Applied Soft Computing 106 (2021) 107351

Contents lists available at ScienceDirect

Applied Soft Computing

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

A survey on the Artificial Bee Colony algorithm variants for binary, integer and mixed integer programming problems

Bahriye Akay [a,*], Dervis Karaboga [a,b], Beyza Gorkemli [a], Ebubekir Kaya [c]

[a] Erciyes University, Department of Computer Engineering, Kayseri, Turkey
[b] King Abdulaziz University, Department of Information Systems, Jeddah, Saudi Arabia
[c] Nevsehir Haci Bektas Veli University, Department of Computer Engineering, Nevsehir, Turkey

ARTICLE INFO

Article history:
Received 27 December 2020
Received in revised form 2 March 2021
Accepted 19 March 2021
Available online 26 March 2021

Keywords:
Artificial Bee Colony
Discrete optimization
Binary optimization
Integer programming
Mixed-integer programming

ABSTRACT

Most of the optimization problems encountered in the real world are discrete type which involves decision variables defined in the discrete search space. Binary optimization problems, integer and mixed integer programming problems are of this category, and they require suitable solution representation and search operators to be solved by nature-inspired algorithms. One of the widely-used and well-known nature-inspired algorithms is Artificial Bee Colony (ABC) that has been originally proposed to solve the problems in the continuous domain, and hence, its standard version employs the search operators to exploit the information of the solution vectors encoded in the continuous domain. To be able to cope with the discrete problems, particularly binary, integer and mixed integer programming problems, which are also a group of numeric optimization problems, various encoding types, search operators and selection operators have been integrated into ABC. In this paper, we review the studies proposing new ABC variants to solve discrete numeric optimization problems. To the best of our knowledge, this will be the first comprehensive survey study on this topic. Therefore, we hope that this study would be beneficial to the readers interested in the use of ABC for the binary, integer and mixed integer discrete optimization problems.

1. Introduction

As a subgroup of nature-inspired algorithms, swarm intelligence attempts to develop algorithms which simulate collective and cooperative behavior of creatures living in a swarm or colony form. The collective behavior arises based on micro-level synergy between agents without any macro-level supervision. Each agent organizes itself according to the synergy and communication between agents and environmental conditions. This self-organization is characterized by positive feedback that repeats the effective patterns, negative feedback that abandons the frequent patterns to avoid saturation, fluctuation that brings diversity into swarm, and multiple interactions achieved by the direct and indirect communication among agents. Several swarm intelligence algorithms have been described and introduced into the literature in the last three decades such as Particle Swarm Optimization (PSO) [1], Ant Colony Optimization (ACO) [2], Artificial Bee Colony (ABC) [3], Firefly Algorithm (FA) [4], Bacterial Foraging Algorithm (BFA) [5], Cuckoo Search (CS) [6], Glowworm Swarm Optimization (GSO) [7], Gray Wolf Optimization (GWO) [8], Bees Algorithm (BA) [9], Bat Algorithm (BAT) [10] and Honey Bee Mating Optimization (HBMO) [11]. According to the Google Scholar search (November, 2020), the numbers of studies related with these swarm intelligence algorithms are 389000, 187000, 50400, 26800, 5500, 26200, 44, 1440, 6400, 11200 and 3450 for PSO, ACO, ABC, FA, BFA, CS, GSO, GWO, BA, BAT and HBMO, respectively. The percentage of the studies with respect to the swarm intelligence algorithms considered is shown in Fig. 1. As seen from Fig. 1, among these swarm algorithms, PSO and ACO are in the first and second places, respectively. ABC which was proposed in 2005 is in the third place of the list because it is relatively younger compared to ACO and PSO, which were proposed in 1992 and 1994, respectively. It can be clearly seen that the ABC algorithm is one of the most widely used swarm intelligence algorithms in the studies.

ABC simulates the collective intelligence in a honey bee colony during foraging task that aims to maximize nectar amount loaded to the hive. In the foraging, the forager bees are assigned to specific tasks within a division of labor. The forager bees are mainly classified into three types: employed bees, onlooker bees and scout bees. Each employed bee memorizes the location of her food source discovered and unloads its nectar to the hive

* Corresponding author.
E-mail addresses: bahriye@erciyes.edu.tr (B. Akay), karaboga@erciyes.edu.tr (D. Karaboga), bgorkemli@erciyes.edu.tr (B. Gorkemli), ebubekir@nevsehir.edu.tr (E. Kaya).

https://doi.org/10.1016/j.asoc.2021.107351
1568-4946/

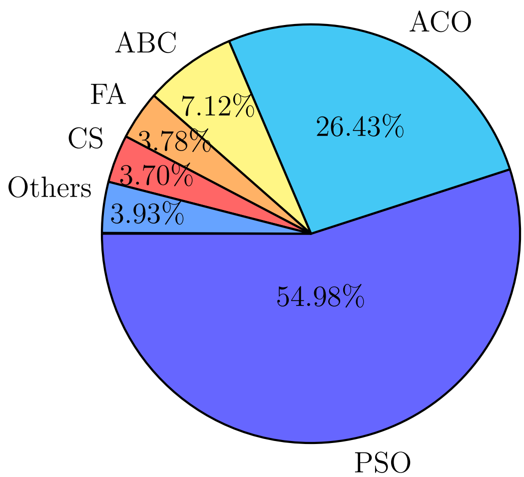

Fig. 1. The percentage of the studies related with the popular swarm intelligence algorithms.

(exploitation). They communicate with other bees (multiple interactions) waiting in the hive, so called onlooker bees. This communication is achieved by dances of the employed bees and recruits the onlooker bees to profitable sources (positive feedback). When the source exploited by an employed bee is exhausted, this source is abandoned (negative feedback) and its employed bee becomes a scout bee to explore a new rich source unvisited before (fluctuation). The ABC algorithm performs exploitation and exploration in the search space by the employed bee, onlooker bee and scout bee phases. The local search and global search abilities of ABC are balanced by positive feedback, negative feedback, fluctuations and multiple interactions properties. Power of the good balance between intensification and diversification has yielded very successful applications of ABC in many research fields [12–18], including unconstrained, constrained, continuous, combinatorial, discrete problems. As many nature-inspired optimization algorithms, the ABC algorithm has been initially proposed for solving continuous optimization problems. To solve the problems defined in discrete space, ABC has been modified to handle with discrete parameters. As known, discrete type optimization problems are classified into several subgroups such as binary, integer and combinatorial. From these, binary, integer and mixed integer groups are also a subgroup of numerical optimization problems and quite different from combinatorial types. In other words, the decision variables of these problems take numeric values such as binary, integer or both of them. Most of the optimization problems arising in many areas fall into this class, and the ABC algorithm has been successfully used to solve these problems, as well [19]. However, to the best of our knowledge, there is no published survey study reviewing the publications related to the use of ABC to solve numerical discrete optimization problems. It is quite clear that a good survey on this topic would be very beneficial to the researchers interested in ABC and its use in these types of problems. Therefore, the purpose of this study is to prepare a comprehensive survey on the variants of ABC, which integrate new representation schemes and search operators into the algorithm. Some constructive heuristics in the initialization phase and new solution production mechanisms have been also incorporated to enhance the convergence rate of the algorithm. Related to these modifications and applications, this paper aims to answer the research questions given below:

- What kind of discrete problems has ABC been applied to?
- Which cost functions and constraints have been involved to guide the search by ABC in discrete domain?
- What type of encoding schemes have been employed to represent solutions in ABC for discrete problems?
- Which operators have been used to evolve solutions represented by the specific encodings?
- How the experiments were validated?
- What are the pros and cons of the approaches?
- What are the future directions related to considered discrete ABC variants?

To be able to answer the questions, we reviewed the studies in the literature in June 2019. Our inclusion criteria is established on keywords (integer, binary, discrete, mixed integer, artificial bee colony), databases indexed (Web of Science, IEEExplore, Springer, Science Direct, Scopus, Google Scholar), publication type (published articles, conference papers, books and book chapters), publication time (starting from 2005 when ABC was proposed and June 2019), the language of the publication (papers written in English and Turkish). We excluded the studies written in the other languages that we could not understand and the studies of which fulltext cannot be accessed. We categorized these studies according to the problem types handled and next we grouped them depending on the type of decision variables in the problems, including binary, integer and mixed integer programming problems. We excluded combinatorial type problems because they use different search characteristics and operators. In the group of binary problems, we considered knapsack, allocation, propositional satisfiability and constraint satisfaction, facility location, feature/attribute space reduction, minimum spanning tree construction, portfolio selection, set covering, unit commitment, bike positioning, satellite technology problems. When there is only one study with a related problem type, it is given in other problems section. In the group of integer programming problems, we reviewed the problems in the communication field, weapon target assignment and some other problems. In the group of mixed integer programming problems, distributions systems, manufacturing production, service composition, software testing, structural optimization and other problem types were investigated.

The rest of the paper is organized as follows. In Section 2, the basic ABC algorithm is presented and described. Sections 3–5 review the studies using ABC algorithm for solving binary, integer and mixed integer problems, respectively. Section 6 presents the summary and discussion about the pros and cons of the approaches and some future directions to fill these gaps. Section 7 is dedicated to the conclusion.

2. Artificial bee colony algorithm

Artificial Bee Colony (ABC) [3] establishes an analogy between the foraging behavior of real honey bee colony and the search for optimal solutions of an optimization problem in the space of solutions which correspond to the food source locations around the hive. As a real honey bee colony has specified bees allocated during foraging, the ABC algorithm has three phases corresponding to these bees: employed bees, onlooker bees and scout bees. The aim of the ABC algorithm is finding the most profitable (fittest) solution in the search space for the problem. The pseudo-code of the basic ABC algorithm is given in Algorithm 1.

ABC starts with assigning values to its control parameters, including the number of solutions (SN), the maximum number of cycles (MNC) or evaluations as termination criterion and the abandonment criteria to leave a food source being exploited, such as a maximum number of exploitations, called *limit*. The first foraging cycle begins with assigning food source positions randomly for the initial scout bees by Eq. (1):

$$x_i^j = x_{\min}^j + \text{rand}[0,1](x_{\max}^j - x_{\min}^j) \quad (1)$$

```
begin
  //Initialization;
  for i = 1 to SN do
    x_i ⟵ a random food source location by Eq. (1);
    f_i = f(x_i);
    c_i = 0;
  end
  while Termination Criteria is not Satisfied do
    //Employed Bees' Phase;
    for i = 1 to SN do
      υ_i ⟵ a neighbor food source location generated by Eq. (2);
      if f(υ_i) < f_i then
        x_i = υ_i ;
        f_i = f(υ_i);
        c_i = 0;
      else
        c_i + +;
      end
    end
    p ⟵ assign probability values by Eq. (3);
    //Onlooker Bees' Phase;
    i = 0 ;
    t = 0 ;
    while t < SN do
      if rand(0, 1) < p_i then
        t = t + 1;
        υ_i ⟵ a neighbor food source location generated by Eq. (2);
        if f(υ_i) < f_i then
          x_i = υ_i ;
          f_i = f(υ_i);
          c_i = 0;
        else
          c_i + +;
        end
      end
      i = (i + 1) mod (SN − 1);
    end
    Memorize the best solution found so far;
    //Scout bee phase;
    si = {i : c_i = max(c⃗)};
    if c_si > limit then
      x_si ⟵ a random food source location by Eq. (1);
      c_si = 0;
    end
  end
end
```

Algorithm 1: Pseudo-code of the basic ABC algorithm

where $i = 1, \dots, SN$, $j = 1, \dots, D$, D is the dimension of the problem (number of optimization parameters), $x^j_{\max}$ and $x^j_{\min}$ are upper and lower bounds of jth parameter, respectively. Each food source is evaluated and assigned a cost function value (f_i) and a counter value (c_i), and the scout bees of more profitable food sources (solutions) start to work as the employed bees which search the vicinity of the sources by Eq. (2) to find better sources to exploit.

$$\upsilon_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \tag{2}$$

where $(k \neq i) \in \{1, 2, \dots, SN\}$ is a randomly chosen neighbor index that is different from i, $j \in \{1, 2, \dots, D\}$ is a random dimension index and $\phi_{ij} \in [-1, 1]$ is a random number generated from uniform distribution. If the new solution, υ_i, is better than the current solution in the employed bee's memory (x_i), it is replaced with the current solution and c_i is set to zero, otherwise the current solution is retained in the population, and c_i is incremented by 1. In the first cycle, onlooker bees who are the initial scouts of less profitable sources (solutions) wait in the hive and choose a profitable food source to fly according to the food source information shared by the employed bees. This is simulated by assigning a probability based on the profitability values (Eq. (3)) to each food source and introducing a stochastic roulette wheel selection by which more bees are likely to be recruited to more profitable sources (solutions) while low quality solutions have also small chance to be selected.

$$p_i = \frac{fitness_i}{\sum_{n=1}^{SN} fitness_n} \tag{3}$$

where p_i is the probability value of the food source i.

After an onlooker bee selects a source, she performs the same search procedure with an employed bee, defined by Eq. (2), and the same greedy selection process is applied to update the counter, c_i. These counters are compared to *limit* parameter value in the scout bee phase to determine the exhausted sources which are exploited sufficiently. If c_i value is higher than the *limit* parameter value, solution x_i is assumed to be exhausted, and a new solution unvisited before is generated by Eq. (1) instead of x_i. These three steps are repeated until termination criterion is satisfied.

3. Artificial bee colony algorithm for binary problems

Binary optimization problems can be defined as a decision problem in which the solutions represent a set of logical variables. There are many binary problems in different research fields, including knapsack, allocation, propositional satisfiability and constraint satisfaction, facility location, feature/attribute space reduction, minimum spanning tree construction, portfolio selection, set covering, unit commitment, bike positioning, satellite technology problems, etc. Some variants of the ABC algorithm have been proposed to solve the binary optimization problems because the original ABC algorithm was proposed for continuous optimization and each solution in this ABC is represented by a real vector. Besides, the solutions produced by basic ABC algorithm may not be feasible or the search operators used in the basic version may not be able to search the binary space efficiently. A group of binary ABC (BABC) algorithms adopts mapping operators which employ a conversion process between continuous and discrete binary spaces. This group generally uses the basic perturbation operations in the standard ABC algorithm. The second group of BABC variants encodes the solutions on binary space and employs a bit operation instead of the local search which results in a real valued vector. Some studies introduce quantum computing and Q-bits to encode solutions.

In this section, we give the details of binary ABC algorithm variants and also brief information about their representation, search operators and repairing mechanisms.

3.1. ABC algorithm for knapsack problem

The 0-1 knapsack problem (KP) is one of the classical NP-hard problems and it in encountered in different areas such as engineering, industry, financial management, etc. The KP is described as finding a subset of n items from the set of all items and resulting in maximum profit while not exceeding the capacity of the knapsack, c with a positive value by Eq. (4).

$$\begin{aligned} &\max\ g(x) = \sum_{i=1}^{n} p_i x_i,\ p_i > 0 \\ &\text{subject to} \sum_{i=1}^{n} w_i x_i \leq c,\ w_i > 0,\ c > 0 \end{aligned} \tag{4}$$

where w_i and p_i correspond to the weight and profit of each item i, respectively and $x_i \in \{0, 1\}$ is binary decision variable that takes value 1 for presence or 0 for absence of the item i in the knapsack. A variant of KP is quadratic KP (QKP) in which profit values are associated with both individual objects and pairs of objects. When both objects of the pair are assigned to the knapsack, the pair profit is also added to the total profit. Another variant of KP is multiple knapsack problem (MKP) is described as assigning the items into m separate knapsacks to maximize the total profit of all selected items without violating the capacity constraint of each knapsack.

Pulikanti and Singh [20] proposed a binary ABC algorithm which uses a bit-vector to represent a solution. The solutions are initialized by a binary tournament strategy. A two-phase heuristic is performed to repair infeasible solutions. In the onlooker bee phase of ABC, binary tournament selection is used instead of roulette wheel selection and different number of employed bees and onlooker bees are used and the number of scout bee in each iteration is not limited. The experiments were validated on a set of 80 quadratic KP benchmark instances. Half of the all instances has 100 objects while other half has 200 objects. In terms of the average solution quality and the minimum time to reach the best solution, the proposed approach outperformed the mini-swarm approach while its performance was comparable to a state-of-the-art hybrid evolutionary algorithm (EA).

Sundar et al. [21] implemented MKP solver based on a binary ABC algorithm in which feasible solutions are generated without violating any of the constraints, initially. A two-phase heuristic is performed to repair infeasible solutions and binary tournament selection is used instead of roulette wheel selection as in [20]. The local search performs 1-1 and 2-1 exchanges in which one or two selected objects are exchanged with an unselected object to ensure an increase in the total profit has been obtained without violating the feasibility. The number of scout bees in each iteration is not restricted. The experiments were validated on the benchmark instances 5.100 and 10.100 from OR-Library. It was concluded that the proposed approach was more efficient and had fast convergence rate compared to other swarm-based approaches.

Sundar and Singh [22] also uses the same initialization, selection procedure and exchange operator in a new binary ABC for solving quadratic KPs. A knapsack replacement procedure is performed to preserve grouping information as far as possible. The perturbation strategy controls each knapsack and iteratively includes as many unselected objects as possible to avoid premature convergence. 60 instances were used in the experiments to test the proposed approach and it was compared to stochastic hill climber, generational GA, steady state grouping GA and steady state GA. The proposed approach improved the best known results on 51 instances and produced better solutions than the other four approaches. However, the computation time of the proposed approach was higher than other approaches except for steady state GA.

Sabet et al. [23] presented a binary variant of ABC in which a hybrid probabilistic mutation scheme is employed for the purpose of local search to improve the exploitation ability and convergence rate. In the initialization phase, instead of random generation, a heuristic-based generation scheme is used. The nectar amount of a solution is proportional to the total profit of the selected items associated with the solution. An elitist selection strategy is also performed to choose the best solutions. A new solution in the neighborhood of current one is constructed by exchanging each item with a probability. Experimental studies were performed on six instances with 25, 50, 100, 300 and 500 items and the results showed that ABC produced better results compared to GA and PSO does especially for the large size problems. Sabet et al. [24] applied the same approach for MKPs using integer representation.

Wei et al. [25] developed a binary ABC algorithm with binary representation and discrete operator for solving KPs. To make solutions feasible so that the positions are within the range [0,1], a normalization procedure and a threshold level are introduced to map all real valued numbers to acceptable range by rounding. To verify the efficiency of the method, four instances with different numbers of items were generated and the results of the proposed approach was compared to binary GA and binary PSO algorithms. It was reported that the proposed method can find the better results robustly and consistently especially on larger scale KP problems.

