solve multi-objective problems remains open as a challenging field of future research.

9. Conclusion

It has been almost two decades since researchers started to develop different algorithms based on Honey Bees' collective intelligence for complex and challenging optimization problems that are not easily solvable by conventional methods such as trial and error, LP and NLP. The literature studies reveal that there has been vast amount of articles which are cantered among the optimization algorithms simulating the collective behavior of the bees. Hence, to make easy for the potential researchers to contribute to the area of bees' based computational intelligent methods we tried to provide brief yet comprehensive details of the algorithms that are extensively used by the researchers. We started with providing need for biologically inspired algorithms and elaborated the discussion on life cycle of honey bees and its major roles that laid foundations for the algorithms discussed. Fig. 1 provides the glimpse of stages the bee carries during her cycle and in subsequent sections we provided discussions on algorithms involved in its each stage. Starting with mating of honey beetill division of labour we have provided a brief overview of more than dozen of algorithms.

To give a succinct and powerful grasp on a particular optimization we gave formal introduction of the most prominent algorithms and then discussed their developments, modifications and applications on challenging practical problems. We only discussed in detail only few algorithms which are developed for large class of problems leaving out the methods which are focused only for particular set of applications or problems. Further we also included a table summarizing the commonly occurring engineering problems and algorithms that are used to solve the problem. Although we tried to incorporate the important contributions of various algorithms yet we haven't provided discussions on many (as our goal is to keep the paper succinct) algorithms and the references to such contributions are available as supplementary material. To present a fair version of survey we not only highlighted the advantages but also the dark side of the algorithms and their potential applications and future directions that are yet to be exploited in section VIII. Further there has not been much focus on theoretical aspects of many of the algorithms and contributions in terms of exploiting stability, convergence and chaos of the swarm remains challenging problems in this community of algorithms.

Though NFL pointed out that finding cure all algorithms is difficult it's worthwhile to choose an optimization algorithm that has advantage on particular problem and research in that direction will enable in developing robust algorithm which can attack problems that are in parallel lines. Considering the various interesting results provided by bees based algorithms we firmly believe that these groups of algorithms have authority over various methods and will continue to serve as state-of-art optimization tools in future.

As a final note, in the field of the nature inspired metaheuristics, new algorithms are being published in countless number of conferences and journals very frequently. Inspirations for

Please cite this article as: A. Rajasekhar, et al., Computing with the collective intelligence of honey bees – A survey, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.06.001

designing an optimizer are coming from diverse sources ranging from human beings to flu virus! But are these numerous algorithms really effective? Will they survive in the long run and in face of the challenges posed by real life problems? Or will their influence remain largely stipulated within the world of paper writing? In a recent article, Sörensen [159] expressed his concern on the current trends in metaheuristic research in the following way – "… it seems that no idea is too far-fetched to serve as inspiration to launch yet another metaheuristic. ….we will argue that this line of research is threatening to lead the area of metaheuristics away from scientific rigor".

We believe careful benchmarking with proper performance measures can filter out the junks and indicate the really powerful natural computing methods for taking up the real challenges. The research on honey bee inspired algorithms, in near future should also be more scientifically structured and rigorously validated. The ideas should be presented in a metaphor-free language and more directly (a trend that is lacking in majority of the papers published on bees-inspired algorithms). Following this line, the researchers in this vibrant area have miles to go before they can sleep!

Appendix A. Supplementary material

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.swevo.2016.06.001.

References

[1] H. A. Abbass, MBO: Marriage in honey bees optimization – a haplometrosis polygynous swarming approach. in: Proceedings of the Congress on Evolutionary Computation, 2001, IEEE, 2001.
[2] A. Aderhold et al., Artificial bee colony optimization: a new selection scheme and its performance, in: Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer, 2010, pp. 283–294.
[3] E. Afarandeh, M. Yaghoobi, and M. Bolouri. Fractal image compression by local search and honey bee mating optimization, in: Computer Sciences and Convergence Information Technology (ICCIT), 5th International Conference on, IEEE, 2010.
[4] B. Akay, D. Karaboga, Artificial bee colony algorithm for large-scale problems and engineering design optimization, J. Intell. Manuf. 23 (4) (2012) 1001–1014.
[5] B. Akay, D. Karaboga, A modified artificial bee colony algorithm for real-parameter optimization, Inf. Sci. 192 (2012) 120–142.
[6] R. Akbari, A. Mohammadi, K. Ziarati, A novel bee swarm optimization algorithm for numerical function optimization, Commun. Nonlinear Sci. Numer. Simul. 15 (10) (2010) 3142–3155.
[7] M.S. Alam, M.W.U. Kabir, and M.M. Islam, Self-adaptation of mutation step size in artificial bee colony algorithm for continuous function optimization, in: Computer and Information Technology (ICCIT), 13th International Conference on, IEEE, 2010.
[8] A. Alizadegan, B. Asady, M. Ahmadpour, Two modified versions of artificial bee colony algorithm, Appl. Math. Comput. 225 (2013) 601–609.
[9] B. Amiri, M. Fathian, Integration of self organizing feature maps and honey bee mating optimization algorithm for market segmentation, J. Theor. Appl. Inf. Technol. 3 (3) (2007) 70–86.
[10] D. Ashlock, and J. Oftelie, Simulation of floral specialization in bees, in: Evolutionary Computation, CEC2004. Congress on, IEEE,2004.
[11] A. Askarzadeh, A. Rezazadeh, Artificial bee swarm optimization algorithm for parameters identification of solar cell models, Appl. Energy 102 (2013) 943–949.
[12] M.F. Azeem, A novel parent selection operator in ga for tuning of scaling factors of fkbc. in Fuzzy Systems, 2006 IEEE International Conference on, IEEE, 2006.
[13] M.F. Azeem, M.F. and A.M. Saad, Modified queen bee evolution based genetic algorithm for tuning of scaling factors of fuzzy knowledge base controller, in: India Annual Conference, Proceedings of the IEEE INDICON, First. 2004.
[14] A. Baig and M. Rashid, Foraging for fitness: a honey bee behavior based algorithm for function optimization, Technical report, NUCES, Pakistan, 2006.
[15] A. Banharnsakun, T. Achalakul, B. Sirinaovakul, The best-so-far selection in artificial bee colony algorithm, Appl. Soft Comput. 11 (2) (2011) 2888–2901.
[16] M. Basu, Bee colony optimization for combined heat and power economic dispatch, Expert Syst. Appl. 38 (11) (2011) 13527–13531.
[17] M. Basu, Hybridization of bee colony optimization and sequential quadratic programming for dynamic economic dispatch, Int. J. Electr. Power Energy Syst. 44 (1) (2013) 591–596.
[18] A. Baykasoglu, L. Ozbakir, P. Tapkan, Artificial bee colony algorithm and its application to generalized assignment problem, Swarm Intell.: Focus Ant. Part. swarm Optim. (2007) 113–144.
[19] P. Bhasaputra, S. Anantasate, and W. Pattaraprakorn, Multiobjective Bees Algorithm for Optimal Power Flow Problem. ECTI Trans on Electrical Eng., Electronics, And Communications, vol. 9(1), 2011, pp. 56–64.
[20] G.M. Bianco, Getting inspired from bees to perform large scale visual precise navigation, in: Intelligent Robots and Systems, 2004.(IROS 2004), Proceedings. 2004 IEEE/RSJ International Conference on, IEEE, 2004.
[21] O. Bulut, M.F. Tasgetiren, An artificial bee colony algorithm for the economic lot scheduling problem, Int. J. Prod. Res. 52 (4) (2014) 1150–1170.
[22] O. Bulut and M.F. Tasgetiren, A discrete artificial bee colony algorithm for the Economic Lot Scheduling problem with returns. in Evolutionary Computation (CEC), IEEE Congress on, IEEE, 2014.
[23] H.S. Chang, Converging marriage in honey-bees optimization and application to stochastic dynamic programming, J. Glob. Optim. 35 (3) (2006) 423–441.
[24] S.-M. Chen, A. Sarosh, Y.-F. Dong, Simulated annealing based artificial bee colony algorithm for global numerical optimization, Appl. Math. Comput. 219 (8) (2012) 3575–3589.