Liu et al. [26] proposed a binary version of ABC which uses global best solution during search to improve the exploitation ability and the bit mutation operator to increase diversity in the population. The approach generates real valued solutions and applies angle modulation during evaluation to map solutions from the continuous space to the discrete space. In new solution production, a new term is introduced into the local search operator of basic ABC algorithm, which weights the difference between the global best and the current solution. In order to ensure at least one dimension of the new food source is changed, a randomly selected bit of the solution is flipped by mutation operation. The proposed algorithm was compared to angle modulation ABC (AMABC), binary ABC and normalized ABC algorithms on knapsack problems with 10 and 20 items. The simulation results indicated that the proposed approach produces better results compared to the traditional algorithms.

Liu and Hu [27] proposed a quantum encoded binary ABC algorithm to solve dynamic KPs. Employed bee phase of ABC performs dynamic mutation while onlooker bee phase select high quality solutions after a fitness sharing. Unlike traditional EAs adopt single-granularity rotation, a two-step rotation mechanism applies multi-granularity rotation and rotation based on evolution conditions to enhance convergence rate. In order to increase the diversity in the population, a dynamic adaptive mechanism is employed. To assess the performance of the algorithm, dynamic KPs whose scales are ranging from 100 to 3000 were tested. Simulation results demonstrated that the proposed approach performs better than quantum based EA.

Ji et al. [28] developed an inductive pheromone based communication into basic ABC algorithm for solving multi dimensional KPs. They used the encoding, initialization, selection and local search operators proposed by Sundar et al. [21]. A new method for constructing feasible solutions was introduced, and pheromone updating and diffusing strategies were developed to intensify the interaction among bees. It was concluded that the proposed approach finds better solutions in shorter time compared to basic ABC algorithm and some state-of-the-art algorithms.

Manochehri and Alizadegan [29,30] presented a quantum ABC algorithm which exploits q-bits and superposition of states. Experiments were validated on KP with 100, 250, and 500 items. Four variants of quantum based ABC were compared and it was concluded that the variant changing the values of four dimensions and the variant quantizing each of employed and onlooker bees are more suitable for KP.

Ozturk et al. [31] presented a binary ABC algorithm in which initial random solutions are converted to binary values and the genetic operators such as two-point crossover and swap operators are used to modify existing solutions. To assess the performance of the proposed approach, six benchmark datasets with 250, 500 and 750 items were tested in the experiments. The proposed approach was compared to the binary PSO, quantum binary PSO, discrete ABC and GA algorithms and the results demonstrated the effectiveness of the proposed algorithm.

Vasko et al. [32] compared teaching–learning-based optimization, ABC, GA, criss-cross optimization algorithm, and binary bat algorithm on 393 multiple choice MKPs. They concluded that all the metaheuristics produces similar results when a simple neighborhood search is integrated to the algorithms.

Barani and Nezamabadi-pour [33] presented a quantum inspired ABC algorithm to solve binary KPs. The proposed algorithm used the principles of quantum computing such as a quantum bit, superposition of states, and a new rotation gate to enhance the convergence rate and exploration ability. A q-gate determines rotation angle considering the current position and the best position so far. The method was validated on five test cases of the 0/1 knapsack problem with 50, 200, 400, 600, and 1000 items. The proposed approach was compared with those of ten state-of-the-art binary optimization algorithms. It was concluded that the introduced method is a promising tool to solve binary optimization problems.

Cao et al. [34] proposed a binary ABC algorithm with DE to solve KP01. The proposed approach uses binary encoding based on the realization of Bernoulli process. The local search operator in basic ABC is replaced by modulus operation to generate a binary vector. Employed bee searching phase integrates the memory and the neighborhood information. The onlooker bee phase conducts local search by the binary swap mutation and crossover operators as in DE algorithm. Since a solution generated can be infeasible violating the capacity constraint, a greedy optimization algorithm is adopted to repair them. Two datasets were used to evaluate the efficiency of the algorithm. The dimension of instances in the datasets were ranging from 4 to 24. The approach was compared to modified binary PSO, genetic-binary ABC, simplified binary harmony search algorithm, improved monkey algorithm and complex valued encoding wind driven optimization algorithms and it was reported that the proposed approach outperforms other algorithms in fitness value and convergence speed in solving KPs.

Hançer [35] introduced a binary-encoded ABC which searches the neighborhood of the solutions using basic Boolean operators, XOR and AND (ABC-LO). The performance of the proposed method was validated on six benchmark problem sets comprising of a wide range of items from 40 to 750 and capacities from 400 to 20351.5. The results were compared to those of PSO, ABC and GA variants including XOR-ABC, AMABC, discrete ABC, XOR-PSO, binary PSO and GA. The results indicated that the proposed ABC-LO algorithm performs well in knapsack and lot sizing problem sets compared to the others.

He et al. [36] implemented a binary ABC algorithm for solving set-union KPs (SUKP). The solutions are represented by real valued vectors and the operators of the basic ABC algorithm are retained. A surjection mapping function converts real valued parameters to binary values during fitness function evaluation, and infeasible solutions are addressed by a greedy repairing and optimization algorithm. To verify the efficiency of the proposed algorithm, Approximation SUKP (A-SUKP), GA, ABCbin and binDE were compared on three set-union KP instances. The results showed that their approach is more suitable than A-SUKP to solve set-union KPs and outperforms GA, ABCbin and binDE algorithms.

Hasoon [37] formulated the project subset selection problem as KP and solved it by a binary ABC algorithm to find the best investment plan yielding the highest profits under pre-defined costs. Initial solutions are constructed by calculating heuristic information by divide profit to weight. The operator to find a neighbor solution was not included in the study. Binary ABC and GA algorithms were tested on 8, 10, 25, 50, and 100 projects and it was shown that the efficiency of ABC algorithm is better than that of GA.

Zhang and Liu [38] introduced a binary representation and a discrete search equation in ABC algorithm to solve KPs. To enhance the exploitation ability of ABC algorithm, a local search operator which makes a modification on the best so far solution is used in the employed and onlooker bee phases. A new scout operator transposes the solution subtracting by 1. The proposed ABC algorithm was compared to the standard ABC algorithm on 10 KP benchmarks, and the results showed that the proposed approach demonstrates better performance on large-scale KPs compared to the standard ABC algorithm.

The studies are summarized in Table 1.

3.2. ABC algorithm for allocation (assignment) problems

The allocation or assignment problem deals with assigning limited resources to competing activities to minimize the emerging cost or to maximize the profit. Several ABC algorithm variants have been used to solve this kind of problems using various representations and operators. A general assignment problem can be formulated by Eq. (5):

$$
\begin{aligned}
&\min \sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}x_{ij} \\
&\text{subject to} \sum_{i=1}^{n} a_{ij}x_{ij} \le b_j, \\
&\sum_{j=1}^{m} x_{ij} = 1, \\
&x_{ij} \in \{0, 1\}
\end{aligned}
\tag{5}
$$

Baykasoglu et al. [40] presented an ABC algorithm to solve general assignment problem. In the proposed approach, each solution is encoded as a binary vector, and initial solutions are constructed by Greedy Randomized Adaptive Search Heuristic (GRAH). To generate a neighbor solution, shift and double shift operators are applied by each employed bee. In the onlooker bee phase, a neighbor solution is generated by ejection-chain operator. An exhausted source is replaced with a new solution by a scout bee using again GRAH algorithm. To verify the efficiency of the approach, it was tested on a set of problems ranging from 5 to 10 agents and tasks from 15 to 60 tasks from OR library. The proposed algorithm was shown to be effective in solving small to medium sized generalized assignment problems.

Hu et al. [41] hybridized ABC and PSO algorithms to solve warehouse allocation problem considering cube-per-order index (COI) as the criterion of categories. In the approach, the position of the ith particle refers to the COI of each product encoded by real valued numbers. It was concluded that ABC can surpass the disadvantage of getting trapped to local minima.

Bernardino et al. [42] implemented three swarm intelligence algorithms, including hybrid ACO, discrete PSO and ABC algorithms, for terminal assignment problem. The solution representation adopted by the algorithm is integer representation. A partial neighborhood search is applied in the local search performed in the employed and onlooker bee phases. In the local search, a neighbor solution is produced by exchanging two terminals between two random concentrators. 9 problems from literature and 3 large instances generated randomly were used to evaluate the validity of the approaches. They compared SI algorithms against EA algorithms, and it was concluded that EAs show poor performance due loosing the diversity, and SI algorithms are more successful when small size populations are used. Among SI algorithms, the ABC algorithm has the smallest standard deviation while discrete PSO consumes less time.

Banda and Singh [43] solved the terminal assignment problem using ABC algorithm to determine the best links between a set of terminals and a set of concentrators. The proposed approach adopts a terminal based representation in which ith element in the solution vector represents the concentrator number to which the terminal i is assigned. A solution vector is generated by partially greedy and partially random. When a feasible solution cannot be generated even after three attempts, the last infeasible solution penalized with a term is included in the population. To generate a neighbor solution in the vicinity of a solution, each terminal in the solution is reassigned with a probability using a greedy approach. They used the binary tournament selection in the onlooker bee phase of ABC. The number of employee and onlooker bees are different and the number of scouts occurring

Table 1
List of the studies using ABC variants for the Knapsack Problem.

No	Year	Study	Application	Representation
1	2009	Pulikanti and Singh [20]	Quadratic KP	Binary
2	2010	Sundar et al. [21]	MKP	Binary
3	2010	Sundar and Singh [22]	Quadratic MKP	Binary
4	2012	Sabet et al. [23]	KP	Binary
5	2012	Wei et al. [25]	KP	Binary
6	2013	Sabet et al. [24]	MKP	Integer
7	2013	Liu et al. [26]	KP	Real-Valued
8	2013	Liu and Hu [27]	Dynamic KP	Quantum Bit
9	2013	Ji et al. [28]	Multi-dimensional KP	Binary
10	2014	Manochehri and Alizadegan [29]	KP	Quantum Bit
11	2015	Manochehri and Alizadegan [30]	KP	Quantum Bit
12	2015	Ozturk et al. [31]	KP	Binary
13	2016	Vasko et al. [32]	Multiple choice MKP	N/A
14	2018	Barani and Nezamabadi-pour [33]	KP	Quantum Bit
15	2018	Cao et al. [34]	KP	Binary
16	2018	Hançer [35]	KP	Binary
17	2018	He et al. [36]	Set-Union KP	Real-Valued
18	2018	Hasoon [37]	Project Selection as KP	Binary
19	2018	Nouioua et al. [39]	KP	Binary
20	2019	Zhang and Liu [38]	KP	Binary

in one cycle is not limited by one unlike in basic ABC. Experiments were performed on 9 benchmark instances. The results demonstrated that the ABC algorithm is comparable with other methods having the advantage of requiring less execution times for majority of the instances.

Behzadi and Sundarakani [44] applied ABC algorithm to quadratic assignment problem (QAP) including weighted distance and fixed locating cost. The approach was tested on a test problem with 4 facilities and 4 locations, and another test problem with 8 facilities and 8 locations and compared to GA algorithm. It was concluded that ABC algorithm produces better and more stable results while computation time of GA algorithm is better when the problem

Shao et al. [45] presented an improved binary ABC algorithm for zoning protected ecological areas in which site attributes and aggregation attributes are considered. In the proposed approach, a possible solution is represented by a binary array such that the value 1 indicates that the associated zone is selected for protection. Multiple strategies are employed to generate the initial population, and the ecological suitability, urban development potential and spatial compactness are evaluated to measure the quality of a source. To generate a neighbor solution, the replace and alter operations are applied on the current solution and a swap operation is also performed as the local search. To verify the performance of the approach, Landsat −8 image of Sanya, DEM data of ASTER, soil texture data from the Second Nationwide Land Survey, and basic geographic data for the city were used, and a comparison was conducted against other methods (agent-based land allocation model, ant colony optimization, and density slicing). It was reported that their approach is more effective and efficient than the other methods.

Metlicka and Davendra [46] proposed a chaos driven discrete ABC algorithm in which Burgers, Lozi, Delayed Logistic and Tinkerbell are utilized in the random number generators instead of common pseudo-random number generators. Solutions are encoded by discrete permutation representation. To generate a neighbor solution, swapping and inserting operations are applied adaptively based on previous successful modifications. In the employed bee phase, the proposed algorithm applies local search after generating a better solution to further improve it. The approach was tested on quadratic assignment and vehicle routing problems. Based on the experiments, it was concluded that integrating chaos maps in the algorithm improves on the Mersenne twister generator.

Sultan et al. [47] presented a discrete ABC algorithm to solve QAP. In the proposed approach, discrete sequential encoding is used to represent the solutions. In the employed bee phase, some selected cells of a solution are copied to new solution while the partial message crossover is employed in the onlooker bee phase. The approach was validated on 15 benchmarks problems and the QAP in Azadi hospital in Kirkuk and compared to GA and Simulated Annealing (SA). It was reported that the proposed approach produces an improvement of 7.37% in the total cost.

Li et al. [48] proposed an ABC algorithm in which solutions are represented by integer vectors and eight neighborhood structures are utilized to solve location allocation problems in reverse logistic systems. In the initialization, only different solutions are accepted to the population to increase diversity. In employed bee and onlooker bee phases, a random selection is performed to select one of the eight neighborhood structures. In the onlooker bee phase, three solutions are selected and the vicinity of the best one is searched by one of the neighborhood structures. The current solution is replaced with the worst solution in the population. A local search ability was also incorporated to the algorithm to improve the search ability. The performance of the proposed approach was verified on large-scale instances and randomly generated 40 instances with problem scales range from 100 customer zones to 250 customer zones. The experimental results indicated that the proposed approach is more robust and efficient than two algorithms considered in the comparison.

Li et al. [49] presented a quantum ABC algorithm to allocate tasks of the multi-autonomous underwater vehicle. The proposed approach utilizes scrolling time quantum concept to find out the optimal position of the food sources and uses the quantum bit to encode solutions. To verify the proposed approach on the distributed task planning performance of the scroll time, experiments were performed, and the results showed that scrolling time quantum ABC converges faster than the quantum ABC and basic ABC algorithms in terms of convergence rate and efficiency. In another study, Li and Zhang [50] proposed a differential evolution quantum ABC algorithm which integrates the ABC search strategy into the iteration process to avoid local minima and premature convergence.

Wang et al. [51] proposed an improved ABC algorithm for robots tasks assignment. The solutions are encoded in a binary square matrix, and initial solutions are generated by random ranking of unit diagonal matrix. A column state shifting method of matrix is used to ensure feasibility of a solution, and an exchange operator is applied to two solutions to generate neighbor solutions. Two assignment matrices were considered in the experiments, and it was reported that the proposed approach has a fast convergence speed and a strong stability.

Yang et al. [52] utilized ABC algorithm for land-use allocation which is a multi-objective problem in urban-development field. A multi-objective ABC algorithm has been proposed by integrating a new search direction guide and knowledge-informed search mechanism to improve the Pareto front's quality defined based on the suitability and compactness. Two new solutions are generated by exchanging the selected cells of the two parent solutions. They used four land-use types within a region with a size of 36 x 36 raster cells to maximize spatial compactness and suitability. Based on the experiments, the proposed approach demonstrated better convergence ability compared to the other state-of-the-art algorithms.

Yilmaz and Basciftci [53] solved resource allocation problem by a binary ABC algorithm. In the proposed algorithm, each solution is encoded by binary values, and a new solution obtained by the formula in basic ABC algorithm is given to sigmoid function. If the value produced by the sigmoid function is bigger than a random value, the value 1 is assigned for the parameter. All other steps of the basic ABC algorithm are retained in the proposed approach. Performance of the proposed approach was validated on three examples of job-machine allocation problems and compared to that of binary PSO algorithm. The experimental results showed that their approach outperforms against binary PSO algorithm.

Samanta et al. [54] presented a modified ABC algorithm by integrating genetic and neighborhood search algorithms to solve bi-objective quadratic dependent location assignment problem. In the initialization, random permutation is used to generate a population. An archive is integrated to the algorithm to keep the non-dominated solutions. A position based crossover (PBX) is adopted in the employed bee phase while swap or insert operations are applied in the onlooker bee phase to find a new solution. The probability for a solution to be selected is calculated based on non-domination ranking and crowding distance. For fast convergence in the single objective case, opportunistic selection [55] is used instead of greedy selection in the employed bee phase. It was noted that the proposed approach can efficiently find the Pareto Fronts within less computation time.

The studies are summarized in Table 2.

3.3. ABC algorithm for propositional satisfiability and constraint satisfaction problems

The satisfiability problem (SAT) is a decision problem in which we search for whether there exists a truth assignment to the variables evaluating a given propositional formula F to true. Constraint satisfaction problems can be regarded as a generalization of satisfiability problems. A variant of SAT problems is the maximum k-Satisfiability problem (MAX-kSAT) to decide an optimal truth assignment with the maximum value in which each clause has at most k literals.

Aratsu et al. [56] proposed a discrete binary ABC algorithm to solve large-scale and hard constraint satisfaction problems. The subtraction operator in the local search of basic ABC algorithm is replaced with a dissimilarity measure based on the Jaccard coefficient as in [57]. Greedy local search procedure GSAT is performed after the onlooker phase in the proposed approach. To verify the approach, a total of 1400 instances were tested and it was shown that the approach is effective to solve constraint satisfaction problem instances. Aratsu et al. [58,59] used the same approach integrated with greedy scout bees in which new solutions are generated by using a partial assignment of the best solution probabilistically.

Kasihmuddin et al. [60] implemented an ABC algorithm integrated with a Hopfield Network to solve MAX-kSAT problem. The variables in MAX-kSAT are represented by a bit string, and the number of satisfied clauses corresponds to the fitness of a solution. Basic boolean operators (AND, OR, XOR) are used in the local search operator. The proposed model was compared with exhaustive search with Hopfield neural network, and it was concluded that the proposed approach produces better global minima ratio, higher number of satisfied clause, ideal fitness energy landscape value and faster computation time compared to the exhaustive search with Hopfield network. Kasihmuddin et al. [61] compared exhaustive search with Hopfield neural network, and ABC with Hopfield neural network on 2SAT problem and obtains better results using ABC approach.

Guo and Zhang [62] presented a hybrid ABC algorithm based on tabu search (TS) to enhance its local search ability. The proposed approach was compared to the basic ABC algorithm and GA on 150 instances from test library SATLIB. It was reported that their approach has higher success rates and requires less running cost.

Guo and Zhang [63] solved SAT problems using ABC algorithm by discretization of the solutions and introducing Best in Random, and Random in Best neighborhood selection strategies. Experiments were performed on 20 satisfied stochastic instances randomly from SAT2014 international competition to test. The results showed that Best in Random, has better performance than Random in Best and the traditional random method.

The studies are summarized in Table 3.