[25] M.-Y. Cheng, L.-C. Lien, A hybrid AI-based particle bee algorithm for facility layout optimization, Eng. Comput. 28 (1) (2012) 57–69.
[26] C.-Y. Chiu, I.-T. Kuo, Applying particle swarm optimization and honey bee mating optimization in developing an intelligent market segmentation system, J. Syst. Sci. Syst. Eng. 19 (2) (2010) 182–191.
[27] C.S. Chong, et al., Using a bee colony algorithm for neighborhood search in job shop scheduling problems, in: Procdeedings of the 21st European Conference on Modeling and Simulation (ECMS 2007), 2007.
[28] C.S. Chong, et al., A bee colony optimization algorithm to job shop scheduling, in: Proceedings of the 38th Conference on Winter Simulation, Winter Simulation Conference, 2006.
[29] L.D.S. Coelho, P. Alotto, Gaussian artificial bee colony algorithm approach applied to Loney's solenoid benchmark problem, Magn. IEEE Trans. 47 (5) (2011) 1326–1329.
[30] S. Das, S. Biswas, S. Kundu, Synergizing fitness learning with proximity-based food source selection in artificial bee colony algorithm for numerical optimization, Appl. Soft Comput. 13 (12) (2013) 4676–4694.
[31] S. Das, et al., A spatially informative optic flow model of bee colony with saccadic flight strategy for global optimization, IEEE Trans. Cybern. 44 (10) (2014) 1884–1897.
[32] T. Davidović, M. Šelmić, and D. Teodorović, Scheduling independent tasks: bee colony optimization approach, in: Control and Automation, MED'09, 17th Mediterranean Conference on, IEEE, 2009.
[33] T. Davidović, et al., Bee colony optimization for scheduling independent tasks to identical processors, J. Heuristics 18 (4) (2012) 549–569.
[34] I.M.S. De Oliveira, R. Schirru, Swarm intelligence of artificial bees applied to in-core fuel management optimization, Ann. Nucl. Energy 38 (5) (2011) 1039–1045.
[35] T. Dereli, G.S. Das, A hybrid'bee (s) algorithm'for solving container loading problems, Appl. Soft Comput. 11 (2) (2011) 2854–2862.
[36] H. Drias, S. Sadeg, S. Yahi, Cooperative bees swarm for solving the maximum weighted satisfiability problem, in: Joan Cabestany, Alberto Prieto, Francisco Sandoval (Eds.), Computational Intelligence and Bioinspired Systems, Springer Berlin Heidelberg, Springer, 2005, pp. 318–325.
[37] H. Duan, Z. Xing, C. Xu, An improved quantum evolutionary algorithm based on artificial bee colony optimization, in: Wen Yu, Edgar N. Sanchez (Eds.), Advances in computational intelligence, Springer Berlin Heidelberg, Springer, 2009, pp. 269–278.
[38] M.A. El-Abd, cooperative approach to the artificial bee colony algorithm, in: Evolutionary Computation (CEC), 2010 IEEE Congress on, IEEE, 2010.
[39] I. Fister, I. Fister Jr, and J.B. Zumer, Memetic artificial bee colony algorithm for large-scale global optimization, in: Evolutionary Computation (CEC), 2012 IEEE Congress on, IEEE, 2012.
[40] I. Fister Jr, I. Fister, J. Brest, A hybrid artificial bee colony algorithm for graph 3-coloring, in: Leszek Rutkowski, Marcin Korytkowski, Rafał. Scherer, Ryszard Tadeusiewicz, Lotfi A. Zadeh, Jacek M. Zurada. (Eds.), Swarm and Evolutionary Computation, Springer Berlin Heidelberg, Springer, 2012, pp. 66–74.
[41] R. Fonseca, M. Paluszewski, P. Winter, Protein structure prediction using bee colony optimization metaheuristic, J. Math. Model. Algorithms 9 (2) (2010) 181–194.
[42] K.Z. Gao, et al., A two-stage artificial bee colony algorithm scheduling flexible job-shop scheduling problem with new job insertion, Expert Syst. Appl. 42 (21) (2015) 7652–7663.
[43] W.-f Gao, S.-y Liu, L.-l Huang, A novel artificial bee colony algorithm based on modified search equation and orthogonal learning, Cybern. IEEE Trans. 43 (3) (2013) 1011–1024.
[44] W.-f Gao, S.-y Liu, L.-l Huang, A novel artificial bee colony algorithm with Powell's method, Appl. Soft Comput. 13 (9) (2013) 3763–3775.
[45] A. Ghasemi, A fuzzified multi objective interactive honey bee mating optimization for environmental/economic power dispatch with valve point effect, Int. J. Electr. Power Energy Syst. 49 (2013) 308–321.
[46] D. Guanlong, X. Zhenhao, G. Xingsheng, A discrete artificial bee colony algorithm for minimizing the total flow time in the blocking flow shop scheduling, Chin. J. Chem. Eng. 20 (6) (2012) 1067–1073.

[47] K. Guney, M. Onay, Bees algorithm for design of dual-beam linear antenna arrays with digital attenuators and digital phase shifters, Int. J. RF Microw. Comput.-Aided Eng. 18 (4) (2008) 337–347.
[48] K. Guney, M. Onay, Bees algorithm for interference suppression of linear antenna arrays by controlling the phase-only and both the amplitude and phase, Expert Syst. Appl. 37 (4) (2010) 3129–3135.
[49] R.L.Z. Gutierrez, M. Huhns, Multiagent-based fault tolerance management for robustness, in: Alfons Schuster. (Ed.), Robust Intelligent Systems, Springer London, Springer, 2008, pp. 23–41.
[50] O.B. Haddad, A. Afshar, M.A. Mariño, Honey-bees mating optimization (HBMO) algorithm: a new heuristic approach for water resources optimization, water Resour. Manag. 20 (5) (2006) 661–680.
[51] O.B. Haddad, A. Afshar, M.A. Mariño, Design-operation of multi-hydropower reservoirs: HBMO approach, Water Resour. Manag. 22 (12) (2008) 1709–1722.
[52] Y.-Y. Han, et al., Effective hybrid discrete artificial bee colony algorithms for the total flowtime minimization in the blocking flowshop problem, Int. J. Adv. Manuf. Technol. 67 (1-4) (2013) 397–414.
[53] P.A. Haris, E. Gopinathan, C. Ali, Artificial bee colony and tabu search enhanced TTCM assisted MMSE multi-user detectors for rank deficient SDMA-OFDM system, Wirel. Personal. Commun. 65 (2) (2012) 425–442.
[54] M.-H. Horng, Multilevel minimum cross entropy threshold selection based on the honey bee mating optimization, Expert Syst. Appl. 37 (6) (2010) 4580–4592.
[55] M.-H. Horng, T.-W. Jiang, Image vector quantization algorithm via honey bee mating optimization, Expert Syst. Appl. 38 (3) (2011) 1382–1392.
[56] M.-H. Horng, R.-J. Liou, J. Wu, Parametric active contour model by using the honey bee mating optimization, Expert Syst. Appl. 37 (10) (2010) 7015–7025.
[57] T.-J. Hsieh, H.-F. Hsiao, W.-C. Yeh, Forecasting stock markets using wavelet transforms and recurrent neural networks: an integrated system based on artificial bee colony algorithm, Appl. soft Comput. 11 (2) (2011) 2510–2525.
[58] S.-J. Huang, et al., Application of enhanced honey-bee mating optimization algorithm to fault section estimation in power systems, Power Deliv. IEEE Trans. 28 (3) (2013) 1944–1951.
[59] Y.-M. Huang, J.-C. Lin, A new bee colony optimization algorithm with idle-time-based filtering scheme for open shop-scheduling problems, Expert Syst. Appl. 38 (5) (2011) 5438–5447.
[60] R.M. Idris, A. Khairuddin, and M. Mustafa. A multi-objective Bees Algorithm for optimum allocation of FACTS devices for restructured power system, in: Proceedings of the TENCON 2009-2009 IEEE Region 10 Conference, IEEE, 2009.
[61] R. Irani, R. Nasimi, Application of artificial bee colony-based neural network in bottom hole pressure prediction in underbalanced drilling, J. Pet. Sci. Eng. 78 (1) (2011) 6–12.
[62] A. Jevtic, et al., Distributed Bees Algorithm for task allocation in swarm of robots, IEEE Syst. J. 6 (2) (2012) 296–304.
[63] D. Jeya Mala, V. Mohan, M. Kamalapriya, Automated software test optimisation framework-an artificial bee colony optimisation-based approach, Softw. IET 4 (5) (2010) 334–348.
[64] S.H. Jung, Queen-bee evolution for genetic algorithms, Electron. Lett. 39 (6) (2003) 575–576.
[65] F. Kang, J. Li, Z. Ma, Rosenbrock artificial bee colony algorithm for accurate global optimization of numerical functions, Inf. Sci. 181 (16) (2011) 3508–3531.
[66] F. Kang, J. Li, Q. Xu, Structural inverse analysis by hybrid simplex artificial bee colony algorithms, Comput. Struct. 87 (13) (2009) 861–870.
[67] Q.-M. Kang, et al., Task allocation for maximizing reliability of distributed computing systems using honeybee mating optimization, J. Syst. Softw. 83 (11) (2010) 2165–2174.
[68] A. Karaarslan, The implementation of bee colony optimization algorithm to Sheppard–Taylor PFC converter, IEEE Trans. Ind. Electron. 60 (9) (2013) 3711–3719.
[69] D. Karaboga, An Idea Based on Honey Bee Swarm for Numerical Optimization, 2005.
[70] D. Karaboga, B. Akay, Artificial bee colony (ABC) algorithm on training artificial neural networks, in: Proceedings of the IEEE 15th Signal Processing and Communications Applications, 2007.
[71] D. Karaboga, B. Akay, A modified artificial bee colony (ABC) algorithm for constrained optimization problems, Appl. Soft Comput. 11 (3) (2011) 3021–3031.
[72] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Glob. Optim. 39 (3) (2007) 459–471.
[73] D. Karaboga, C. Ozturk, A novel clustering approach: Artificial Bee Colony (ABC) algorithm, Appl. soft Comput. 11 (1) (2011) 652–657.
[74] N. Karaboga, A new design method based on artificial bee colony algorithm for digital IIR filters, J. Frankl. Inst. 346 (4) (2009) 328–348.
[75] K. Karabulut and M.F. Tasgetiren, A discrete artificial bee colony algorithm for the traveling salesman problem with time windows, in: Evolutionary Computation (CEC), IEEE Congress on, 2012, IEEE.
[76] Karabulut, K. and M.F. Tasgetiren. A discrete artificial bee colony algorithm for the team orienteering problem with time windows, in: Computational Intelligence In Production And Logistics Systems (CIPLS), 2013 IEEE Workshop on, IEEE, 2013.
[77] A. Karcı, Imitation of bee reproduction as a crossover operator in genetic algorithms, in: PRICAI 2004: Trends in Artificial Intelligence, Springer, 2004, pp. 1015–101.
[78] A. Kavousi, et al., Application of the Bee Algorithm for Selective Harmonic Elimination Strategy in Multilevel Inverters, IEEE Trans. Power Electron. 27 (4) (2012) 1689–1696.