3.4. ABC algorithm for facility location problems

Facility location problem is an important real-world problem in strategic planning field, which deals with locating facilities optimally such that the transportation costs are minimized with limited or unlimited capacity of each supply. There is a set of customers whose demands must be satisfied by the open facilities or demand points. In capacitated facility location problems, the total demand that each facility may satisfy is limited while in uncapacitated facility location problems, capacity of facilities are not limited. The uncapacitated facility location problem (UFLP) can be defined by Eq. (6):

$$\begin{aligned}
&\min f = \sum_{i=1}^{n}\sum_{j=1}^{m} c_{ij}x_{ij} + \sum_{i=1}^{n} f_i y_i\\
&\text{subject to}\\
&\sum_{j=1}^{n} x_{ij} = 1, \forall i = 1,\dots,m\\
&x_{ij} \le y_i, \forall i = 1,\dots,m, j = 1,\dots,n\\
&x_{ij} \in \{0,1\}, \forall i = 1,\dots,m, j = 1,\dots,n\\
&y_j \in \{0,1\}, \forall j = 1,\dots,n
\end{aligned} \tag{6}$$

where n is the number of facilities, m is the number of demand points, f_i is the cost of locating a facility in i, and c_{ij} is the transportation cost between the ith facility location and the j th demand point. x_{ij} indicates whether customer j is served by the facility at location i, and y_j is 1 if facility is opened at location j.

Kashan et al. [57] designed a binary ABC algorithm which adopts the concept of dissimilarity between binary vectors instead of differential expression in the basic ABC algorithm. In the initialization, each solution is generated as a binary vector using Bernoulli process. To generate a new neighbor solution, a new binary solution generator is proposed. A local search method based on swap moves is performed after the onlooker phase. The experiments were conducted on the set of 15 UFLPs available in OR-Library. The proposed approach was compared with two binary DE and PSO algorithms and shown to be competitive against them.

Tuncbilek et al. [64] presented an ABC algorithm to solve UFLP, in which the solution vectors are converted to binary variables

Table 2
List of the studies using ABC variants for the Allocations Problems.

No	Year	Study	Application	Representation
1	2007	Baykasoglu et al. [40]	General Assignment Problem	Binary
2	2012	Hu et al. [41]	Warehouse Allocation	Real-Valued
3	2013	Bernardino et al. [42]	Terminal Assignment	Integer
4	2014	Banda and Singh [43]	Terminal Assignment	Integer
5	2014	Behzadi and Sundarakani [44]	QAP	N/A
6	2015	Shao et al. [45]	Zone Allocation	Binary
7	2015	Metlicka and Davendra [46]	QAP, Vehicle Routing	Discrete
8	2016	Sultan et al. [47]	QAP	Discrete
9	2017	Li et al. [48]	Location Allocation	Integer
10	2017	Li et al. [49]	Multi-AUV Autonomous Task Allocation	Quantum-Bit
11	2017	Li and Zhang [50]	Multi-AUV Autonomous Task Allocation	Quantum-Bit
12	2017	Wang et al. [51]	Robot Task Assignment	Binary
13	2018	Yang et al. [52]	Land-use Allocation	Binary
14	2018	Yilmaz and Basciftci [53]	Resource Allocation	Binary
15	2018	Samanta et al. [54]	Bi-objective QAP	Discrete
16	2019	Samanta et al. [55]	QAP	Discrete

Table 3
List of the studies using ABC variants for the Satisfiability and Constraint Satisfaction Problems.

No	Year	Study	Application	Representation
1	2012	Aratsu et al. [56]	CSP	Binary
2	2013	Aratsu et al. [58]	CSP	Binary
3	2013	Aratsu et al. [59]	CSP	Binary
4	2016	Kasihmuddin et al. [60]	MAX-kSAT	Binary
5	2017	Kasihmuddin et al. [61]	2SAT	Binary
6	2017	Guo and Zhang [62]	SAT	Binary
7	2017	Guo and Zhang [63]	SAT	Binary

by modulus and floor operations. After onlooker bee phase, flip operator is applied to the global best solution as local search. The proposed method was validated on well-known benchmark problems and also a real world problem on demand and transportation cost data from a fertilizer manufacturer from Turkey. The method was compared to continuous and discrete PSO algorithms, and it was reported that the proposed approach produces better performance compared to the PSO-based algorithms.

Kiran and Gündüz [65] presented a binary ABC algorithm in which solutions are encoded as binary vectors using Bernoulli process, and a new solution is produced based on XOR operator. The proposed approach was compared to binary PSO, the discrete ABC [57] algorithm, and improved binary PSO algorithms. It was demonstrated that their method is a competitive among these algorithms in terms of solution quality, robustness, and simplicity.

Yurtkuran and Emel [66] implemented ABC algorithm with a random key-based coding scheme to solve p-center problem which locates p-centers on a network to minimize the maximum of the distances from each node to its nearest center. The proposed approach employs a new multisearch strategy which uses different search strategies including basic ABC operator, global best guided basic operator, DE/best/1 operator, Powell's method integrating the best and a random solutions. To validate the performance of the algorithm, 40 well-known p-median benchmark problems were used in which the number of customers ranges within [100,900], and the number of centers ranges within [5,200]. The proposed approach was compared with improved bee colony algorithm, multistart interchange, variable neighborhood search, two TS variants, and scatter search. It was demonstrated that the ABC algorithm is more effective and better compared to the other algorithms in terms of both solution quality and CPU time on most of the benchmarks and achieves best-known solutions for 37 out of 40 of the benchmark problems.

Kiran [67] presented an ABC algorithm which applies the steps of the basic algorithm in continuous search space while the solutions are converted to binary vectors during objective function evaluation. To validate the approach, well-known 15 benchmark instances of uncapacitated facility location problem were used and the results of the algorithm were compared to those of PSO, binary PSO, improved binary PSO, binary ABC and discrete ABC algorithms. It was concluded that the proposed ABC variant is an alternative tool considering the solution quality and robustness.

Basti and Sevkli [68] used ABC algorithm to solve the p-median facility location problems which belongs to the class of NP-hard problems. Each variable in solution vector is encoded by real values. However, while evaluating the objective function, these values are converted to binary values by assigning 1 to the smallest p variables and 0 to the others. Local search and selection operators are retained as in the basic ABC algorithm. The proposed approach was validated on OR-Library instances and Galvao problems, and compared to PSO, SA, GA and TS algorithms based on relative percentage of error metric. It was concluded that the approach produces promising results compared to the other metaheuristic methods.

Watanabe et al. [69] developed an ABC algorithm in which initial solutions are generated by randomly assigning 0 or 1. In the employed and onlooker bee phases, a solution is updated by changing flipping a random facility in the solution vector. The approach was tested on 12 datasets of UFLP benchmark problems from OR-Library based on average relative percent error, hit to optimum rate and average computational processing time. Several fitness functions were investigated, and the fitness function employing the best solution is found to be effective while parameter Q should be tuned according to the problem.

The studies are summarized in Table 4.

3.5. ABC algorithm for feature/attribute space reduction

Feature selection is one of the fundamental pre-processing steps, which removes redundant or irrelevant features based on the interactions among the features in the feature space. Assuming that a dataset has N instances and S feature set, selecting a subset $S_n \subset S$ is defined as binary optimization problem in which a binary vector denotes whether an attribute or feature is selected or not. The aim in this optimization problem is finding

Table 4
List of the studies using ABC variants for the Facility Location Problems.

No	Year	Study	Application	Representation
1	2012	Kashan et al. [57]	UFLP	Binary
2	2012	Tuncbilek et al. [64]	UFLP	Binary
3	2013	Kiran and Gündüz [65]	UFLP	Binary
4	2014	Yurtkuran and Emel [66]	p-center problem	Random Keys
5	2015	Kiran [67]	UFLP	Continuous
6	2015	Basti and Sevkli [68]	p-median facility location	Continuous
7	2015	Watanabe et al. [69]	UFLP	Binary

the smallest size subset while retaining the quality of a target measure such as the classification performance or some other goal. Some studies have been presented to solve this problem using ABC algorithm which are briefly described below and listed in Table 5.

Chahkandi et al. [70] presented a chaotic ABC algorithm based on fuzzy membership functions. The population is a collection of binary vectors, and a random number determines the number of features which will be used after feature selection using the logistic map. Each solution is evaluated based on the classification rate using the nearest neighbor algorithm. In this step, three Gaussian membership functions are used to eliminate the ambiguity and to decide the type of bee and the search to be performed. A chaotic method is used to generate new sources and bring diversity in the population. To show the validity of the algorithm, it was tested on wine, diabetes and iris datasets. The results of the proposed approach was compared to those of Relief and Filter feature selection algorithms. The classification accuracy of the proposed approach was shown to be better than those of the other algorithms.

Cai et al. [71] proposed an improved ABC algorithm to solve the binary minimal time cost reduction optimization problem defined by Eq. (7) which aims to reduce the attributes while retaining the classification accuracy of the original problem.

$$T\_C = \max_{1\leq i\leq n}\left(\sum_{j=1}^{i} t_{+}w_i\right) \tag{7}$$

where T_C is the minimal time cost of an attribute subset, n is the number of tests, t_j represents the testing time cost of the jth test, and w_i refers to the waiting cost of the ith test. In the proposed approach, each solution is represented by a binary vector of size M where M is the number of attributes. A new food source production strategy that is suitable for binary problems is proposed based on the global best solution information and inversion. It is ensured that the new solution is better than the current solution, and the search accelerates toward the global optimum. The experiments were validated on the UCI datasets including Zoo, Voting, Tic-tac-toe, and Mushroom. The proposed method was shown to produce better performance in terms of the finding optimal factor, maximal exceeding factor, and average exceeding factor metrics compared to the existing ABC algorithms, especially with the medium-sized Mushroom dataset.

Hancer et al. [72] developed a binary ABC algorithm for feature selection purpose. The proposed algorithm utilizes the idea of evolutionary operators such as mutation, recombination and selection. To adapt evolutionary operators to the binary feature selection problem, the binary dissimilarity is integrated into the neighbor solution production. The proposed algorithm was compared to the binary variants of ABC and PSO algorithms and deterministic non-evolutionary computation methods on 10 benchmark datasets. The results showed that the proposed algorithm can remove the redundant features efficiently and produce higher classification rate compared to the other approaches. The analysis of the CPU times demonstrated that the proposed approach does not consume longer computational time.

Ye and Chen [73] proposed a multidimensional binary local search operator in ABC algorithm for attribute reduction problem. Unlike basic ABC, the proposed approach encodes the solutions as boolean vectors and uses different multi-dimensional local search operators based on rough set dissimilarity in the employed and the onlooker bee phases to enhance diversity in the neighborhood. The so-far-best solution information contributes in local searches and scout search. The approach was tested and compared to TS, GA, PSO, ACO and ABC variants on fourteen UCI datasets. It was reported that the proposed approach has better solution quality compared to other algorithms.

Özger et al. [74] utilized binary ABC for the feature selection problem on 10 UCI datasets. After feature selection step, k-nearest neighbor (k-NN) was used as classifier. 70% of samples randomly drawn was used as training set and the remaining part was used as test set. Results were examined in terms of execution time, number of selected features and train and test dataset errors. It was show that the binary ABC algorithm which uses bitwise operators has good global search ability with less time and satisfying classification performance.

Shunmugapriya and Kanmani [75] combined ACO and ABC algorithms for feature selection purpose. The aim of hybridizing is to avoid the getting stuck in ACO search and the cost of global search in ABC algorithm. First, a set of binary bit strings are generated and assigned to the ants. ACO determines the best solution and ABC performs search on these best solutions. In the scout bee phase, new solutions are produced by the ants. The experiments were performed on thirteen UCI datasets by evaluating the optimal feature subset and maximum prediction accuracy gain. It was reported that the proposed hybrid approach can converge more effectively compared to the individual ACO and ABC algorithms.

Zabidi et al. [76,77] investigated a binary ABC algorithm for feature selection of Nonlinear Autoregressive Moving Average with Exogenous Inputs (NARMAX) model and compared its performance to binary PSO algorithm. In each solution vector, the probabilities of changes are encoded as 1 if they are higher than 0.5 or 0 otherwise. Akaike information criterion, final prediction error and model descriptor length are considered in the fitness evaluations. It was stated that the proposed binary ABC algorithm outperforms the binary PSO algorithm on the problem under consideration.

Hancer et al. [78] treated the feature selection problem as a multi-objective optimization problem which aims to maximize the classification accuracy and to minimize the number of selected features to reduce dimensionality. A multi-objective ABC algorithm using non-dominated sorting procedure and genetic operators was proposed to achieve two objectives. Both binary and continuous representations were investigated on 12 benchmark datasets and the results were compared with those of linear forward selection, greedy stepwise backward selection, two single objective ABC algorithms and three well-known multi-objective evolutionary computation algorithms. The proposed multi-objective ABC algorithm with the binary representation produces better performance compared to the other methods in terms of both classification accuracy and the dimensionality.

Rao et al. [79] proposed a new feature selection method incorporating ABC and gradient boosting decision tree to enhance informative quality of the selected features. The features are selected by ABC algorithm to maximize their contribution in the decision making and passed to the decision tree to perform classification. The ABC algorithm is initialized with binary solutions and employed bees traverse food sources. If an employed bee is chosen by an onlooker bee, then the position of the employed bee in the array is 1, otherwise, the position is 0. Experiments were performed on two breast cancer datasets and six public datasets. It was shown that the ABC integrated with decision tree can successfully reduce the dimensionality and produce superior classification accuracy.

Santana et al. [80] presented a binary ABC algorithm and applied it to feature selection and knapsack problem. In the proposed approach, binary initial solution vectors are generated by Bernoulli process. To generate a new solution in the employed and onlooker bee phases, some random dimensions of a random solution are transferred to the current solution. The proposed method was compared to five binary ABC algorithms and other four well-known methods on the one-max problem, five 0/1 Knapsack problems and eight feature selection datasets. It was demonstrated that the proposed algorithm obtains competitive results and outperforms the other methods.

Zorarpacı and Özel [81] combined ABC and DE algorithms for feature selection before the learning process to avoid overfitting due to dimensionality. Each solution is a binary-coded vector, and a classifier is used to evaluate the selected features of data. Difference and OR operators are employed in the neighborhood production. In the onlooker bee phase, NOT operator is applied to a random dimension of the inverted solution. The proposed approach was validated on fifteen datasets from UCI Repository and compared to pure ABC and DE algorithms and also information gain, chi-square, and correlation feature selection methods. It was reported that the proposed approach improves the classification accuracy.

The studies are summarized in Table 5.

3.6. ABC algorithm for minimum spanning tree construction

A spanning tree $T = (N, E_T, W_T)$ contains all nodes in an undirected graph $G = (N, E, W)$ where N is the set of nodes, E is the set of edges, $W = \sum_{e\in E} w_e$. The minimum spanning tree (MST) problem is finding a spanning acyclic tree having the least total weight among all spanning acyclic trees of G. It is especially useful in ad hoc networks to achieve reliable and efficient routing.

Singh [82] proposed a binary ABC algorithm to solve leaf constrained MST problem in which the minimum spanning tree is restricted to have at least ℓ leaves. In the proposed approach, solutions are encoded using subset encoding and leaf constrained spanning tree is represented by the set of its interior nodes only. In the neighborhood production, the remove and insert operations are applied and some repairing operations follow them to avoid collusions and duplicate solutions. While evaluating the cost of a solution, the solution is transformed to a MST by the set of interior nodes, and each leaf is connected to its nearest interior node. The proposed ABC algorithm was compared to ACO, TS and subset-coded genetic algorithm on 65 problem instances, and from the results, the proposed ABC outperforms the other methods in terms of average solution quality and computational time.

Zhang and Zhang [83] presented a binary ABC algorithm to construct a MST connecting all nodes in vehicular ad hoc networks. Each solution is represented by bit strings where each bit corresponds to an edge in the tree indicating that the edge is covered in the tree if its value is 1. In the initialization, a simple heuristic is performed to make the tree cover $|N| - 1$ 1s in the string. In the employed and onlooker bee phases, a two-element variation technique is applied to keep the feasibility of the solutions. The experiments were validated on a roadside-to-vehicle network represented with a graph G of 16 nodes and 32 edges. It was seen that the proposed approach can find the optimal MST with 92% probability but slower than Kruskal algorithm in terms of computational time.

Li et al. [84] combined quantum computing and ABC algorithm to obtain multiple solutions in one calculation and enhance the reliability of industrial wireless sensor networks. In the proposed approach, $|E|$ bit-length vectors represent solutions where each bit denotes an edge in the graph. The fitness function is the summation of edge weights contained the tree constructed. The employed bees perform a local search based on interchange of bits randomly selected. In the onlooker bees phase, the quantum idea is introduced and a feasibility based selection strategy is applied. Experiments were performed on an industrial wireless sensor network and the results of the proposed algorithm were compared to those of Kruskal and basic binary ABC algorithm. It was reported that the proposed algorithm produces more alternative solutions and has more efficient search ability.

Sundar and Singh [85] proposed an ABC algorithm for quadratic MST (QMST) in which ordered pairs of distinct edges are also considered in the objective function. To represent solutions, edge-set encoding is adopted due to its easiness to be incorporated with metaheuristics. Each solution is initialized by Prim's-like iterative process. Instead of roulette wheel selection, binary tournament selection method is used in the onlooker bee phase. To generate a neighbor solution, an edge of current solution is deleted and added to the tabu list. A least cost candidate edge is selected from a randomly selected neighbor. In the proposed approach, there is no upper limit on the number of scouts in a single iteration. After ABC algorithm is terminated, a local search is applied to further enhance the solution found by the algorithm. The approach was compared to GA on 18 random instances and to some other GA variants on 6 instances. It was concluded that the proposed approach produces better quality solutions compared to these approaches while GA is found to be faster.

Singh and Sundar [86] presented an ABC algorithm for the minimum routing cost spanning tree (MRCST) problem in which the cost is defined as the sum of the costs of the paths connecting all possible pairs of distinct vertices in that spanning tree. The edge-set encoding is used to represent solutions. The solutions are generated in an iterative way similar to Prim's algorithm. The neighbor solution production strategy is derived from the method in [82]. In the approach, different number of employed and onlooker bees are used unlike from the basic ABC. The best routing cost spanning tree generated by the ABC algorithm is further enhanced by a local search operator. The proposed approach was compared to basic ABC, PB-LS, edgeset-coded genetic algorithm, blob-coded GA, stochastic hill climber on instances from OR-library. The proposed approach was stated to be superior over these approaches in terms of solution quality, and it was also seen that except for basic ABC, the approach is faster than other approaches.

Singh and Sundar [87] developed an ABC algorithm for solving the tree t-spanner problem (Tree t-SP) that aims to find a spanning tree whose stretch factor is minimum amongst all spanning trees of the graph. The encoding and initialization scheme employed in [85] are used in the study. The fitness is defined based on the stretch factor. In the onlooker bee phase, binary tournament selection method is applied to select a food source. To generate a new solution, two neighborhood strategies are performed in a mutual exclusive way. Both strategies employ knowledge-directed deletion of an edge of the current solution.

Table 5
List of the studies using ABC variants for the Feature Selection/Attribute Reduction Problems.

No	Year	Study	Application	Representation
1	2013	Chahkandi et al. [70]	Feature Selection	Binary
2	2014	Cai et al. [71]	Attribute Reduction	Binary
3	2015	Hancer et al. [72]	Feature Selection	Binary
4	2015	Ye and Chen [73]	Attribute Reduction	Binary
5	2016	Özger et al. [74]	Feature Selection	Binary
6	2017	Shunmugapriya and Kanmani [75]	Feature Selection	Binary
7	2017	Zabidi et al. [76]	Feature ans Structure Selection	Binary
8	2017	Zabidi et al. [77]	Feature ans Structure Selection	Binary
9	2018	Hancer et al. [78]	Feature Selection	Binary and Continuous
10	2019	Rao et al. [79]	Feature Selection	Binary
11	2019	Santana et al. [80]	Feature Selection, OneMax, KP	Binary
12	2016	Zorarpacı and Özel [81]	Feature Selection	Binary

However, after this step, the first strategy applies insertion of an edge taken from another solution while second neighborhood strategy applies insertion of an edge from its graph. It was concluded that introducing problem-specific knowledge in the neighborhood strategies helps the algorithm to find high quality solutions in lesser computational time, and the proposed approach outperforms GA on a large set of instances.

The studies are summarized in Table 6.

3.7. ABC algorithm for portfolio selection

In finance and economics, spreading the investment among assets instead of investing all in only one is preferred to reduce the investment risks. Choosing optimal allocation of capital which maximizes the return and minimizes the risk simultaneously is called portfolio optimization (PO). An extended version of the problem can be formulated as follows:

$$\begin{aligned}
&\min \lambda \left[\sum_{i=1}^{N}\sum_{j=1}^{N} x_i x_j \sigma_{ij}\right] - (1-\lambda)\left[\sum_{i=1}^{N} x_i \mu i\right] \\
&\text{subject to} \\
&\sum_{i=1}^{N} x_i = 1 \\
&\sum_{i=1}^{N} z_i = K \\
&\varepsilon_i z_i \le x_i \le \delta_i z_i,\ z_i \in \{0,1\},\ i = 1,2,3,\ldots,N
\end{aligned} \tag{8}$$

where σ_{ij} is the objective of the total risk, λ represents an explicit tradeoff between risk and return, x_i and x_j are the weights of assets i and j, respectively. μ_i is the return of the ith asset. The maximum number of assets allowed to be in the portfolio is K. z_i indicates whether the asset i will be selected in the portfolio. ε_i and δ_i are the lower and upper bounds of the assets, respectively. N is the number of all assets. ABC algorithm has been applied to solve portfolio optimization (selection) problems in several studies presented as below.

Hong-mei et al. [88] investigated real estate investment portfolio based on risk preference coefficient considering the return per unit of risk. An improved ABC algorithm was proposed to solve the model. In the proposed approach, the order based discrete representation is used, and the path constructed is treated as a solution to the real estate PO problem. In the study, information about the initialization or local search operator was not provided. The historical data collected between 2000 and 2009 in Nanjing, including rental and price indexes were used to validate the approach. It was indicated that ABC is able to produce better solutions compared to the GA algorithm.

Wang et al. [89] aimed to solve cardinality-constrained PO problem using the ABC algorithm. The proportional variables (continuous) and the decision variables (binary) are represented in a solution vector and initialized in suitable ranges and appropriate representation. Since the problem has constraints to be satisfied, the arrangement algorithm is used to guarantee the feasibility of solutions. The algorithm was applied to the data of five stock market indices: the Hong Kong HangSeng, the German DAX 100, the British FTSE 100, the US S&P 100, and the Japanese Nikkei and compared to GA, TS, SA and PSO. It was concluded that the proposed approach could produce solutions with higher quality compared to the other algorithms. The authors combined this approach with differential evolution strategy to improve the convergence rate, and Wang et al. [90] compared to the same algorithms on the same dataset. The experiment results indicated that the proposed hybrid approach produces efficient frontiers more close to the standard efficient frontiers, and it is better under the same risk values.

Chen et al. [91] implemented an ABC algorithm to balance the trade-off between risk and return in a cardinality-constrained PO problem. The proposed approach employs a hybrid encoding for both discrete (assets' indexes) and continuous variables (percentage of each asset). The approach was validated on four stock market indices-Hang Seng in Hong Kong, DAX 100 in Germany, S&P 100 in USA, and Nikkei 225 in Japan. The results of the ABC algorithm were compared to those of SA, TS and variable neighborhood search (VNS) methods. It was reported that ABC has better diversity, convergence and efficiency on three datasets. Chen et al. [92] improved the ABC algorithm in which two additional constraints are also considered, the cardinality and quantity constraints. The improved algorithm was compared to the same state-of-the-art algorithm in the previous study and on the same problem instances. The improved approach achieves better performance on all four datasets. Based on the investigation of the effect of choosing different number of stocks, fewer number of stocks selected in a portfolio was shown to build a more efficient frontier with lower risk and higher return more quickly.

Ge [93] focused on the semi-variance cardinality constraints PO which is a quadratic 0/1 integer programming model. The model considers the return per unit of risk as an investment decision and has bounds on holdings and cardinality. In the study, an ABC algorithm which combines the extended semi-variance model and the global best solution in the local search was proposed. The method was validated on fifty stocks collected in China selected from January to December during 2013. Based on the experiments, it was stated that the proposed approach produces better solutions in terms of the number of iterations, convergence rate, and quality of solution compared to the standard ABC.

Tuba and Bacanin [94] combined FA and ABC to solve cardinality-constrained mean variance PO problem. FA intensification is incorporated into ABC algorithm to improve convergence rate. To ensure that the solutions are feasible, an arrangement algorithm is performed. The proposed approach was compared

Table 6
List of the studies using ABC variants for the Minimum Spanning Tree Construction.

No	Year	Study	Application	Representation
1	2009	Singh [82]	MST	Discrete Subset Coding
2	2010	Sundar and Singh [85]	QMST	Discrete Edge-Set Encoding
3	2011	Singh and Sundar [86]	MRCST	Discrete Edge-Set Encoding
4	2017	Zhang and Zhang [83]	MST	Binary
5	2018	Singh and Sundar [87]	Tree t-SP	Discrete Edge-set Cncoding
6	2019	Li et al. [84]	MST	Binary+Quantum

to the GA, TS, SA and PSO algorithms, and it was demonstrated that the proposed approach produces better Euclidean distance from the efficiency frontier.

Chen [95] introduced using fuzzy variables for returns of risky assets because they change over time and this change may lead to inaccuracy in the predictions. The author proposed a new possibilistic mean-semiabsolute deviation model under the real-world constraints. The transaction cost is assumed to be a V-shaped function of differences between a new portfolio and the current portfolio. The vagueness and ambiguity in the probability distribution of the portfolio return are modeled by trapezoidal fuzzy numbers, and this PO problem is solved by a modified ABC algorithm. The proposed approach generates the initial population by using a chaotic initialization scheme based on logistic equation to improve the convergence rate. In the neighborhood search, the best solution information is exploited as in PSO algorithm. Both in the employed and the onlooker bee phases, because a new solution may violate the constraint related to the desired number of assets in the portfolio (K), the K largest values of the new solution are retained, and all the other dimensions are set to zero. In the experiments, the effect of the real-world constraints on optimal investment decision was analyzed and the performance of the proposed approach was compared to those of the GA, SA, PSO, DE, and standard ABC algorithms on a dataset of the 30 stocks obtained from the Shanghai Stock Exchange between November 2004 and November 2005. From the results, it was concluded that real-world constraints influence the optimal investment strategy significantly, and the proposed approach has a better performance compared to the other algorithms in the study.

Seyedhosseini et al. [96] combined harmony search (HS) and ABC algorithms to draw efficient frontier portfolios for PO problem. After generating the harmonies, the ABC algorithm is applied followed by the steps of HS. When the constraint related to the number of assets is violated, the relative effect of each asset on the fitness function is measured. The assets with higher relative effects are likely to be added to the collection while the assets with a lower effect are likely to be removed. The historical data was collected from 102 active companies in the Tehran Stock Exchange. The proposed hybrid approach was compared to HS and GA, and it was stated that the proposed approach is better than the others in obtaining efficient frontiers.

Kalayci et al. [97] presented an ABC based approach to find an optimal solution for cardinality-constrained PO in a reasonable time. In the proposed approach, feasibility enforcement and infeasibility toleration procedures are introduced to the basic ABC algorithm. In feasibility enforcement, the solution is avoided to violate the predetermined limits. Two repairing mechanisms are presented: one is for hard constraint and the other one is for soft constraints. The mean, median, minimum, maximum percentage errors, variance of return error and mean return error, mean Euclidean distance are used to evaluate the validity of the approaches on the DAX 100, Hang Seng, FTSE 100, S&P 100, Nikkei, XU030, XU030 datasets. The proposed approach was compared to an ABC algorithm that utilizes a repair procedure alone and the existing solution approaches in the literature. The experimental results revealed that the proposed approach can produce efficient favorable solutions compared to feasibility enforcement alone. Kalayci et al. [98] proposed a hybrid of the continuous ACO, ABC and GA algorithms for cardinality-constrained PO problem. The elitism strategy of GA and modification rate of ABC algorithm are integrated to the continuous ACO algorithm. On the same datasets, and using similar performance metrics, four different parameter combinations (with and without elitism, with and without ABC modification rate) were examined, and it was found that the proposed approach with elitist strategy and modification rate is very efficient and competitive compared to the other studies in the literature. As Kalayci et al. [97], Mansourinia and Momeni [99] proposed an ABC based approach with feasibility enforcement and infeasibility toleration procedures and applied it to 150 companies' data from the Tehran Stock Exchange during the period from 2014 to 2018. Two versions of the ABC algorithm were investigated and it was shown that both algorithms have a good performance for PO problem, and the second version which uses Deb's rules to calculate the feasibility of the solution can solve PO problem with a higher accuracy and efficiency.

Strumberger et al. [100] hybridized the ABC algorithm and GA to balance the exploration and exploitation better. The hybrid approach replaces the scout bee phase in late cycles with a guided onlooker bee phase which exploits the best solution. The crossover and mutation operators of GA are adopted in the proposed approach. In the initialization, the decision variables are initialized as {0, 1} randomly while the others are initialized between the lower and upper bounds of the parameters. In new solution production, rounding is used to make the decision variables 0 or 1. ABC local search operator is used for the variables other than logical decision variables. Experiments were performed on the dataset collected from the Hong Kong Hang Seng, the German DAX 100, the British FTSE 100, the US S&P 100, and the Japanese Nikkei during the period March 1992 and September 1997. It was concluded that the proposed approach is efficient compare to all cutting edge algorithms, including GA, TS, SA, PSO, ABC-FS and mFA.

The studies are summarized in Table 7.

3.8. ABC algorithm for set covering

The set covering problem (SCP) is a fundamental problem in computer science and important in the field of the approximation algorithms. The minimum-cost set covering problem aims to find a minimum cardinality $J \subseteq \{1, \dots, n\}$ such that each point is covered and $\sum_{j \in J} c_j$ is minimum where c_j is the cost of the subset $S_j \subseteq J$. The problem can be formulated as below:

$$\begin{aligned} &\text{Minimize } Z = \sum_{j=1}^{n} c_j x_j \\ &\text{subject to } \sum_{j=1}^{n} a_{ij} x_j \geq 1, \forall i = \{1, \dots, m\} \\ &x_j \in \{0, 1\}, \forall j = \{1, \dots, n\} \end{aligned} \tag{9}$$

where $A = \{a_{ij}\}$ is an $m \times n$ binary matrix holding the covering possibilities. The first constraint ensures that each row i is covered by at least one column. If x_j is 1, the column j is selected.

Table 7
List of the studies using ABC variants for the Portfolio Selection.

No	Year	Study	Application	Representation
1	2010	Hong-mei et al. [88]	Semi-variance PO	Discrete
2	2012	Wang et al. [89]	Cardinality-constrained PO	Binary+Continuous
3	2012	Chen et al. [91]	Cardinality-constrained PO	Discrete+Continuous
4	2013	Chen et al. [92]	Cardinality-constrained PO	Discrete+Continuous
5	2013	Wang et al. [90]	Cardinality-constrained PO	Binary+Continuous
6	2014	Ge [93]	Cardinality-constrained PO	Mixed Binary and Continuous
7	2014	Tuba and Bacanin [94]	Cardinality-constrained PO	Mixed Binary and Continuous
8	2015	Chen [95]	PO with Real-world Constraints	Mixed Integer and Fuzzy
9	2016	Seyedhosseini et al. [96]	Semi-variance PO	Continuous
10	2017	Kalayci et al. [97]	Cardinality-constrained PO	Continuous
11	2018	Mansourinia and Momeni [99]	Cardinality-constrained PO	Continuous
12	2018	Strumberger et al. [100]	Cardinality-constrained PO	Mixed Binary and Continuous
13	2020	Kalayci et al. [98]	Cardinality-constrained PO	Continuous

Although there are exact algorithms to solve this problem, they are time consuming and can solve instances of limited size. Therefore, to find good solutions in reasonable times, heuristics and metaheuristics have been applied, including the ABC algorithm.

Crawford et al. [101] and Cuesta et al. [102] proposed an ABC algorithm to solve SCP. The proposed approach uses an integer encoding as the encoding rule, a solution holds the column numbers as integers. A solution is modified by changing its position based on a neighbor solution selected randomly. When a duplicate solution occurs, its bee becomes a scout bee and adds random number of columns. A repairing mechanism is applied in case new solution does not meet the constraints. Validity of the approach was tested on 65 standard non-unicost SCP instances available from OR Library, and a comparison among GA, SA, and ant colony algorithm with local search (ANT+LS) was performed. The results showed that the proposed approach is better in terms of robustness, and its performance does not depend on control parameters significantly. Based on the ABC algorithm in this work, Crawford et al. [103,104] proposed a GA for parameter tuning of the ABC algorithm and applied it to SCP problem. The number of food sources, the limit value, the columns to add and the columns to be eliminated are encoded in a chromosome of GA. The proposed approach was tested on 65 standard non-unicost SCP instances available from OR Library. Crawford et al. [103] compared against ANT+LS and Crawford et al. [104] compared the proposed approach against binary cat swarm optimization, binary firefly optimization, binary shuffled frog leaping algorithm, binary electromagnetism-like algorithm and ABC in [102] based on the best solution, average solution and relative percentage deviation. It was reported that the ABC algorithm could find the optimal solution consistently, and it has higher success rate compared to the ant algorithm. In another study, Crawford et al. [105] compared the approach against FA, GA, TS, SA and ANT+LS. It was stated that both FA and ABC produce high quality of the solutions.

The studies are summarized in Table 8.

3.9. ABC algorithm for unit commitment

Unit commitment problem (UCP) aims to schedule the operation of the generating units (setting on/off status) at minimum operating cost (TOC) production cost, start-up and shut-down costs meeting the demand and other equality and inequality constraints.

$$TOC = \sum_{t=1}^{T}\sum_{i=1}^{N}\left[C_i\left(P_i^t\right)U_i^t + U_i^t\left(1-U_i^{t-1}\right)SU_{i,t} + U_i^{t-1}\left(1-U_i^t\right)SD_{i,t}\right] \quad (10)$$

where N is number of thermal units and T is total scheduled time horizon, $SU_{i,t}$ start-up cost and $SD_{i,t}$ is a constant defined for each unit.

$$C_i\left(P_i^t\right) = a_i + b_i \times P_i^t + c_i \times \left(P_i^t\right)^2 \quad (11)$$

where a_i, b_i and c_i are cost coefficients of fuel cost of generating unit i.

$$SU_{i,t} = \left\{\begin{array}{l} HS_i,\ \text{if } T_{i,down} \le T_{i,off}^t \le T_{i,down} + T_{i,cold} \\ CS_i,\ \text{if } T_{i,off}^t > T_{i,down} + T_{i,cold} \end{array}\right\} \quad (12)$$

The UCP problem has several system and unit constraints:

(i) Ramp-rate limit constraints: The power generated at the output of ith thermal unit at time t may affect its output power in the next time step:

$$\begin{array}{ll} P_i^t - P_i^{t-1} \le UR_i, & \text{as generation increases} \\ P_i^{t-1} - P_i^t \le DR_i, & \text{as generation decreases} \end{array} \quad (13)$$

where UR_i and DR_i are the ramp up and ramp down limit of unit i.

(ii) Generation limit constraint: Power generation of each generating unit must be within its minimum ($P_{i,t}^{\min}$) and maximum ($P_{i,t}^{\max}$) range.

$$P_{i,t}^{\min} \le P_i^t \le P_{i,t}^{\max} \quad (14)$$

(iii) Power (or load) balance constraint: The generated total power of all committed units must be less than or equal to the load demand at that particular interval t.

$$\sum_{i=1}^{N} P_i^t U_i^t \le P_D^t \quad (15)$$

(iv) Minimum capacity limit constraint: The total minimum power output of the committed units at each hour must be less than or equal to the load of the corresponding hour.

$$\sum_{i=1}^{N} P_i^{\min} U_i^t \le P_D^t,\ t = 1, 2, \ldots, T \quad (16)$$

(v) Spinning reserve constraint: Spinning reserve must be available during the operation of a power system

$$\sum_{i=1}^{N} P_i^{\max} U_i^t \ge P_D^t + R^t,\ t = 1, 2, \ldots, T \quad (17)$$

(vi) Minimum up and down time constraints: A unit must be on/off for a minimum period before committing or decommitting.

$$U_i^t = \left\{\begin{array}{ll} 0 \to 1, & \text{if } T_{i,off}^{t-1} \ge T_{i,down} \\ 1 \to 0, & \text{if } T_{i,on}^{t-1} \ge T_{i,up} \\ 0 \text{ or } 1, & \text{otherwise} \end{array}\right\} \quad (18)$$

Table 8
List of the studies using ABC variants for the Set Covering Problem.

No	Year	Study	Application	Representation
1	2014	Crawford et al. [101]	SCP	Integer
2	2014	Cuesta et al. [102]	SCP	Integer
3	2014	Crawford et al. [103]	SCP	Integer
4	2014	Crawford et al. [105]	SCP	Integer
5	2017	Crawford et al. [104]	SCP	Integer

Computational cost of solving UCP is expensive due to curse-of-dimensionality, especially for large power systems and meta-heuristics can produce near-optimal solutions.

Chandrasekaran et al. [106] proposed a binary ABC to produce the ON/OFF status of the generating thermal units and used the real coded ABC for the economic dispatch. In the proposed binary ABC, changes of probability are encoded in the solutions and the unit status is decided based on a threshold value. In order to scale the solutions within [0,1], the hyperbolic tangent function is utilized. When the constraints are not satisfied, a problem-specific repair scheme is performed. When they are satisfied and an optimum commitment schedule is obtained, the economic dispatch is done for the feasible positions using the real coded ABC. Performance of the proposed algorithm was verified on a standard ten-unit system, on IEEE 118-bus test system and IEEE RTS 24-bus system. It was reported that the proposed binary ABC is efficient in generating feasible schedules.