[79] D. Kizilay, et al., A discrete artificial bee colony algorithm for the assignment and parallel machine scheduling problem in DYO paint company, in: Evolutionary Computation (CEC), IEEE Congress, IEEE, 2014.
[80] S.Y. Ko, I. Gupta, Y. Jo, A new class of nature-inspired algorithms for self-adaptive peer-to-peer computing, ACM Trans. Auton. Adapt. Syst. 3 (3) (2008) 11.
[81] M. Koudil, et al., Using artificial bees to solve partitioning and scheduling problems in codesign, Appl. Math. Comput. 186 (2) (2007) 1710–1722.
[82] J.Y. Lee, A.H. Darwish, Multi-objective environmental/economic dispatch using the bees algorithm with weighted sum, in: EKC2008 Proceedings of the EU-Korea Conference on Science and Technology, Springer, 2008.
[83] J.-q Li, Q.-k Pan, Solving the large-scale hybrid flow shop scheduling problem with limited buffers by a hybrid artificial bee colony algorithm, Inf. Sci. 316 (2015) 487–502.
[84] J.-Q. Li, Q.-K. Pan, J. Chen, A hybrid Pareto-based local search algorithm for multi-objective flexible job shop scheduling problems, Int. J. Prod. Res. 50 (4) (2012) 1063–1078.
[85] J.-Q. Li, Q.-K. Pan, K.-Z. Gao, Pareto-based discrete artificial bee colony algorithm for multi-objective flexible job shop scheduling problems, Int. J. Adv. Manuf. Technol. 55 (9-12) (2011) 1159–1169.
[86] J.-Q. Li, Q.-K. Pan, M.F. Tasgetiren, A discrete artificial bee colony algorithm for the multi-objective flexible job-shop scheduling problem with maintenance activities, Appl. Math. Model. 38 (3) (2014) 1111–1132.
[87] J.-q Li, et al., A hybrid artificial bee colony algorithm for flexible job shop scheduling problems, Int. J. Comput. Commun. Control 6 (2) (2011) 286–296.
[88] X. Liu and Z. Cai, Artificial bee colony programming made faster, in: Natural Computation, ICNC'09, Fifth International Conference on, IEEE, 2009.
[89] M. Low, M. Chandramohan, and C. Choo, Application of multi-objective bee colony optimization algorithm to Automated Red Teaming, in: Simulation Conference (WSC), Proceedings of the 2009 Winter, 2009.
[90] X. Lu, and Y. Zhou, A Genetic Algorithm Based on Multi-bee population evolutionary for numerical optimization, in: Intelligent Control and Automation, WCICA, 7th World Congress on, IEEE, 2008.
[91] P. Lucic D. Teodorovic. Bee system: modeling combinatorial optimization transportation engineering problems by swarm intelligence, in: Proceedings of the Preprints of the TRISTAN IV Triennial Symposium on Transportation Analysis. 2001.
[92] P. Lučić, D. Teodorović, Computing with bees: attacking complex transportation engineering problems, Int. J. Artif. Intell. Tools 12 (03) (2003) 375–394.
[93] G.-H. Luo, et al., A parallel Bees Algorithm implementation on GPU, J. Syst. Arch. 60 (3) (2014) 271–279.
[94] J. Luo, Q. Wang, X. Xiao, A modified artificial bee colony algorithm based on converge-onlookers approach for global optimization, Appl. Math. Comput. 219 (20) (2013) 10253–10262.
[95] R. Luo, et al. Parallelized artificial bee colony with ripple-communication strategy, in: Proceedings of the Genetic and Evolutionary Computing (ICGEC), Fourth International Conference on, 2010. IEEE.
[96] M. Ma, et al., SAR image segmentation based on Artificial Bee Colony algorithm, Appl. Soft Comput. 11 (8) (2011) 5205–5214.
[97] V. Manoj, E. Elias, Artificial bee colony algorithm for the design of multiplier-less nonuniform filter bank transmultiplexer, Inf. Sci. 192 (2012) 193–203.
[98] Y. Marinakis, and M. Marinaki. A hybrid honey bees mating optimization algorithm for the probabilistic traveling salesman problem. in: Proceedings of the Evolutionary Computation, 2009. CEC'09. IEEE Congress on. 2009. IEEE.
[99] Y. Marinakis, M. Marinaki, G. Dounias, Honey Bees Mating Optimization algorithm for large scale vehicle routing problems, Nat. Comput. 9 (1) (2010) 5–27.
[100] Y. Marinakis, M. Marinaki, N. Matsatsinis, A hybrid clustering algorithm based on honey bees mating optimization and greedy randomized adaptive search procedure, in: Vittorio Maniezzo, Roberto Battiti, Jean-Paul Watson. (Eds.), Learning and Intelligent Optimization, Springer Berlin Heidelberg, Springer, 2007, pp. 138–152.
[101] N. Mazhar, M. Farooq, BeeAIS: Artificial immune system security for nature inspired, MANET routing protocol, BeeAdHoc, in: Leandro Nunes de Castro, Fernando José Von Zuben, Helder Knidel. (Eds.), Artificial Immune Systems, Springer Berlin Heidelberg, Springer, 2007, pp. 370–381.
[102] N. Mazhar, and M. Farooq, A sense of danger: dendritic cells inspired artificial immune system for manet security, in: Proceedings of the 10th annual conference on Genetic and evolutionary computation, ACM, 2008.
[103] M. Mernik, et al., On clarifying misconceptions when comparing variants of the Artificial Bee Colony Algorithm by offering a new implementation, Inf. Sci. 291 (2015) 115–127.
[104] E. Mezura-Montes, O. Cetina-Domínguez, Empirical analysis of a modified artificial bee colony for constrained numerical optimization, Appl. Math. Comput. 218 (22) (2012) 10943–10973.
[105] E. Mezura-Montes, M. Damián-Araoz, and O. Cetina-Domíngez, Smart flight and dynamic tolerances in the artificial bee colony for constrained optimization, in: Evolutionary Computation (CEC), IEEE Congress on, IEEE, 2010.
[106] E. Mezura-Montes and R.E. Velez-Koeppel, Elitist artificial bee colony for constrained real-parameter optimization, in: Proceedings of the Evolutionary Computation (CEC), IEEE Congress on, IEEE, 2010.
[107] Ming, H., J. Baohui, and L. Xu, An improved bee evolutionary genetic