Chandrasekaran and Simon [107] extended the work in [106] and presented a fuzzified binary/real coded ABC algorithm which can handle the multiple objectives in the unit commitment problem, such as fuel cost, emission and reliability level of the system. The multiple conflicting objectives are transformed into a single-objective optimization problem by weighting fuzzy coefficients. The fuzzy membership functions are formulated by quasi-optimization solutions and tuned using real coded ABC. The objective function for the fuel cost and emission are tailored as a strict monotonically decreasing function to be minimized. Efficiency of the proposed method is verified on six unit system which considers the fuel cost and emission as a quadratic function, IEEE 30-bus system in which the fuel cost is defined as a quadratic function and the emission function is given as sum of the quadratic and exponential term, ten unit system in which the cost and emission functions are defined by both sinusoidal and exponential terms with a quadratic function, 40 unit system and IEEE RTS 24-bus system. The proposed technique was compared against the other methods reported in the literature and it was shown that the proposed method is superior and has potential for solving non-smooth multi-objective problems in a power system.

Govardhan and Roy [108] utilized the global best solution information in the local search of ABC algorithm and implemented traditional priority list method based on full load average production cost. The approach was tested on IEEE 26 unit test system. It was stated that the solution produced by the approach provides more profit compared to traditional unit commitment.

Singhal et al. [109] presented a binary ABC algorithm which uses dissimilarity between the binary vectors instead of subtraction of real numbers in the neighbor solution production. When a solution does not meet the constraints, the solution is repaired by a problem-specific heuristic. Performance of the proposed approach was tested on 2 test systems with quadratic nature comprising of 5-unit and 26-unit over the scheduling time horizon of 24 h, and the results were compared to those of state-of-the-art methods. It was concluded that the proposed approach is better in minimizing the total operating cost without any constraint violation. Singhal et al. [110] extended this approach by using an intelligent scout bee phase that replaces the exhausted solution with the global best solution and GA crossover operator in the local search to obtained more diversified solutions. The proposed approach was tested on a problem with quadratic cost and 10, 20, 40, 60, 80, 100, 150, 200 and 300 thermal units over 24-hour time interval in which the ramp rate constraint is included. The approach was compared to state-of-the-art methods in the literature, and it was shown that the approach is superior over the other methods in terms of production cost and robustness using small population sizes. It was also noted that the genetic crossover accelerates the search and decreases the computational cost and most of the computational budget is dedicated to the repairing heuristic. Singhal et al. [111] further extended this approach incorporating a local search in the ABC algorithm. The local search operator searches the best possible solution in the neighborhood of the current solution based on the swap moves and simultaneously flips the status of other unit providing that the number of online units is fixed. The efficiency of the method was tested on the systems with 10, 20, 40, 60 and 100 thermal units over 24-h scheduling time horizon with 1-h time interval and the real Turkish interconnected power system consisting of eight thermal units over 8-h scheduling time horizon. From the results, it was reported that the approach produces high quality solutions in terms of total generation cost and the local search adopted improves the convergence abilities of the ABC algorithm. Singhal et al. [112] compared these three approaches for solving the Wind-Thermal Unit Commitment problem in which the uncertainty of wind power is introduced by the Weibull probability density function to calculate the overestimation and underestimation costs due to the fluctuation. The experiments were performed on an IEEE 10-unit thermal system combined with a wind farm over the planning period of 24 h. It was shown that the binary ABC algorithm with intelligent scout and genetic crossover produces better results compared to the other methods implemented in the study.

The studies are summarized in Table 9.

3.10. ABC algorithm for bike positioning

Since bicycles are environment-friendly and economical, people prefer them in the transportation. An efficient way in bicycle trips is bike sharing that provides rental bicycles in specified stations. However, they suffer from the imbalanced distribution of the bikes in which enough bikes cannot be provided at each station. In order to avoid the imbalance and distribute bicycles to the stations balancedly as much as possible, the bicycles are transferred between the stations. Therefore, determining whether a bike will travel from station i to station j and routing the transporting vehicles among the stations to minimize the distance, service time is an optimization problem.

Shui and Szeto [113] modeled the public bike positioning problem which includes two linear objectives and constraints.

Table 9
List of the studies using ABC variants for the Unit Commitment Problem.

No	Year	Study	Application	Representation
1	2012	Chandrasekaran et al. [106]	UCP	Binary+Real
2	2012	Chandrasekaran and Simon [107]	UCP	Binary+Real
3	2013	Govardhan and Roy [108]	UCP	N/A
4	2015	Singhal et al. [109]	UCP	Binary
5	2015	Singhal et al. [110]	UCP	Binary
6	2015	Singhal et al. [111]	UCP	Binary
7	2016	Singhal et al. [112]	Wind Thermal UCP	Binary

The problem and the constraints can be formulated as follows

$$
\begin{aligned}
&\min Z = \max\left(\sum_{v\in V} D_{dis,v} - TDD, 0\right) + \sum_{v\in V}(T_v + S_V)\\
&\text{subject to} =\\
&q_{ijv} \le k x_{ijv}, \forall i,j \in N_0, i \ne j, \forall v \in V\\
&\sum_{j\in N_0, j\ne i} x_{ijv} = \sum_{j\in N_0, j\ne i} x_{jiv}, \forall i \in N_0, \forall v \in V\\
&\sum_{v\in V}\sum_{j\in N_0, j\ne i} x_{ijv} = 1, \forall i \in N\\
&T_v + S_v \le l_T, \forall v \in V\\
&g_{jv} \ge g_{iv} + 1 - M(1 - x_{ijv}), M \text{ is a penalty factor},\\
&\quad \forall i \in N, j \in N_0, i \ne j, \forall v \in V\\
&x_{ijv} = \{0,1\}, \forall i \in N, j \in N_0, i \ne j, \forall v \in V\\
&D_{dis,v}, D_{sur,v} \in Z^+\\
&q_{iv} \ge 0, \forall i \in N_0, \forall v \in V
\end{aligned}
\tag{19}
$$

$$
\begin{aligned}
&D_{dis,v} = \sum_{j\in N_0}\left\{\max\left(d_j - \sum_{i\in N_0} q_{ijv}, 0\right)\right\}, \forall v \in V\\
&D_{sur,v} = \sum_{j\in N_0}\left\{\max\left(\sum_{i\in N_0} q_{ijv} + s_j - k, 0\right)\right\}, \forall v \in V\\
&T_v = \sum_{i\in N_0}\sum_{j\in N_0} x_{ijv} t_{ij}, \forall v \in V\\
&S_v = \left(\sum_i\sum_j x_{ijv} s_j - D_{sur,v}\right) L + U, \forall i,j \in N_0, \forall v \in V
\end{aligned}
\tag{20}
$$

x_{ijv} indicates whether the vehicle v directly travels from node i to node j, q_{iv} is the number of bikes carried by vehicle v when it travels directly from nodes i to j, N_0 is the set of stations, N is the set of stations excluding depot, V is the set of transporting vehicles, s_i is the number of excess bikes at station i, d_i is the number of outstanding bikes at station i, k is the capacity of the vehicle, t_{ij} is traveling time from station i to j, L is the time needed to load a bike onto a vehicle, U is the time needed to unload a bike from a vehicle, TDD is tolerance limit, $D_{dis,v}$ is the total demand dissatisfaction along the whole route of vehicle v, $D_{sur,v}$ is the total demand surplus along the whole route of vehicle, T_v is the travel time of the whole route of vehicle v, S_v is the service time of the whole route of vehicle v, g_{jv} is the auxiliary variable associated with node j of vehicle v used for the sub-tour elimination constraint [113].

Shui and Szeto [113] proposed an ABC algorithm to solve bike repositioning problem. In the proposed ABC algorithm, solutions are represented by a sequence of the bike stations and the swap-reverse operator is used to generate new solutions. In the onlooker bee phase, if the value of trial counter of the source is the largest among all the existing food sources and the new solution is better than the old solution, the new solution is replaced with the old one. In the scout bee phase, a neighborhood operator is applied to generate a new solution. They investigated various neighborhood operators on an instance with one vehicle and passing 60 bike stations and compared the modified approach with basic ABC algorithm on eight instances with up to four vehicles and 99 stations. It was demonstrated that the proposed approach is superior on the instances in the experiments.

Tian and Xie [114] applied ABC algorithm for bike scheduling problem which considers time-windows, transport costs and service satisfaction objectives in which each station is visited only once and one vehicle is used. In the experiments, an instance with 9 stations is used. The results revealed that the approach provides high service satisfaction and minimum transport costs.

Shui and Szeto [115] solved green bike repositioning problem which also considers the fuel consumption due to engine performance and the tractive power, and CO2 emission cost of the repositioning vehicle in addition to the objectives in basic problem. The proposed problem is divided into sub-stages that cover only a part of the horizon and an enhanced ABC algorithm integrated with a route truncation heuristic for revising the route is applied in each stage. A rolling horizon approach is adopted in the study to handle the dynamic demand. In the proposed ABC algorithm, a solution corresponds to the sequence of visited nodes. For finding neighbor solutions, a neighborhood operator is randomly chosen from a pool of operators including random swaps, subsequence reverse, and random swaps of reversed subsequence. In the onlooker bee phase, a solution is replaced with the new solution if its trial counter is the largest among the other solutions and new solution's quality is higher than the old solution. In the scout bee phase, the exhausted solution is replaced with a solution in the neighborhood of the old solution instead of random search. The proposed approach was compared to GA on 45 instances with different network sizes up to 180 nodes, horizon lengths. The results demonstrated that when shorter stage length is used, better results can be achieved and the proposed ABC algorithm outperforms GA.

The studies are summarized in Table 10.

3.11. ABC algorithm in satellite technology

Satellite technology is significant in spatial monitoring and scheduling. Efficient scheduling of the satellite resources is an important optimization problem in this field. As the spatial targets increase, it becomes difficult to solve, and the time complexity grows dramatically.

Zhao et al. [116] presented a binary ABC algorithm for satellite resource allocation problem with multiple objectives and constraints. In the satellite resource scheduling problem, it is assumed that each observing satellite has a sensor to observe the space target and the monitoring of each space target requires a number of satellites. The availability of a satellite at each stage is encoded using the matrix group coding method. In the initialization of the proposed ABC algorithm, the logistic function is used to map a value produced to the binary value. Generating a new solution in the neighborhood of a solution is carried out by function overlap which replaces a randomly selected column of two solutions and then, new solution is repaired based on the constraint matrix. Since the problem has multiple objectives, the probability values are calculated using dominance of the

Table 10
List of the studies using ABC variants for the Bike Positioning Problem.

No	Year	Study	Application	Representation
1	2015	Shui and Szeto [113]	Bike Positioning	Discrete
2	2017	Tian and Xie [114]	Bike Scheduling	N/A
3	2018	Shui and Szeto [115]	Green Bike Positioning	Discrete

solutions. The experiments were performed on the Walker Constellation data to observe the space target with different number of targets. It was reported that as the number of targets increases, switching times and relaxation index rise concurrently and all the satellite working time is balanced.

Luo [117] combined the genetic operators and two-phase repair operator in the ABC algorithm to solve satellite photograph scheduling problem which involves scheduling the onboard cameras of the Earth observation satellite. The problem has too many variables and constraints resulting a sparse logical constraint coefficient matrix. Since satellite photograph scheduling is binary problem, the decision vector is binary coded, and one-point crossover and simple mutation are employed to generate new solutions. Infeasible solutions are repaired and a greedy selection is applied. The proposed approach was compared with TS, quantum PSO and a variant of HS algorithm on a randomly generated test set and an open source library with 13 instances without any recording capacity constraint and 7 instances with a fixed memory limitation. It was also concluded that the proposed algorithm outperforms the other three state-of-the-art algorithms

The studies are summarized in Table 11.

3.12. ABC algorithm for other binary problems

There are also some other applications of the ABC algorithm to various binary problems in different fields.

Oliveira et al. [118] extended the basic ABC algorithm to be able to operate in binary space. In the new variant of ABC, a sigmoid function converts real values to binary values. The proposed approach was used for identification of transients in a nuclear power plant and shown to be efficient compared to PSO algorithm.

Pamparà and Engelbrecht [119] proposed three binary ABC variants, including angle modulated ABC, binary ABC with PSO updating rule, and binary ABC converting the continuous variables to binary values in the fitness function. The presented approaches were tested on a number of optimization problems and it was concluded that the angle modulated binary ABC outperforms the other variants except for the angle modulated PSO algorithm.

Qi et al. [120] proposed a binary ABC algorithm to minimize both the reader-to-reader interference and total system transaction time in RFID reader networks. The local search operator in the basic ABC is retained but the result is rounded to obtain a binary value. The experiments were performed on two test cases with 30 and 60 readers. The proposed binary ABC algorithm was compared to binary PSO and binary GA algorithm. The results showed that binary ABC could find the optimal schedule in less number of cycles and time, and when the number of readers increases, binary ABC is still robust and consistent.

Kaur and Singh [121] developed a graph partitioning algorithm which combines the ABC algorithm and inver-over operator. The proposed approach is used to solve Min-Cut problem in which a set of nodes in a graph is partitioned into subsets to minimize the number of edges between the partitions. A binary vector is used to represent solutions where a bit value one indicates a cell is connected to net. In the employed and onlooker bee phases, inver-over operator is used to generate new solutions. Simulated annealing is applied after the onlooker bee phase for further improvement. Efficiency of the proposed approach was validated on UCLA small circuit partitioning instances and the experimental results demonstrates that the proposed approach is better than UCLA branch and bound partitioner over different size range when the average results are considered.

Haris et al. [122] proposed binary ABC and TS algorithms for Multi User Detection (MUD) in Turbo Trellis Coded Modulation based Space Division Multiple Access Orthogonal Frequency Division Multiplexing system (SDMA-OFDM). In the proposed ABC algorithm, to avoid pure random search which may cause inefficiency, the initial sources are constructed using the estimated length-L transmitted symbol vector generated by minimum mean square error combiner and represented by complex symbol representation based on the binary encoding. In order to generate a solution, a randomly selected symbol is removed and another different symbol from a randomly selected solution is inserted. The simulation results were obtained using a quadrature amplitude modulation scheme. ABC, GA and TS algorithms were compared in terms of BER performance and convergence rate. It was concluded that ABC and TS optimized multiple-input and multiple-output (MIMO) symbol detection mechanisms are able to find near-optimal solutions with reduced computational complexity.

Wei and Hanning [123] presented a binary ABC algorithm in which the solutions are encoded in binary space and rounding is applied to the generated solutions to convert them into binary values. The proposed approach, binary GA and binary PSO algorithms were verified on discrete benchmark functions. From the convergence graphs, solution quality metrics and ANOVA tests, the binary ABC was shown to have better performance compared to binary GA and PSO algorithms.

Liu et al. [124] aimed to utilize a binary ABC algorithm to minimize interfering and reader collusion between multiple readers physically located near one another and total system transaction time in RFID reader networks. The food source locations are defined in a binary space and the values obtained by the search operators of the basic ABC algorithm are normalized and thresholded to map the values into binary space. To verify the efficiency of the proposed method, four RFID reader network cases with 30, 60, 120 and 200 readers were scheduled and it was shown that the proposed binary ABC algorithm can accommodate a considerable potential for scheduling large scale RFID reader networks.

Bayraktar et al. [125] presented a memory-integrated ABC algorithm for solving 1-D bin-packing problems in which the decision parameters are binary and the aim is to pack all the items into as minimum number of bins and to minimize the total wasted space. The authors integrated tabu-list concept with the ABC algorithm to avoid the non-promising regions in the next iterations. The proposed approach was compared to the First-Fit Packing, Best-Fit Packing, Worst-Fit Packing, Best-Fit Decreasing Packing and basic ABC algorithms on two datasets from OR library. It was demonstrated that memory integration is useful especially on highly complex and large-size problem instances.

Jia et al. [126] implemented a binary ABC algorithm in which the arithmetic operation used in the basic ABC algorithm is replaced with a binary bitwise operation. The proposed approach was compared to three binary ABC variants and GA on a set of 13 widely used benchmark functions. It was demonstrated that

Table 11
List of the studies using ABC variants in the Satellite Technology.

No	Year	Study	Application	Representation
1	2017	Zhao et al. [116]	Satellite Resource Scheduling	Discrete Matrix Group Coding
2	2019	Luo [117]	Satellite Photograph Scheduling	Binary

the proposes approach produces better or similar results than the other algorithms does in terms of accuracy, convergence rate and robustness.

Yahya and Saka [127] presented a multi-objective Levy flight ABC algorithm for construction site layout planning considering the safety hazards and the total flow cost between facilities. In the problem the orientation of each facility is binary digit indicating whether it is horizontal or vertical. The approach was compared against the basic multi-objective ABC model and max–min Ant system. From the results, it was stated that the proposed approach produces better results compared to the other methods in the study. It was also noted Levy flight in the employed bee phase contributes to the efficiency of the algorithm by introducing randomness in the local search.

Mandala and Gupta [128] proposed a global-best guided binary ABC algorithm for GENCOs' profit maximization under pool electricity market. The proposed approach encodes the solutions in binary space and new solution generated after applying local search operator is thresholded by a tangent hyperbolic function. A bit mutation operator is performed on new solution to increase diversity. The approach was validated on IEEE 30-bus system and IEEE-57 bus system and compared to basic ABC and basic global best ABC. It was demonstrated that the proposed approach is superior in terms of solution quality, accuracy, computation time and convergence speed.

Zhang and Ye [129] proposed an ABC based approach for solving the rectilinear Steiner minimal tree (RSMT) problem which is a basic problem in very largescale integrated circuit (VLSI) circuit design. Binary encoding is used to represent solutions and a key-node-based heuristic algorithm named greedy randomized adaptive search procedure heuristic distance network heuristic (GRASP-DNH) to construct a feasible the Steiner tree. The objective function is defined as the weight of the feasible solution. To generate a new solution, two local search operators are combined in the employed and onlooker bee phases of the algorithm, including Walk Around Search and Purifying Search. In addition to the global search strategy of the basic ABC algorithm, a merging operation is performed to improve the global search ability. The approach was validated on the instances in SteinLib and 22 commonly used benchmark circuits for the obstacle avoiding RSMT (OARSMT) and RSMT problems. It was concluded that the proposed approach produces better solution quality and running time compared to the jumping PSO and LS_GSTP algorithms, and the approach improves all the best-known solutions for OARSMT and RSMT problems.