algorithm. in: Proceedings of the Intelligent Computing and Intelligent Systems (ICIS), IEEE International Conference on, IEEE, 2010.
[108] S. Moradi, L. Fatahi, P. Razi, Finite element model updating using bees algorithm, Struct. Multidiscip. Optim. 42 (2) (2010) 283–291.
[109] P. Moreno, et al., Parameter Extraction Method using Genetic Algorithms for an Improved OTFT Compact Model. in Electron Devices, Spanish Conference on, IEEE, 2007.
[110] S. Nakrani, C. Tovey, On honey bees and dynamic server allocation in internet hosting centers, Adapt. Behav. 12 (3-4) (2004) 223–240.
[111] S. Nakrani, C. Tovey, From honeybees to internet servers: biomimicry for distributed management of internet hosting centers, Bioinspiration Biomim. 2 (4) (2007) S182.
[112] P. Návrat. et al., Exploring social behaviour of honey bees searching on the web, in: Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops, IEEE Computer Society, 2007.
[113] T. Niknam, Application of honey-bee mating optimization on state estimation of a power distribution system including distributed generators, J. Zhejiang Univ. Sci. A 9 (12) (2008) 1753–1764.
[114] T. Niknam, An efficient hybrid evolutionary algorithm based on PSO and HBMO algorithms for multi-objective distribution feeder reconfiguration, Energy Convers. Manag. 50 (8) (2009) 2074–2082.
[115] T. Niknam, A new HBMO algorithm for multiobjective daily Volt/Var control in distribution systems considering distributed generators, Appl. Energy 88 (3) (2011) 778–788.
[116] T. Niknam, H.Z. Meymand, H.D. Mojarrad, An efficient algorithm for multi-objective optimal operation management of distribution network considering fuel cell power plants, Energy 36 (1) (2011) 119–132.
[117] T. Niknam, J. Olamaie, R. Khorshidi, A hybrid algorithm based on HBMO and fuzzy set for multi-objective distribution feeder reconfiguration, World Appl. Sci. J. 4 (2) (2008) 308–315.
[118] S. Omkar, et al., Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures, Appl. Soft Comput. 11 (1) (2011) 489–499.
[119] C. Ozkan, O. Kisi, B. Akay, Neural networks with artificial bee colony algorithm for modeling daily reference evapotranspiration, Irrig. Sci. 29 (6) (2011) 431–441.
[120] C. Ozturk and D. Karaboga, Classification by neural networks and clustering with artificial bee colony (ABC) algorithm, in: Proceedings of the 6th International Symposium on Intelligent and Manufacturing Systems, Features, Strategies and Innovation, 2008.
[121] M. Packianather, M. Landy, and D. Pham, Enhancing the speed of the bees algorithm using pheromone-based recruitment, in: Proceedings of the 7th IEEE International Conference on Industrial Informatics, INDIN, IEEE, 2009.
[122] P.-F. Pai, S.-L. Yang, P.-T. Chang, Forecasting output of integrated circuit industry by support vector regression models with marriage honey-bees optimization algorithms, Expert Syst. Appl. 36 (7) (2009) 10746–10751.
[123] Q.-K. Pan, et al., A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem, Inf. Sci. 181 (12) (2011) 2455–2468.
[124] Q.-K. Pan, et al., A novel discrete artificial bee colony algorithm for the hybrid flowshop scheduling problem with makespan minimisation, Omega 45 (2014) 42–56.
[125] Q.-K. Pan, et al., An effective artificial bee colony algorithm for a real-world hybrid flowshop problem in steelmaking process, Autom. Sci. Eng. IEEE Trans. 10 (2) (2013) 307–322.
[126] R.S. Parpinelli, C.M.V. Benitez, H.S. Lopes, Parallel approaches for the artificial bee colony algorithm, in: Bijaya Ketan Panigrahi, Yuhui Shi, Meng-Hiot Lim. (Eds.), Handbook of Swarm Intelligence, Springer Berlin Heidelberg, Springer, 2011, pp. 329–345.
[127] K.M. Passino, Systems biology of group decision making, in: Proceedings of the 14th Mediterranean Conference on Control and Automation, MED'06, IEEE, 2006.
[128] D. Pham and A.H. Darwish. Fuzzy selection of local search sites in the Bees Algorithm, in: Proceedings of the 4th Virtual International Conference on Intelligent Production Machines and Systems, 2008.
[129] D. Pham and A. Ghanbarzadeh, Multi-objective optimisation using the bees algorithm, in: Proceedings of the IPROMS 2007 Conference, 2007.
[130] D. Pham. et al., The bees algorithm. Technical note. Manufacturing Engineering Centre, Cardiff University, UK, 2005, pp. 1–57.
[131] D. Pham, E. Koç, Design of a two-dimensional recursive filter using the bees algorithm, Int. J. Autom. Comput. 7 (3) (2010) 399–402.
[132] D. Pham, et al., Data clustering using the bees algorithm, in: Proceedings of the 40th CIRP International Manufacturing Systems Seminar, 2007.
[133] D. Pham, et al., Dynamic optimisation of chemical engineering processes using the bees algorithm, in: Proceedings of the 17th IFAC World Congress COEX, 2008.
[134] D. Pham and M. Sholedolu, Using a hybrid PSO-bees algorithm to train neural networks for wood defect classification. in: Proceedings of the 4th International Virtual Conference on Intelligent Production Machines and Systems, IPROMS, 2008.
[135] D.T. Pham, A. Haj Darwish, E.E. Eldukhri, Optimisation of a fuzzy logic controller using the bees algorithm, Int. J. Comput. Aided Eng. Technol. 1 (2) (2009) 250–264.
[136] D.T. Pham, et al., Optimisation of the weights of multi-layered perceptrons using the bees algorithm, in: Proceedings of the 5th International Symposium On Intelligent Manufacturing Systems, Sakarya, Turkey, 2006.
[137] A.H. Purnamadjaja, R.A. Russell, Pheromone communication in a robot swarm: necrophoric bee behaviour and its replication, Robotica 23 (06) (2005) 731–742.