Al-Salamah [130] proposed an ABC algorithm to minimize the makespan for a single batch-processing machine in which the decision variables are binary indicating whether a job is assigned to a batch, and the objective function has discontinuity. The constraints which ensures a batch do not violate the machine's capacity are penalized and added to the objective function. The first fit decreasing algorithm is modified to generate initial solutions which are not necessarily feasible. To produce a neighbor solution, the move operation moves a job to another batch. In the scout bee phase, a new solution is generated randomly by distributing each job to a randomly chosen open batch. The ABC algorithm was compared to CPLEX optimizer in terms of running time on the instances drawn from the uniform distribution. From the results, it was seen that while CPLEX cannot solve problems with 50 jobs and more, it is faster than ABC for problems with 30 jobs and less. Moreover, the experiments on the control parameters of ABC showed that the feasibility of the solution is influenced by the values of the control parameters.

Rigakis et al. [131] implemented a binary ABC algorithm to evolve game strategies for Iterated Prisoner's Dilemma. For each played games, the binary encoding scheme assigns 1 when a player cooperates and assigns 0 when the player defects. The fitness function corresponds to the payoff of each player. The neighbor solution production in the basic ABC algorithm is applied and sigmoid function is used to map the values into binary space. The proposed approach was compared to PSO and man-made strategies, including random, always cooperate, pavlov, tit-for-tat, evil tit-for-tat on benchmark strategies for 20 different executions. They considered the binary ABC with and without memory. It was demonstrated that the ABC algorithm with memory produces better strategies with higher payoff compared to the original memory-less ABC algorithms and PSO is better than ABC without memory but worse than the ABC with memory.

Arul Jeyaraj et al. [132] proposed a fuzzified binary ABC algorithm for solving multi-objective optimal placement of phasor measurement units (PMU). The aim is to minimize the number of phasor measurement units and to maximize the voltage stability level of the system, simultaneously. In the problem, the possibility of PMU on a bus, elements of the connectivity matrix, and the possibility of a PMU on a load bus are binary variables. In order to convert real representation to discrete one, the probability value is calculated for each solution to determine whether the state variable is set to 0 or 1. In the initialization, violated solutions are set to zero to uninstall the PMU. After neighbor solution production, a repair strategy is performed for constraint management. Fuzzy membership functions are employed for each objective function to determine the best solution. The effectiveness of the ABC algorithm was tested on standard IEEE 14, 30 and 57 bus systems and compared to other conventional and non conventional techniques. it was observed that the proposed binary ABC approach is able to determine the optimal number of PMUs and their locations efficiently.

Kim et al. [133] proposed an ABC clustering approach using binary encoding for wireless sensor network clustering. The method tries to minimize a weighted combination of the average dissipated energy and the standard deviation of residual energy of the nodes in the network for a long network lifetime. The cluster heads and clusters are optimized by the ABC algorithm in the set-up phase and network data is transmitted to these nodes and then to the base station in the steady-state phase. Solutions that correspond to network configurations are represented by binary encoding. New sources are produced by probabilistically changing cluster head to sensor node or vice versa for randomly selected nodes of the current food source. The proposed approach was compared to PSO, group search optimization, low-energy adaptive clustering hierarchy (LEACH), LEACH-centralized (LEACH-C), and the hybrid energy-efficient distributed clustering algorithms on randomly generated networks with different number of nodes for two scenarios, one with the sink node located at the origin and the other with the sink node located at the center of the network. The experimental results revealed that the designs based on the second scenario are better compared to those based on the first one. The performance of group search optimization is better than that of PSO, while the proposed ABC algorithm produces the best solutions for all instances. The proposed approach is generally better than low-energy adaptive clustering hierarchy, LEACH-C and the hybrid energy-efficient distributed clustering algorithms and is substantially better than the others as the number of nodes increases.

Gao et al. [134] presented an ABC based approach for urban traffic light scheduling problem to minimize the network-wise total delay time of all vehicles in a fixed time window. In the approach, each intersection is two bit binary string and therefore, the length of solution vector is two times of the intersection number. Three local search strategies are proposed for three neighborhood structures of a traffic network. The first local search reverses the traffic lights of the selected intersection, the second one reverses the traffic lights of a group of selected intersections and the third one does the traffic lights of the intersections in the sub-region in one time interval. The Taguchi method was used for the parameter settings of the ABC algorithm. Sixteen instances with different problem-scales were used to validate the approach. The experiment results revealed that the proposed approach is better than four metaheuristics and CPLEX.

Li et al. [135] developed an ABC algorithm for damaged ship righting optimization which is a hard due to the feasible combinations of counter-flooding approaches with different compartments and loads are numerous. The aim is to minimize the inclination angle under a group of constraints regarding stability, floatation and available compartments and loads. The range of moving loads, removing loads and ballast loads are discrete variables. A train ferry with two different damage scenarios was used to validate the proposed approach. The results indicated that the ABC-based approach can produce better flotation state with lower calculation time cost.

Kong et al. [136] established a combinatorial optimization model with 0–1 integer decision variable to solve the interval-valued intuitionistic fuzzy multi-attribute decision-making (MAGDM) problems based on the weighted relative distance. An improved ABC algorithm was proposed in which binary variables are coded as integers to enhance coding efficiency and reduce dimension. A selection strategy based on rank values is adopted because it is hard to measure fitness of solutions due to the relativity of decision-making. In local search operation, they utilizes the information of elite individuals to improve convergence ability. The research group dispatch and group weapon-group target assignment problems were used to test the approach and compare to some state-of-the-art approaches. It was concluded that traditional methods cannot solve the problems with high dimensions while the improved ABC algorithm can effectively solve and produce satisfactory solution within a fixed time.

Chen et al. [137] implemented a binary ABC incorporated with a local search procedure to solve max-cut problem which aims to partition the set of vertices into two subsets, such that the weight sum of the edges across two subsets is maximized. In the approach, solutions in the population are encoded as continuous variables. Before evaluating in the cost function, the continuous values are converted to binary vectors using surjection mapping, and a local search is applied to improve the convergence. They tested their approach on 24 instances from the literature and compared it to the ant colony algorithm and hierarchical social metaheuristic. The results showed that the proposed approach improves more instances out of 25 instances than the others do.

Chen et al. [138] presented a network repair model to ensure that the information flow can be maintained when a military communications network is damaged by an attack. The proposed model introduces new edges to maximize network invulnerability with the consideration of connect cost limit and network connectivity, and it is solved by a binary ABC algorithm. Binary logical operators are used in the operators of the algorithm. Performance of the proposed approach was checked on random attacks and intentional attacks and compared to random addition, low degree first and low betweenness first algorithms. The simulation results demonstrated the efficiency of the proposed approach in repairing the damaged military communication network.

Saad et al. [139] developed a goal programming-based multi-objective artificial bee colony optimization for topological designing of distributed local area networks based on network reliability, network availability, average link utilization, monetary cost, and network delay metrics. Goal programming (GP) is incorporated to aggregate the multiple design objectives into a single objective function. The proposed method integrates simulated evolution in multi-objective ABC algorithm to enhance local search ability. A solution is encoded as a 2-D matrix, where one means that there is a link between two nodes and zero indicates that there is no link. Solutions are checked to validate their feasibility. If a solution violates any constraint, it is discarded. A new solution is generated by breaking a randomly selected link in the current solution and building a new link. The experiments were carried out on five test cases consisting of 15, 25, 33, 40, and 50 nodes local area networks. The results indicated that the proposed approach could efficiently produce higher quality solutions.

The studies are summarized in Table 12.

4. Integer programming artificial bee colony algorithm

An integer programming problem in which all variables are integer is called a pure integer programming problem defined in Eq. (21).

$$min\, f(x), x \in S \subseteq Z^n \tag{21}$$

where Z is the set of integers, and S is the feasible region.

4.1. ABC algorithm for integer communication problems

Wang et al. [140] applied ABC based on angular concept and distance between angles to minimize peak-to-average power ratio (PAPR) in OFDM systems. Each food source is represented by a vector of phase factors represented by angles and the angles are converted to $\{-1, 1\}$ or $\{-j, j, -1, 1\}$ values when W=2 or 4, respectively. The proposed algorithm corporates the global best solution information and a learning factor in new solution production. Taspinar et al. [141] proposed an ABC-based Partial transmit sequences (PTS) scheme for multicarrier code division multiple access (MC-CDMA) to minimize PAPR. In the proposed approach, each food source is represented by a vector of phase factors, $b_i \in \{-1, 1\}$. The solutions generated in the continuous space are converted to binary values, and then, the elements with the value of 0 are replaced with 1 and the elements with the value of 1 are replaced with -1. The neighborhood operator in the basic ABC is substituted with the cycling bit flipping mechanism in which a neighbor is generated based on Hamming distance. The proposed approach was compared to iterative flipping algorithm for PTS, the gradient descent search algorithm for PTS, random search PTS and optimum PTS methods found by exhaustive search. It was shown that an important improvement is obtained in PAPR reduction in addition to having low complexity. The proposed approach converges faster than the random search PTS and decreases search cost by a factor of about four. Taspinar et al. [142] applied this approach to PAPR reduction in orthogonal frequency division multiplexing (OFDM) systems. It was compared to random search and shown that the proposed approach reduces the complexity of the conventional PTS in the OFDM system. Dung et al. [143] applied ABC based on angular concept and distance between angles to minimize PAPR in coherent optical OFDM systems. The angles

Table 12
List of the studies using ABC variants for Other Binary Problems.

No	Year	Study	Application	Representation
1	2011	Oliveira et al. [118]	Identification in Nuclear Power Plant	Binary
2	2011	Pampará and Engelbrecht [119]	Benchmark Problems	Binary
3	2012	Qi et al. [120]	RFID Network Scheduling	Binary
4	2012	Kaur and Singh [121]	Circuit Min-Cut	Binary
5	2012	Haris et al. [122]	MUD in SDMA-OFDM	Binary
6	2012	Wei and Hanning [123]	Benchmark Functions	Binary
7	2013	Liu et al. [124]	Reader Collision in RFID	Binary
8	2014	Bayraktar et al. [125]	1-D Bin Packing	N/A
9	2014	Jia et al. [126]	Benchmark Functions	Binary
10	2014	Yahya and Saka [127]	Construction Site Layout Planning	Mixed
11	2014	Mandala and Gupta [128]	Profit Maximization	Binary
12	2015	Zhang and Ye [129]	RSMT	Binary
13	2015	Al-Salamah [130]	Single Machine Batch Processing	Binary
14	2016	Rigakis et al. [131]	Iterated Prisoner's Dilemma	Binary
15	2016	Arul Jeyaraj et al. [132]	Placement of Phasor Units	Binary
16	2017	Kim et al. [133]	WSN Clustering	Binary
17	2017	Gao et al. [134]	Traffic Light Scheduling	Binary
18	2018	Li et al. [135]	Damaged ship righting	Discrete
19	2019	Kong et al. [136]	Fuzzy Multi-attribute decision-making	Integer
20	2019	Chen et al. [137]	Max-Cut Problem	Continuous/Binary
21	2020	Chen et al. [138]	Repair Military Network	Binary
22	2018	Saad et al. [139]	Network Topology Design	Binary Matrix

are converted to $\{-1, 1\}$ or $\{-j, j, -1, 1\}$ values when W=2 or 4, respectively. It was concluded that the proposed approach is helpful in terms of reduced PAPR and low complexity even in the large number of sub-blocks for the PTS method. Yu et al. [144] used the modification proposed in [140]. From the results, they concluded that their approach is efficient in terms of both PAPR reduction and BER without compromising the computational complexity. Cheng et al. [145] used a chaotic phase sequence and the sign function value of the elements are considered as the phase factor values. Flipping operator is applied to generate a new solution. They obtained superior performance over the other existing PTS schemes. Yin et al. [146] proposed multiuser detector based on ABC and suboptimal code mapping in direct-sequence ultra-wideband systems under the additive white Gaussian noise channel. A suboptimal multiuser detection and its variations correspond to the initial values encoded by -1 or +1. The fitness of a solution is the criterion of optimum multiuser detection. The BER performance, user capacity, and the near-far effect resistant ability of the proposed approach is found to be very close to the optimal solution and its complexity is lower than the optimal multiuser detector. Ashrafinia et al. [147] presented a discrete ABC algorithm for the joint symbol detection problem to find a nearly optimal solution in real time in a multidevice Space-Time Block Code (STBC) MIMO communication system. To generate a neighbor solution in both the employed and onlooker bee phases, a random integer is generated between x^l_j and $2x^i_j - 2x^l_j$. The proposed approach was shown to have less computational complexity and significantly better performance compared to GA, Estimation of Distributions Algorithm (EDA), and Biogeography-Based Optimization (BBO) for this problem.

The studies are summarized in Table 13.

4.2. ABC algorithm for weapon target assignment

The weapon target assignment (WTA) problem aims to minimize total expected survival value of targets by assigning weapons to targets optimally. The problem can be formulated by Eq. (22) as a nonlinear integer programming problem:

$$\begin{aligned} &\min f(x) = \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} q_{ij} x_{ij} \right) \\ &\text{subject to} : \sum_{j=1}^{n} x_{ij} \leq W_i, \forall i = 1, \ldots, m \\ &x_{ij} \geq 0, \forall i = 1, \ldots, m, \forall j = 1, \ldots, n \end{aligned} \tag{22}$$

where n is the number of targets and m is the weapon types. x_{ij} is the number of weapons of type i to be assigned to target j, V_j is the value of the target j, W_i is the number of weapons of type i, p_{ij} is the probability of destroying target j by type i weapon, q_{ij}: the probability of survivability of target j.

Durgut et al. [148] introduced an ABC algorithm using integer representation to solve WTA problem. To generate a neighbor solution, swap operator is used. They compared the effectiveness of swap, insertion and inversion operators. It was concluded that adopting swap operator in ABC algorithm is more efficient while the others consume more time. Moreover, they have found that ABC and SA can find similar results while the proposed approach is faster than the SA on all instances. Chang et al. [149] presented an improved integer encoding ABC algorithm for handling slow convergence rate and the low search efficiency in solving dynamic WTA. The improved ABC algorithm employs a new initialization strategy using four rule-based heuristic factors, ranking selection and elite guidance. The proposed approach was compared to PSO, DE and EA algorithms on small, medium and large-scale problems in which the number of weapons is larger than, less than or equal to that of targets are tested. It was reported that the proposed algorithm could produce high-quality initial solutions, enhance the convergence rate in solving dynamic WTA problems. Guo et al. [150] presented an ABC algorithm for WTA with multi-to-multi interception and grouping constraints. The problem is a constrained and integer programming problem. Therefore, the search space is mapped by rounding the solutions in each evaluation. The constraints are handled based on penalty function method. To validate the effectiveness of the approach, the combat scenario was generated under 8 missiles and 6 intercepting targets condition. It was concluded that the proposed approach can find the optimal allocation results.

The studies are summarized in Table 14.

4.3. ABC algorithm for other integer programming problems

Akay and Karaboga [19] applied ABC algorithm to integer programming problems. To be able to cope with integer parameters, the ABC algorithm truncates the parameter values to the closest integer when a new solution is generated. Its performance was compared against the PSO algorithms and Branch and Bound technique. The experiments on benchmark functions revealed that the performance of the ABC algorithm is similar to or better

Table 13
List of the studies using ABC variants for the Integer Communication Problems.

No	Year	Study	Application	Representation
1	2010	Wang et al. [140]	PTS for OFDM	Angular
2	2011	Taspinar et al. [141]	PTS for MC-CDMA	Integer
3	2011	Taspinar et al. [142]	PTS for OFDM	Integer
4	2012	Dung et al. [143]	PTS for Coherent Optical OFDM	Angular
5	2013	Yu et al. [144]	PTS for OFDM	Angular
6	2013	Ashrafinia et al. [147]	Symbol Detection in STBC MIMO	Integer
7	2013	Yin et al. [146]	Multiuser Detection in DS-UWB Systems	Integer
8	2018	Cheng et al. [145]	PTS for OFDM	Chaotic Phase Sequence

Table 14
List of the studies using ABC variants for the Weapon Target Assignment Problem.

No	Year	Study	Application	Representation
1	2017	Durgut et al. [148]	WTA	Integer
2	2018	Chang et al. [149]	Dynamic WTA	Integer
3	2019	Guo et al. [150]	WTA	Integer

than can those of PSO algorithms and the Branch and Bound technique for integer programming problems.

Pacurib et al. [151] and Yusiong et al. [152] implemented discrete ABCs for solving sudoku puzzle. Pacurib et al. [151] generates initial digits 1–9 on each square for each bee randomly. A new solution is generated by changing the value in one dimension. Modulo operation is performed when the generated value is greater than 9. When the subgrids violate the sudoku rules, a swap operation is applied. The proposed approach was compared to Hard sudoku and GA-based sudoku solver. It was shown that the approach is able to fine the optimal solution in each run and outperforms GA-solver.

Akay and Kirmizi [153] aimed to optimize wavelet packets by determining the filter type, the level of decomposition and threshold values of wavelet packets. An index array is created for filter types, and the optimum indices corresponding to filter types and the level of decomposition, and the values of continuous variables are truncated to obtain discrete values. The problem is handled as a multi-objective problem which considers two conflicting objectives: compression rate and image quality. It was concluded that ABC algorithm is more efficient and robust for hard cases in which the feasible region is narrow.

Draa and Bouaziz [154] used ABC algorithm for histogram equalization. The proposed approach encodes the gray levels as integers within the range [0,255]. When a solution is generated in the employed and onlooker bee phases, it is converted to an integer and bounded between acceptable gray level range. Performance of the proposed approach was verified on both gray-level and color images and compared to GA and CS algorithm. The results have shown the superiority of ABC algorithm.

Guo and Zhang [155] presented a discrete ABC algorithm with greedy adjustment to solve vehicle routing and location allocation problem as a whole (location and routing problem, LRP)in which the accurate algorithms can only be applied to small scale problems. The aim is to use the existing resources as much as possible and to reduce cost of RL networks which consist of recycle centers and processing centers. The decision variables indicating whether new recycle center can be built at a location, whether a location is chosen, whether a vehicle goes from a location to another location are discrete decision variables. The real numbers are simply transformed into the nearest integers in the initialization and new solution production. Since the solutions produced by the algorithm may not meet up the constraints, they are repaired by shifting customers of overloaded vehicles or rearranging the routes. The fitness function is also adjusted according to greedy adjustment. It was shown that the ABC version with greedy adjustment is better than ABC without adjustment. When compared to PSO with greedy adjustment, ABC with greedy adjustment converges faster.

Arun and Kumar [156] discretized the ABC algorithm to solve the view selection problem. In the approach, the solutions are encoded as integers corresponding to the indices of the nodes of a lattice structure. The total view evaluation cost is used as the cost function to be minimized. N-point random insertion operations are used in new solution generation. In the scout bee phase, global best based 1-point and 2-point random insertions are carried out to exploit the information of the global best solution. The results revealed that the proposed approach outperforms the greedy view selection algorithm according to the quality of Top-K views and their total view evaluation cost values. It was concluded that the proposed approach for materialization of view selection can improve the processing speed of analytical queries.