[138] A.H. Purnamadjaja, R.A. Russell, Guiding robots' behaviors using pheromone communication, Auton. Robot. 23 (2) (2007) 113–130.
[139] L.D. Qin, et al, A queen-bee evolution based on genetic algorithm for economic power dispatch, in: Proceedings of the 39th International Universities Power Engineering Conference, UPEC, 2004.
[140] H. Quan and X. Shi. On the analysis of performance of the improved artificial-bee-colony algorithm, in: Proceedings of the Fourth International Conference onNatural Computation, ICNC'08, IEEE, 2008.
[141] N. Quijano, K.M. Passino, Honey bee social foraging algorithms for resource allocation: theory and application, Eng. Appl. Artif. Intell. 23 (6) (2010) 845–861.
[142] A. Rajasekhar, A. Abraham, and R.K. Jatoth. Controller tuning using a cauchy mutated artificial bee colony algorithm, in: Proceedings of the 6th International Conference SOCO on Soft Computing Models in Industrial and Environmental Applications, Springer, 2011.
[143] R.S. Rao, S. Narasimham, M. Ramalingaraju, Optimization of distribution network configuration for loss reduction using artificial bee colony algorithm, Int. J. Electr. Power Energy Syst. Eng. 1 (2) (2008) 116–122.
[144] I. Ribas, R. Companys, X. Tort-Martorell, An efficient Discrete Artificial Bee Colony algorithm for the blocking flow shop problem with total flowtime minimization, Expert Syst. Appl. 42 (15) (2015) 6155–6167.
[145] S.L. Sabat, S.K. Udgata, A. Abraham, Artificial bee colony algorithm for small signal model parameter extraction of MESFET, Eng. Appl. Artif. Intell. 23 (5) (2010) 689–694.
[146] S. Sadeg, H. Drias, A selective approach to parallelise Bees Swarm Optimisation metaheuristic: application to MAX-W-SAT, Int. J. Innov. Comput. Appl. 1 (2) (2007) 146–158.
[147] S. Sadik, et al., Using Honey Bee Teamwork Strategy in Software Agents, in: Proceedings of the 10th International Conference on Computer Supported Cooperative Work in Design, 2006.
[148] M. Saleem, M. Farooq, Beesensor: a bee-inspired power aware routing protocol for wireless sensor networks, in: Mario Giacobini. (Ed.), Applications of Evolutionary Computing, Springer Berlin Heidelberg, Springer, 2007, pp. 81–90.
[149] Saleem, M., S.A. Khayam, and M. Farooq. A formal performance modeling framework for bio-inspired ad hoc routing protocols, in: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, ACM, 2008.
[150] M. Salim, M.-T. Vakil-Baghmisheh, Discrete bee algorithms and their application in multivariable function optimization, Artif. Intell. Rev. 35 (1) (2011) 73–84.
[151] Sato, T. and M. Hagiwara. Bee system: finding solution by a concentrated search, in: Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation, IEEE International Conference on, IEEE, 1997.
[152] F. Sayadi, et al., Multi-objective optimization using the bees Algorithm in time-varying channel for MIMO MC-CDMA systems, Eur. J. Sci. Res. 33 (3) (2009) 411–428.
[153] A.E. Shermeh, H. Azimi, Blind signal-type classification using a novel robust feature subset selection method and neural network classifier, Ann. Telecommun.– Ann. Des. télécommunications 65 (9-10) (2010) 625–633.
[154] A.E. Shrme, Hybrid intelligent technique for automatic communication signals recognition using bees algorithm and MLP neural networks based on the efficient features, Expert Syst. Appl. 38 (5) (2011) 6000–6006.
[155] A. Singh, An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem, Appl. Soft Comput. 9 (2) (2009) 625–631.
[156] M. Soltanjalili, O. Bozorg-Haddad, M.A. Mariño, Effect of breakage level one in design of water distribution networks, Water Resour. Manag. 25 (1) (2011) 311–337.
[157] M. Sonmez, Artificial Bee Colony algorithm for optimization of truss structures, Appl. Soft Comput. 11 (2) (2011) 2406–2418.
[158] M. Sonmez, Discrete optimum design of truss structures using artificial bee colony algorithm, Struct. Multidiscip. Optim. 43 (1) (2011) 85–97.
[159] K. Sörensen, Metaheuristics—the metaphor exposed, Int. Trans. Oper. Res. 22 (1) (2015) 3–18.
[160] S. Sundar, A. Singh, A swarm intelligence approach to the quadratic multiple knapsack problem, in: Kok Wai Wong, B. Sumudu U. Mendis (Eds.), Abdesselam Bouzerdoum. Neural Information Processing. Theory and Algorithms, Springer Berlin Heidelberg, Springer, 2010, pp. 626–633.
[161] S. Sundar, A. Singh, A hybrid heuristic for the set covering problem, Oper. Res. 12 (3) (2012) 345–365.
[162] K. Sundareswaran, V. Sreedevi, Boost converter controller design using queen-bee-assisted GA, IEEE Trans. Ind. Electron. 3 (56) (2009) 778–783.
[163] J. Taheri, et al., A Bee Colony based optimization approach for simultaneous job scheduling and data replication in grid environments, Comput. Oper. Res. 40 (6) (2013) 1564–1578.
[164] M.F. Tasgetiren, O. Bulut, and M.M. Fadiloglu, A discrete artificial bee colony algorithm for the economic lot scheduling problem, in: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), IEEE, 2011.
[165] M.F. Tasgetiren, et al., A discrete artificial bee colony algorithm for the no-idle permutation flowshop scheduling problem with the total tardiness criterion, Appl. Math. Model. 37 (10) (2013) 6758–6779.
[166] M.F. Tasgetiren, et al., A discrete artificial bee colony algorithm for the permutation flow shop scheduling problem with total flowtime criterion, in: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), IEEE, 2010.
[167] M.F. Tasgetiren, et al., A discrete artificial bee colony algorithm for the total flowtime minimization in permutation flow shops, Inf. Sci. 181 (16) (2011)