Adebiyi et al. [157] developed an adaptive dynamic scheduling algorithm based on ABC for vehicular traffic control, which schedules green light timing according to traffic condition to minimize the average waiting time at the cross intersection. In the developed model, each lane is represented by a double digit array/number, where the first digit is 0 for right, and 1 for left direction, and second digit is road identifier. The results showed that the approach minimizes the average waiting time successfully.

Amarjeet et al. [158] proposed a multi-objective ABC algorithm to solve software clustering problem. The proposed algorithm uses quality indicator, Lp-norm-based distances, and two external archives concepts. Each solution is encoded as a vector of decision variables corresponding to the clusters. In the initialization, a random integer value is assigned to the variables. In new solution generation, a random integer is generated by the randomly selected dimension from the population and archives (CA + DA) with 0.5 probabilities. The approach was compared to some existing many-objective optimization algorithms on seven open-source software systems (JFreeChart, JHotDraw, JavaCC, JUnit, Java Servlet API, XML API DOM, and DOM 4J). It was reported that the proposed approach is superior over state-of-the-art algorithms in terms of modularization quality, cohesion, coupling, and inverted generational distance (IGD).

The studies are summarized in Table 15.

5. Mixed integer programming artificial bee colony algorithm

If some variables are integer and some are not then the problem given in Eq. (21) is called a mixed integer programming problem. The case where the integer variables are 0 or 1 is called pure (mixed) 0/1 programming problems or pure (mixed) binary integer programming problems

Table 15
List of the studies using ABC variants for the Other Integer Programming Problems.

No	Year	Study	Application	Representation
1	2009	Akay and Karaboga [19]	Benchmark Problems	Integer
2	2009	Pacurib et al. [151]	Solving Sudoku Puzzle	Integer
3	2010	Yusiong et al. [152]	Solving Sudoku Puzzle	Integer
4	2012	Akay and Kirmizi [153]	Wavelet Packets	Integer
5	2014	Draa and Bouaziz [154]	Histogram Equalization	Integer
6	2017	Guo and Zhang [155]	LRP	Integer
7	2017	Arun and Kumar [156]	View Selection	Integer
8	2017	Adebiyi et al. [157]	Traffic Control	Integer
9	2018	Amarjeet et al. [158]	Software Clustering	Integer

5.1. ABC algorithm for distribution systems

The network reconfiguration problem in a distribution system deals with finding a best configuration of radial network to minimize power loss under certain operating constraints, including voltage profile, current capacity of the feeder and radial structure of the distribution system. The total energy losses of distribution network can be defined by Eq. (23):

$$f_1(X) = \sum_{i=1}^{n_{br}} R_i |I_i|^2 \tag{23}$$

where nbr is the number of branches, R_i and I_i are the resistance and the current magnitude of i th the branch, respectively. The objective function for voltage stability index can be formulated by Eq. (24):

$$f_2(X) = \min\left(\frac{1}{SI(m2)}\right) \tag{24}$$

where the stability index for node $m2$, $SI(m2)$, is described by Eq. (25):

$$SI(m2) = |V(m1)|^4 - 4\left[P(m2)X_{jj} - Q(m2)R_{jj}\right]^2 - 4\left[P(m2)R_{jj} + Q(m2)X_{jj}\right]|V(m1)|^2 \tag{25}$$

$V(m1)$ is voltage of node $m1$, $P(m2)$ and $Q(m2)$ are total real and reactive power load fed through node $m2$, Xjj is reactance of branch jj, and Rjj is resistance of branch jj. The third objective is based on minimization of emission which can be defined by Eq. (26):

$$f_3(X) = \sum_{i=1}^{N_{GT}} E_{GT_i} + \sum_{i=1}^{N_{FC}} E_{FC_i} + \sum_{i=1}^{N_{WT}} E_{WT_i} + E_{Grid} \tag{26}$$

$$E_{GT_i} = \left(CO_2^{GT} + NO_x^{GT} + SO_2^{GT}\right) \times P_{GT_i} \tag{27}$$

$$E_{FC_i} = \left(CO_2^{FC} + NO_x^{FC} + SO_2^{FC}\right) \times P_{FC_i} \tag{28}$$

$$E_{WT_i} = \left(CO_2^{Wind} + NO_x^{Wind} + SO_2^{Wind}\right) \times P_{WT_i} \tag{29}$$

$$E_{Grid} = \left(CO_2^{Grid} + NO_x^{Grid} + SO_2^{Grid}\right) \times P_{Grid} \tag{30}$$

where E and P are design emissions produced and active power generation by the ith energy sources including gas turbine, fuel cell, wind turbine and grid. NGT, NFC and NWT are the numbers of the GT, FC and WT units, respectively.

The fourth objective function that is minimization of cost of electrical energy produced by substation and resources can be expressed by Eq. (31):

$$f_4(X) = \sum_{i=1}^{N_{GT}} C_{GT_i} + \sum_{i=1}^{N_{FC}} C_{FC_i} + \sum_{i=1}^{N_{WT}} C_{WT_i} + C_{Sub} \tag{31}$$

$$C = a + b \tag{32}$$

$$a = \text{Capital cost} \times \text{Capacity} \times \eta \tag{33}$$

$$b = (\text{Fuel cost} \times O\&M\ \text{Cost}) \times P \tag{34}$$

$$C_{sub} = P_{sub} \times \Pr_{sub} \tag{35}$$

Rao et al. [159] applied ABC algorithm for determining the status of switches to minimize the power loss. In the algorithm, each solution corresponds to the open switches in the configuration. They tested the approach on 14, 33, and 119-bus systems and compared to SA and DE algorithms. It was stated that the proposed algorithm is superior in terms of solution quality and CPU time.

Abu-Mouti and El-Hawary [160] used ABC algorithm to minimize the total system power loss in distribution systems, which is a mixed integer nonlinear optimization problem. The proposed algorithm recruits more than one onlooker bee to a high quality source in the onlooker bee phase. The approach was tested on the 33-bus and 69-bus radial distribution feeder systems, and it was seen that the proposed approach can find the optimal solutions in less CPU time and converges with less evaluation number.

Nasiraghdam and Jadid [161] presented a non-dominated sorting ABC algorithm to solve the distribution system reconfiguration and hybrid energy system sizing. The algorithm searches for the optimal integer switch numbers that will be opened in the reconfiguration process and the active power generated by each hybrid system. The total power loss, the total electrical energy cost, and the total emission produced by hybrid energy system and the grid are minimized and the voltage stability index (VSI) of distribution system is maximized by the algorithm. The experiments were performed on 33 bus distribution systems. The proposed approach was compared to Non-dominated Sorting GA-II (NSGA-II) and multi-objective PSO. The results revealed that the proposed approach achieves quality solutions and maintains a better diversity of the Pareto front against the other methods.

Abedinia and Barazandeh [162] presented an interactive ABC algorithm for distribution network problem which considers voltage profile and load demand. In the problem, a set of variable is defined as matrix, and associated element is 1 if a line is joint with another line, otherwise it is 0. In the proposed algorithm, the onlookers consider the Newtonian law of universal gravitation improve exploitation capacity. To verify the efficiency of the algorithm, 4 scenarios of distribution generation penetration and 3 scenarios of load demand growth were used, and it was reported that the proposed approach can enhance the bus voltages in the distribution network upgrading phase.

El-Zonkoly [163] proposed an ABC based approach to optimally schedule of multiple hybrid photovoltaic-diesel distributed generation in distribution systems. The algorithm searches for the optimal values for the rating of the photovoltaic systems and batteries banks and the buses at which hybrid photovoltaic-diesel system are placed. It minimizes the power loss, the amount of imported power from the transmission grid and the unserved load in case of emergency, the investment cost, replacement and operation and maintenance costs of the systems. The performance of the system was verified on the radial 33-bus test system and the Egyptian meshed 45-bus system of Alexandria. The approach was shown to have good performance at minimizing the objective function and satisfying the operational requirements.

Kefayat et al. [164] searched for optimal location and sizing of DSs based on a hybrid configuration of ACO and ABC. The main aim is minimizing the power loss, emission and cost and maximizing the voltage stability index. Since ACO is good at discrete optimization and ABC is powerful in continuous optimization, the problem is decomposed into two parts. In the first part, candidate distributed energy resources are generated by ACO and the size of them is optimized by ABC. The approach was validated on the standard IEEE 33- and 69-bus distribution systems, and its performance was compared to PSO with constriction factor approach (PSO–CFA), ABC and modified Teaching–Learning Based Optimization (MTLBO). It was reported that the proposed ACO-ABC based approach is efficient for distributed energy resources allocation problems.

Das et al. [165] proposed a network expansion planning approach for power systems based on a modified ABC algorithm. In the problem the line numbers per right of way are integer variables while the pv bus generation values and their terminal voltage settings are real variables. The approach minimizes the line investment cost, the generation costs, and reactive power compensation cost. The algorithm uses a new neighbor solution production in the employed bee phase, which modifies all dimensions rather than changing only one dimension as in the basic ABC algorithm. It also uses the information of global best solution. The real values are converted to the nearest integer values. In the onlooker bee phase, the gravitational forces are normalized to eliminate the proportionality constant. The approach was applied to standard IEEE 24 bus system, South Brazilian 46 bus system, 93 bus Colombian system, and Garver 6 bus AC system. The proposed approach is shown to be able to get the lowest investment costs within less computational complexity compared to GA, BF–DEA, ABC, and improved HS and it has fast convergence rate.

Das et al. [166] implemented an ABC-based strategy for optimal placement of distributed energy storage systems in DSs. The algorithm optimizes the decision variables, including the size of energy storage systems with unity power factor and the energy storage system positions in the network to minimize voltage deviation, line loading, and power losses. The approach was validated and compared with PSO on a medium voltage IEEE-33 bus distribution system. It was stated that the proposed strategy can obtain voltage profile improvement, line loading minimization, and power loss reduction, and thereby significantly improve distribution network performance.

The studies are summarized in Table 16.

5.2. ABC algorithm for manufacturing production

The manufacturing production focuses on the acquisition of resources and materials and planning the production activities to convert a raw material into a completed product by satisfying the customer demand in the most cost efficient way and minimizing cost, shortage cost and overtime cost. The objective function can be defined to minimize the total makespan time by determining job sequences by Eq. (36):

$$\min \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} y_{ij} T_{ij} \tag{36}$$

$$\begin{aligned} &\min \sum_{\substack{j=1 \\ j\neq i}}^{n} y_{ij} = 1, j = 1, \dots, n \\ &\min \sum_{\substack{i=1 \\ i\neq j}}^{n} y_{ij} = 1, i = 1, \dots, n \end{aligned} \tag{37}$$

$$y_{ij} = \begin{cases} 1, & \text{if } P_i \text{ is followed by } P_j, i, j = 1, \dots, n \text{ and } i \neq j \\ 0, & \text{otherwise} \end{cases} \tag{38}$$

where n is the number of different part types, P_i is the part type, m is the number of machines in the line, d_i is the number of required parts of type P_i in the demand list, T_{ij} the setup time required to switch the line from processing part type P_i to part type P_j. The line workloads on the kth, W_k can be estimated by Eq. (39):

$$W_k = \sum_{i=1}^{n} x_{ik} (d_i P_i^b + \sum_{\substack{j=1 \\ j\neq i}}^{n} y_{ij} T_{ij}) \tag{39}$$

where P_i^b denotes the process time of part P_i by the bottleneck machine and x_{ij} is given by Eq. (40)

$$x_{ij} = \begin{cases} 1, & \text{if } k\text{th part is a part of } P_i \\ 0, & \text{otherwise} \end{cases} \tag{40}$$

The line workloads on the kth and $(k-1)$th part batches satisfy Eq. (41):

$$|W_k - W_{k-1}| < \varepsilon_k, k = 2, \dots, n \tag{41}$$

Ajorlou et al. [167] implemented an ABC-based model to find optimal sequence of jobs and work in process (WIP) level in a serial constant work in process (CONWIP) production line, which is a mixed integer programming problem. The aim is to determine the WIP level and job sequencing to minimize the overall completion time. In the algorithm, each solution represents each part type to be processed in the production line. Since the solutions are sequence of the jobs, a combinatorial operator is used to generate a new solution using current solution. The algorithm is executed for each WIP level up to the size of the demand list and the job sequencing with the minimum WIP level is taken as the optimal solution. The approach was verified on a single serial CONWIP production line with 3 machines, producing 6 part types. Ajorlou and Shams [168] applied the approach on CONWIP production line with 3 machines, producing 12 part types and line with 12 machines, 21 part types. It was concluded that the proposed approach does not use simplifying assumptions or linearized model of the production system and it is more efficient compared to GA on the problem considered.

The studies are summarized in Table 17.

5.3. ABC algorithm for service composition and optimal selection

Service composition involves dynamic selection and allocation of heterogeneous available services. The basic service selection problem (SSP) consists of a requirement specification, a composite process, and sets of candidate services. A solution to the SSP corresponds to a plan of assigning an appropriate service to each task to optimize the composite Quality of Service (QoS).

Assuming that M is the number of tasks in the composite process, D is the number of QoS attributes, SSP can be expressed

Table 16
List of the studies using ABC variants for the Distribution Planning Problem.

No	Year	Study	Application	Representation
1	2008	Rao et al. [159]	Network Configuration	Mixed
2	2009	Abu-Mouti and El-Hawary [160]	Generation Sizing	Mixed
3	2012	Nasiraghdam and Jadid [161]	Reconfiguration and Sizing	Mixed
4	2013	Abedinia and Barazandeh [162]	Distribution Planning	Mixed
5	2014	El-Zonkoly [163]	Scheduling	Mixed
6	2015	Kefayat et al. [164]	Placement and Sizing	Mixed
7	2017	Das et al. [165]	Expansion Planning	Mixed
8	2018	Das et al. [166]	Optimal Placement	Mixed

Table 17
List of the studies using ABC variants for the Manufacturing Production Problem.

No	Year	Study	Application	Representation
1	2011	Ajorlou et al. [167]	CONWIP Production	Permutation
2	2013	Ajorlou and Shams [168]	CONWIP Production	Permutation

by Eq. (42):

$$\begin{aligned}
&\text{maximize} \sum_{i=1}^{D} w_i f_i \left(\left\{ \sum_{v=1}^{N(u)} p_u^v s_u^v \middle| \, u = 1, 2, \ldots, M \right\} \right) \\
&s.t. f_i \left(\left\{ \sum_{v=1}^{N(u)} p_u^v s_u^v \middle| \, u = 1, 2, \ldots, M \right\} \right) \geq L_i, \\
&\sum_{i=1}^{D} w_i = 1, p_u^v \in \{0, 1\}, \sum_{v=1}^{N(u)} p_u^v = 1 \\
&i = 1, 2, \ldots, D, u = 1, 2, \ldots, M, v = 1, 2, \ldots, N(u)
\end{aligned} \tag{42}$$

where w_i is the weight of attribute i, p_u^v whether or not the vth service is selected for the uth task.

Wang et al. [169] proposed a QoS-aware service selection method based on ABC and greedy search in which new solutions are generated based on the neighboring services of its component services. They also proposed a threshold-based and a distance-based algorithm. The approaches were validated on 10,000 services where each service has four QoS attributes, including response time, reliability, throughput, and price. The methods were compared in terms of optimality, rate of convergence and stability. In terms of optimality and stability, threshold based approach is superior over the other approaches while both threshold based approach and distance based approach have higher time complexity than ABC, and the proposed algorithms can achieve better optimality within limited time.

Xu and Liu [170] defined finding global optimal service selection for concurrent requests to improve users' satisfaction and service broker's into an optimization problem and then proposed a Service-Oriented ABC algorithm based on Similarity and Priori to solve the problem. Food source vector corresponds to the selected services for the kth request. The neighbor solution production is achieved based on crossover and mutation in the employed and onlooker bee phases. A learning operation based on global solution information is used in the scout bee phase. In the experiments, there are 50 concurrent requests, 1000 candidate services and four QoS attributes. Experiment results showed that the proposed approach is effective, and considering service domain characteristics is helpful for enhancing the performance.

Huo et al. [171] presented a discrete global-best guided ABC based approach for cloud service composition which is defined as a nonlinear integer programming problem. Each solution corresponds to an integer array composition solution and each element in the array is generated by rounding-down the real numbers. The global best solution guides the search while exploiting current solution. To verify the validity of the proposed approach, QWS dataset and a random dataset were used and it was compared to GA, PSO and DE algorithms. The results showed that the proposed approach with the time attenuation function provides more consistent and quality service in a short period of time compared to the other algorithms.

Lartigau et al. [172] proposed an approach that considers quality of service (QoS) parameters of cloud services in addition to the physical location of the manufacturing resources, which impose additional constraints. Because the composition is a time consuming exhausted operation, they optimized their approach by using ABC algorithm. The fitness function allows to rank the compositions and the composer agent must ensure that the least number of requirements is satisfied. In the initialization phase, solutions are generated randomly in the space of possible compositions. The initial composition is established based on the solution with the minimum transportation distance. They modified the initialization scheme of ABC algorithm which provides a search around the neighborhood of the shortest transportation path. The proposed ABC-based approach was shown to be faster than PSO and GA.

Xu et al. [173] focused on designing service domain-oriented optimization algorithms with service domain feature and proposed a set of service domain oriented ABC algorithm. Initial solutions are guided by the priori knowledge on service schemes and service set for a task node. In the fitness function, the user QoS satisfaction, service correlation and service domain constraint are considered. The neighborhood search strategy in the priori knowledge on service schemes and the similar service set for tasks scheme determines a search direction and search step and generates a new source to be replaced with the current one. For the general service set, a random search method is adopted to generate a neighbor solution. In the scout bee phase, two methods are proposed. The first new food source generation method works for the service space search strategy while the second one uses service space equilibrium search strategy. The proposed ABC based approach was reported to improve the efficiency and effectiveness of solving service optimization algorithms on two case studies of the concurrent service selection and the service composition problems.

Deepa and Sathiaseelan [174] designed a new method based on ABC with multi-objective constraints and QoS-based Reusable Service Selection algorithm to improve quality of the service composition optimal selection. In the algorithm, each solution corresponds to a feasible service composition solution, and each dimension of the candidate solution must be an integer satisfying the boundary conditions. The approach was validated on benchmark problems according to search time, execution time and reusability constraint and compared to PSO and ACO. It was stated that the proposed approach is able to provide competitive performance in terms of search time and execution time.

Cheng and Ding [175] improved web services composition based on a non-dominated sorting multi-objective ABC algorithm which is improved by a new Boltzman-based selection strategy instead of roulette wheel selection, a new neighborhood search operator and a new scout bee strategy. A fixed-length integer encoding method is used to encode the composite Web service solutions. In the initialization, the initial Web services set is generated by producing QoS attribute values of candidate services randomly. The proposed approach was tested on a dataset of candidate Web services sets and QoS attribute values generated by simulation. It was concluded that the proposed algorithm is able to produce well-distributed solutions that satisfy the requirements of users for the various QoS properties of the service.