Please cite this article as: A. Rajasekhar, et al., Computing with the collective intelligence of honey bees – A survey, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.06.001

3459–3475.
[168] J. Teo, and H.A. Abbass, An Annealing Approach to the Mating-Flight Trajectories in the Marriage in Honey Bees Optimization Algorithm, 2001.
[169] D. Teodorović, Swarm intelligence systems for transportation engineering: Principles and applications, Transp. Res. Part C: Emerg. Technol. 16 (6) (2008) 651–667.
[170] D. Teodorović and M. Dell'Orco. Bee Colony Optimization–A Cooperative Learning Approach to Complex Transportation Problems.
[171] D. Teodorović, et al., Bee colony optimization: principles and applications, in: Proceedings of the 8th Seminar on Neural Network Applications in Electrical Engineering, NEUREL, IEEE, 2006.
[172] D. Teodorovic and M. Šelmic, The BCO algorithm for the p-median problem, in: Proceedings of the XXXIV Serbian Operations Research Conference, 2007, pp. 417–420.
[173] V. Tereshko, Reaction-diffusion model of a honeybee colony's foraging behaviour, in: Marc Schoenauer, Kalyanmoy Deb, G.ünther Rudolph, Xin Yao, Evelyne Lutton, Juan Julian Merelo, Hans-Paul Schwefel. (Eds.), Parallel Problem Solving from Nature PPSN VI, Springer Berlin Heidelberg, Springer, 2000.
[174] A. Thammano, P. Poolsamran, SMBO: a self-organizing model of marriage in honey-bee optimization, Expert Syst. Appl. 39 (5) (2012) 5576–5583.
[175] N. Todorovic, S. Petrovic, Bee colony optimization algorithm for nurse rostering, Syst. Man Cybern.: Syst. IEEE Trans. 43 (2) (2013) 467–473.
[176] H.-C. Tsai, Integrating the artificial bee colony and bees algorithm to face constrained optimization problems, Inf. Sci. 258 (2014) 80–93.
[177] P. Tsai, et al., Interactive artificial bee colony (iabc) optimization. ISI2008, 2008.
[178] S. Uehara, K.J. Fujimoto, S. Tanaka, Protein–ligand docking using fitness learning-based artificial bee colony with proximity stimuli, Phys. Chem. Chem. Phys. 17 (25) (2015) 16412–16417.
[179] M.-T. Vakil-Baghmisheh, M. Salim, The design of PID controllers for a Gryphon robot using four evolutionary algorithms: a comparative study, Artif. Intell. Rev. 34 (2) (2010) 121–132.
[180] M. Vakil-Baghmisheh and M. Salim. A modified fast marriage in honey bee optimization algorithm. in: Proceedings of the 5th International Symposium on Telecommunications (IST), IEEE, 2010.
[181] J.S. Wang, C. Liu, and Y. Zhang. Application of improved bee evolutionary genetic algorithm on vehicle routing problem with time window, in: Proceedings of the Control Conference (CCC), 29th Chinese, 2010.
[182] S, Wang, Solving aircraft-sequencing problem based on bee evolutionary genetic algorithm and clustering method, in: Proceedings of the Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, DASC'09, IEEE, 2009.
[183] Y. Wang, W. Chen, C. Tellambura, A PAPR reduction method based on artificial bee colony algorithm for OFDM signals, Wirel. Commun. IEEE Trans. 9 (10) (2010) 2994–2999.
[184] H.F. Wedde M. Farooq. BeeHive: new ideas for developing routing algorithms inspired by honey bee behavior, in: Handbook of Bioinspired Algorithms and Applications, 2005, pp. 321–339.
[185] H.F. Wedde and M. Farooq, The wisdom of the hive applied to mobile ad-hoc networks, in: Proceedings of the Swarm Intelligence Symposium, SIS, IEEE, 2005.
[186] H.F. Wedde., et al., in: Peter Holleczek, Birgit Vogel-Heuser Mobilität und Echtzeit (Eds.), Highly dynamic and adaptive traffic congestion avoidance in real-time inspired by honey bee behavior, Springer Berlin Heidelberg, Springer, 2007, pp. 21–31.
[187] H.F., Wedde, C. Timm, and M. Farooq, BeeHiveAIS: A Simple, Efficient, Scalable and Secure Routing Framework Inspired by Artificial Immune Systems, in: T.P. Runarsson, et al., (Ed.), Parallel Problem Solving from Nature - PPSN IX: 9th International Conference, Reykjavik, Iceland, September 9-13, 2006, Springer Berlin Heidelberg: Berlin, Heidelberg, 2006, pp. 623–632.
[188] L.-P. Wong, M. Low, and C.S. Chong, A bee colony optimization algorithm with the fragmentation state transition rule for traveling salesman problem, in: Proceedings of the 2009 Conference on Innovative Production Machines and Systems (IPROMS), Cardiff University, Cardiff, UK, 2009.
[189] L.-P. Wong, M.Y.H. Low, and C.S. Chong. An efficient bee colony optimization algorithm for traveling salesman problem using frequency-based pruning, in: Proceedings of the 7th IEEE International Conference on Industrial Informatics, INDIN, IEEE, 2009.
[190] L.-P. Wong, M.Y.H. Low, and C.S. Chong. A generic bee colony optimization framework for combinatorial optimization problems, in: Proceedings of the Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation (AMS), IEEE, 2010.
[191] L.-P., Wong, et al. Bee colony optimization algorithm with big valley landscape exploitation for job shop scheduling problems, in: Proceedings of the Simulation Conference, WSC, Winter, IEEE, 2008.
[192] Y. Xiongm, et al., The label-constrained minimum spanning tree problem, in: S. Raghavan, Bruce Golden, Edward Wasil. (Eds.), Telecommunications Modeling, Policy, and Technology, Springer US, Springer, 2008, pp. 39–58.
[193] C. Xu, H. Duan, Artificial bee colony (ABC) optimized edge potential function (EPF) approach to target recognition for low-altitude aircraft, Pattern Recognit. Lett. 31 (13) (2010) 1759–1772.
[194] C. Xu, H. Duan, F. Liu, Chaotic artificial bee colony approach to Uninhabited Combat Air Vehicle (UCAV) path planning, Aerosp. Sci. Technol. 14 (8) (2010) 535–541.
[195] C. Xu, et al. A bee swarm genetic algorithm for the optimization of dna encoding, in: Proceedings of the 3rd International Conference on Innovative Computing Information and Control, ICICIC'08, IEEE, 2008.
[196] S. Xu, et al., Bio-inspired binary bees algorithm for a two-level distribution optimisation problem, J. Bionic Eng. 7 (2) (2010) 161–167.
[197] C. Yang, J. Chen, and X. Tu. Algorithm of fast marriage in honey bees optimization and convergence analysis, in: Automation and Logistics, 2007 IEEE International Conference on, 2007.
[198] X.-S. Yang, Engineering optimizations via nature-inspired virtual bee algorithms, in: José Mira, José R. Álvarez (Eds.), Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach, Springer Berlin Heidelberg, Springer, 2005, pp. 317–323.
[199] W.-C. Yeh, T.-J. Hsieh, Solving reliability redundancy allocation problems using an artificial bee colony algorithm, Comput. Oper. Res. 38 (11) (2011) 1465–1473.
[200] W.-C. Yeh, T.-J. Hsieh, Artificial bee colony algorithm-neural networks for S-system models of biochemical networks approximation, Neural Comput. Appl. 21 (2) (2012) 365–375.
[201] W.-C. Yeh, et al., Approximate reliability function based on wavelet latin hypercube sampling and bee recurrent neural network, Reliab. IEEE Trans. 60 (2) (2011) 404–414.
[202] C. Zhang, D. Ouyang, J. Ning, An artificial bee colony approach for clustering, Expert Syst. Appl. 37 (7) (2010) 4761–4767.
[203] G. Zhu, S. Kwong, Gbest-guided artificial bee colony algorithm for numerical function optimization, Appl. Math. Comput. 217 (7) (2010) 3166–3173.
[204] K. Ziarati, R. Akbari, V. Zeighami, On the performance of bee algorithms for resource-constrained project scheduling problem, Appl. Soft Comput. 11 (4) (2011) 3720–3733.

Please cite this article as: A. Rajasekhar, et al., Computing with the collective intelligence of honey bees – A survey, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.06.001