Zhou and Yao [176] proposed a hybrid ABC approach for large-scale composited cloud manufacturing service optimal selection problems. The proposed approach uses the probabilistic model of Archimedean copula estimation of distribution algorithm and the chaos operators of global best-guided ABC in onlooker searching strategy. A task is decomposed into N subtasks in N-dimensional space, and each element of a solution corresponds to the index of the selected manufacturing cloud services. Zhou and Yao [177] presented a multi-objective hybrid ABC algorithm for service composition and optimal selection in cloud manufacturing, which employs the Pareto dominance to guide the search. They combined ABC with cuckoo search with Levy flight in the employed bee phase to increase diversity of population. The comprehensive learning strategy is adopted in the onlooker bee phase integrated with an external archive of elite solutions. Zhou and Yao [178] implemented multi-population parallel self-adaptive differential ABC algorithm which uses multiple parallel subpopulations. These subpopulations are evolved by different mutation strategies used in DE, and their sizes are dynamically adjusted. Zhou and Yao [179] proposed context-aware ABC algorithm which considers service domain features during the initialization and heuristic operators in onlooker bees phase. The proposed approach uses competition-based DE operator in the employed bee phase to enhance exploration ability for large-scale problems and an adaptive selection pressure is adopted for the onlookers. Zhou et al. [180] enhanced the ABC algorithm by introducing DE mutation strategies to improve the information exchange. The presented approach adjusts the control parameters for offspring reproduction and uses scalarization to enhance the selection pressure. In the onlooker bee phase, the quality-indicator based fitness assignment approach is utilized to guide the search to promising regions. It also has multiple variable-size subpopulations reproduced with distinct reproduction operators and an external archive to guide the search operators. To validate the performance of the proposed approaches, different-scale composited cloud manufacturing problems were considered. It was shown that the proposed approaches can produce high quality solutions in terms of searching ability, stability in solving cloud manufacturing service selection problems within an acceptable time complexity.

Chandra and Niyogi [181] defined service selection problem as a constrained optimization problem and proposed a modified ABC algorithm, which uses chaotic map function and opposition learning method in the initialization. A new search operator is utilized in the employed bee phase while the onlooker bee phase exploits the knowledge due to the best solution. The solutions violating the constraints are added penalty to guide them to feasible region. The proposed approach was compared to DE, modified GWO, gbest-guided ABC and improved ABC on QWS dataset. It was shown that the proposed approach outperforms the other state-of-the-art approaches in terms of response time, latency, availability, and reliability.

Dahan et al. [182] improved ABC algorithm for solving the service selection problem. The algorithm initializes a food source by assigning first task and jumping until they reach the last task. Then, the algorithm swaps two solutions to improve them. The selection operator considers the distance between the neighboring web services to select a task from the best solution. Performance of the approach was validated on 60 different datasets and compared to threshold-based algorithm and an ABC variant in terms of solution quality and execution time. The proposed approach provides 6% improvement over threshold-based algorithm and 3% over ABC variant in terms of solution quality.

Wang et al. [183] developed an approximate approach for the neighborhood search of ABC to enhance the local search efficiency in discrete space for service selection problem. The proposed approach approximates to the optimal continuity in continuous domain by an improved neighborhood search. They designed three algorithms based on the approach while maintaining the simplicity. The aim is to find optimal service set which satisfies constraints and maximize the fitness function that is the summation of QoS attributes of selected services. Each solution is encoded by a vector of integer values which corresponds to a service selected and initialized by rounding the real values generated randomly. In the employed and onlooker bee phases, current solution is modified by changing a random component service. The algorithms were validated on a DWS dataset which includes 2,500 real-world services with their QoS information measured using commercial benchmark tools and on a dataset including 90,000 services. The algorithms were compared in terms of optimality and computation time metrics. According to the results, the proposed method is able to achieve high accuracy and convergence speed with the advantage of employing less parameter.

The studies are summarized in Table 18.

5.4. ABC algorithm for software testing

Automated software testing aims to find input that drives the software through the branch predicates in the control flow graph of the source file to maximize a fitness function defined based on coverage criteria. The solution encoding depends on the type of the test data such as real, binary, integer, string, symbolic, etc.

Mala and Mohan [184] presented an ABC based approach in which the solutions represent test cases, and the aim is to maximize the coverage by the global best solution. In the study, state coverage, code coverage, branch coverage and path coverage are considered as test adequacy criteria. A heuristic approach is employed during fitness assignment associated with the solution quality. First, the algorithm is executed to find test data for the initial state and then search is repeated for the next state by keeping the previous one as initial search point. The approach was validated on the comparison, triangle code fragments and compared to GA algorithm. It was reported that the proposed approach is more scalable and requires less computation time to cover the entire test cases. Srikanth et al. [185] presented an ABC-based tool to find the optimal paths. The source is converted to a CFG and independent paths, which correspond to test cases, are extracted from the CFG and traversed using the test data discovered by ABC. It was compared to GA and ACO algorithms and was shown to be more effective. Joseph and Radhamani [186] combined PSO and ABC to optimize test suits that maximize coverage found the proposed approach better than ACO, PSO and basic ABC algorithms. Sahin and Akay [16] implemented ABC, PSO, DE and FF metaheuristics to explore test data in order to maximize a coverage metric. Various fitness functions, including path-based, dissimilarity-based and approximation level + branch distance, were investigated to observe the behavior of the algorithms in the

Table 18
List of the studies using ABC variants for the Service Composition and Optimal Selection Problem.

No	Year	Study	Application	Representation
1	2013	Wang et al. [169]	Service Selection	Integer
2	2014	Xu and Liu [170]	Service Selection	Integer
3	2015	Huo et al. [171]	Service Composition	Integer
3	2015	Lartigau et al. [172]	Cloud Service Composition	Mixed
4	2017	Xu et al. [173]	Service Selection and Composition	N/A
5	2017	Deepa and Sathiaseelan [174]	Service Selection	Integer
6	2017	Cheng and Ding [175]	Web Service Composition	Integer
7	2017	Zhou and Yao [176]	Cloud Manufacturing Service Composition	Integer
8	2017	Zhou and Yao [177]	Cloud Manufacturing Service Composition	Integer
9	2017	Zhou and Yao [178]	Cloud Manufacturing Service Composition	Integer
10	2017	Zhou and Yao [179]	Cloud Manufacturing Service Composition	Integer
11	2018	Zhou et al. [180]	Cloud Manufacturing Service Composition	Integer
12	2019	Chandra and Niyogi [181]	Service Selection	Integer
13	2019	Dahan et al. [182]	Service Selection	Permutation
14	2019	Wang et al. [183]	Service Selection	Integer

search space. The approaches and fitness functions were analyzed on the triangle classifier, quadratic equation, even–odd, largest number, remainder, leapyear, division of mark problems. Parameter sensitivity of the algorithms was also studied for finding suitable values. Results of success rates and runtime analysis showed that metaheuristics are efficient even when the search space is large and approximation level + branch distance-based fitness function is able to guide the search accurately.

Bansal et al. [187] proposed an ABC algorithm integrated with greedy approach (Covering Array Generator) to generate covering an array and a mixed covering array for pair-wise testing. In the approach, each food source represented by integer value corresponds to a covering array, and the fitness is measured by the total number of distinct 2-way interactions covered by the solution. The real values generated are rounded to the nearest integer. The global best solution is exploited during local search performed in the onlooker bee phase which replaces the worst solutions with smart cases. The approach was tested on a Traffic collision avoidance system benchmark and a case study of various features of a printer and compared to some state-of-the-art algorithms. It was reported that the proposed approach is better or comparable against them. Alsewari et al. [188] implemented an ABC based combinatorial testing approach to reduce the redundant test cases. Assuming D is the number of system configuration parameters, each solution test case is as a D-dimensional vector. This vector is optimized by the phases of ABC algorithm. Alazzawi et al. [189] developed an ABC algorithm for pairwise testing to reduce computation time which rises exponentially by an increase in the number of parameters. In the initial stage, a system configuration with ten V-valued parameters are selected and in the second stage, a number of system configurations are used to compare the performance of the proposed approach alongside other strategies. It was shown that the proposed approach is competitive for pairwise testing reduction. Alazzawi et al. [190] applied ABC to for both a uniform and variable strength test suite to reduce the number of input combinations in t-way combinatorial testing. Input analysis, interaction generation and optimization steps are performed and a supporting variable strength is introduced in the approach. The number of interaction element combination is considered as the fitness function. The proposed approach has the ability of generating a test suite with an interaction strength of 6. Alazzawi et al. [191,192,193] hybridized ABC and PSO algorithms for t-way testing to provide a high-interaction strength combinatorial test suite up to the strength of parameter's interaction is 6. Ali et al. [194] modified ABC algorithm as t-way strategy which supports uniform strength t-way testing. The random solution production in the scout bee is replaced with an operator based on flipping to avoid visiting an already discovered food source. It was reported that the proposed approach produces the best result compared to other strategies and improves the results when more input values are configured.

The studies are summarized in Table 19.

5.5. ABC algorithm for structural optimization

Most of the structural engineering design problems are nonlinear mixed integer problems that involve continuous, binary, discrete and integer variables. Assuming that $S = \{x | g_z(x) \leq 0$ or $g_z(x) = 0, z = 1, 2, \ldots, p+q, l_i \leq x_i \leq u_i\}$ is the set of feasible solutions and considering the mixed variables, the problem can be formulated by Eq. (43):

$$\begin{array}{lll} \textit{Minimize} & f(x) & \\ s.t. & h_k(x) = 0; & k = 1, 2, \ldots, p \\ & g_j(x) \leq 0; & j = 1, 2, \ldots, q \\ & l_i \leq x_i \leq u_i; & i = 1, 2, \ldots, n \end{array} \tag{43}$$

where $x = [x_1, x_2, \ldots, x_n]^T$ denotes the decision vector, f is the cost function, l_i and u_i are the lower and upper bound values of the ith variable respectively, p and q are the number of equality and inequality constraints, respectively. g_z be the set of equality and inequality constraints [195].

Hadidi et al. [196] applied the ABC algorithm for structural optimization of planar and space trusses under stress, displacement and buckling constraints. In the study, more than one dimension is changed during local search around a food source and tournament selection operator is employed instead of the roulette wheel selection. In the scout bee phase, Gaussian mutation operator is used rather than uniform initialization. Ten-bar, seventeen-bar, one hundred twenty-bar, forty five-bar truss structure problems are solved by basic ABC and the proposed approach. The proposed approach was stated to be more convergent and efficient compared to the basic ABC algorithm.

Omkar et al. [197] proposed a multi-objective design optimization of composites based on a modified Vector Evaluated ABC algorithm which optimizes discrete variables. The aim is to minimize multiple objectives, including the weight and total cost of a composite to satisfy a specific strength. The primary design variables are number of layers, lamina thickness and the stacking sequence. The plies at different fiber orientation angles and the lamina thickness can take integer values. The efficiency of the approach was verified on a number of different in-plane loading configurations from different regions of the failure envelope, and it was compared to those of PSO, Artificial Immune System (AIS) and GA algorithms. It was concluded that the approach is robust end efficient in composite design that results in significant weight savings.

Table 19
List of the studies using ABC variants for the Software Testing Problem.

No	Year	Study	Application	Representation
1	2009	Mala and Mohan [184]	Test Data Generation	Mixed
2	2011	Srikanth et al. [185]	Test Data Generation	Mixed
3	2011	Joseph and Radhamani [186]	Test Data Generation	Mixed
4	2016	Sahin and Akay [16]	Test Data Generation	Mixed
5	2016	Bansal et al. [187]	Pairwise Testing	Integer
6	2017	Alazzawi et al. [189]	Pairwise Testing	Integer
7	2017	Alsewari et al. [188]	Combinatorial Testing	N/A
8	2019	Alazzawi et al. [190]	t-way testing	Integer
9	2019	Alazzawi et al. [191]	t-way testing	N/A
10	2019	Alazzawi et al. [192]	t-way testing	N/A
11	2020	Alazzawi et al. [193]	t-way testing	Integer
12	2020	Ali et al. [194]	t-way testing	Binary

Akay and Karaboga [198] extended the ABC algorithm by introducing constraint handling technique for constrained design problems, including Welded Beam, Pressure Vessel, Tension/Compression Spring, Speed Reducer, Gear Train. Some of the design problems considered in the paper are mixed integer programming problems. The proposed approach truncates the real values to the closest integer value. It was compared to society and civilization algorithm, PSO, $(\mu+\lambda)$-ES and unified PSO with mutation (UPSOm) algorithms and shown to be a promising tool for optimizing constrained engineering design problems.

Aydogdu et al. [199] presented an ABC based approach for optimum design of steel space frames. The sequence numbers of W steel section listed in steel profile table are discrete design variables. The approach was applied to two steel space frames with four story 132 member steel space frame and eight story 1024 member steel space frame. The proposed approach was compared to the dynamic harmony search and ant colony optimization algorithms and it was found that the weights of optimum frames produced by the proposed approach have lighter weight than the dynamic harmony search and ant colony optimization algorithms.

Garg [195] implemented a penalty guided ABC algorithm for the structural design optimization problems, including design of pressure vessel, welded beam, and tension/compression string. It was shown that the best solutions obtained by ABC are all near-global solutions in each problem tested, and the proposed approach is more robust in terms of standard deviation.

Sevim et al. [200] aimed to minimize the weight of structures subjected to some design requirements by ABC for discrete optimization of planar truss structure. The results showed that the prosed algorithm is robust and can be used effectively for solving such problems.

Carbas and Saka [201],Carbas et al. [202] developed an ABC algorithm for the optimum geometric dimensions of cold-formed thin-walled open steel sections under various external loading. Since the displacement and stress constraints are introduced in the problem definition, the problem becomes a mixed integer and discrete programming problem. The design variables include the sequence number of the geometric dimensions of thin plates. The continuous values are discretized by rounding process. A design example was employed to verify the efficiency of the proposed approach. It was reported that the proposed approach is an efficient and robust in optimizing shape design of cold-formed thin-walled open steel sections under various external loading, where the geometric nonlinearity and effect of warping is taken under consideration.

Yancang et al. [203] improved the initialization scheme of ABC so that the initial solutions can evenly distributed in the search domain. The tabu idea was integrated to ABC in the selection step by introducing a penalty parameter which is compared with the fitness of a solution. The proposed approach was applied to truss structure optimization. The experimental results revealed that the

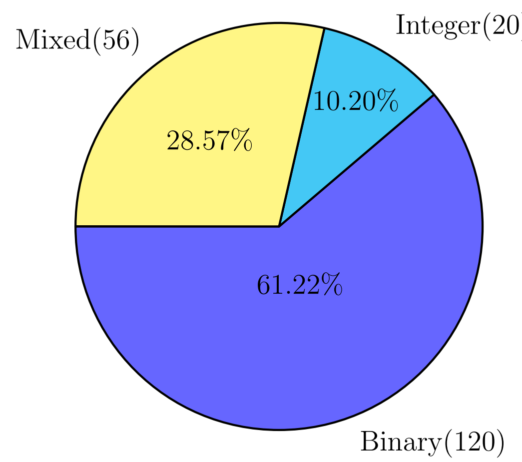

Fig. 2. Distribution of the studies by problem type.

proposed modification improves the local search ability of the algorithm.

Dong et al. [204] extended the ABC algorithm by combining with Levy flight and differential self-perturbation to expand the search space and enhance the exploitation ability. A chaotic mechanism is incorporated to the initialization phase and adaptive selection is applied in the scout bee phase to increase diversification ability. The approach was validated on the cantilever beam, gear train and three bar truss design problems which have some discrete variables. The proposed algorithm was compared to basic ABC and some state-of-the-art algorithms. These results demonstrated that the proposed algorithm can effectively solve discrete design problems with low computational budget.

The studies are summarized in Table 20.

5.6. ABC algorithm for other mixed integer programming problems

Yeh and Hsieh [205] presented a penalty-based ABC algorithm to solve the reliability redundancy allocation problem (RRAP) under the constraints, cost, weight and volume. Violated constraints are added to the cost function as penalty term. Four mixed integer programming RRAPs were used to verify the efficiency of the approach. Compared to the traditional heuristics, ABC has the advantage of memory, multi-character, local search and solution improvement mechanisms, and it can achieve the global solution or near-global solution.

Szeto and Jiang [206],Jiang et al. [207],Szeto and Jiang [208] enhanced ABC for solving the problem of bus network design in Tin Shui Wai, Hong Kong, which minimizes the weighted sum of the number of transfers and the total travel time of the users. The proposed approach employs a new representation to encode routes, and a heuristic assigns frequency to the route during the

Table 20
List of the studies using ABC variants for the Structural Optimization Problem.

No	Year	Study	Application	Representation
1	2010	Hadidi et al. [196]	Truss Bar Structure	Mixed
2	2011	Omkar et al. [197]	Composite Design	Mixed
3	2012	Akay and Karaboga [198]	Engineering Design Problems	Mixed
4	2012	Aydogdu et al. [199]	Steel Space Frames	Mixed
5	2014	Garg [195]	Engineering Design Problems	Mixed
6	2015	Sevim et al. [200]	Truss Bar Structure	Mixed
7	2015	Carbas and Saka [201]	Cold-Formed Thin-Walled Open Steel Sections	Mixed
8	2016	Carbas et al. [202]	Cold-formed thin-walled sections	Mixed
9	2017	Yancang et al. [203]	Truss Bar	
10	2019	Dong et al. [204]	Engineering Design Problems	Mixed

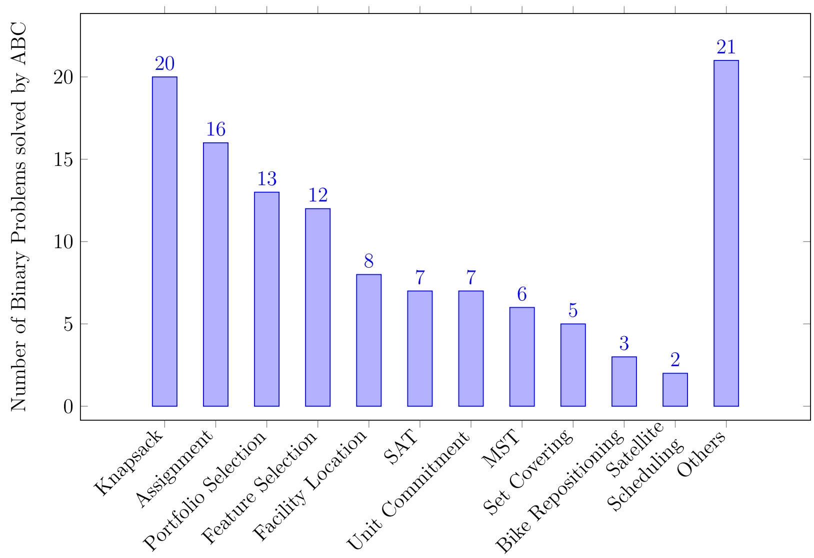

Fig. 3. The number of Binary Problems solved by ABC.

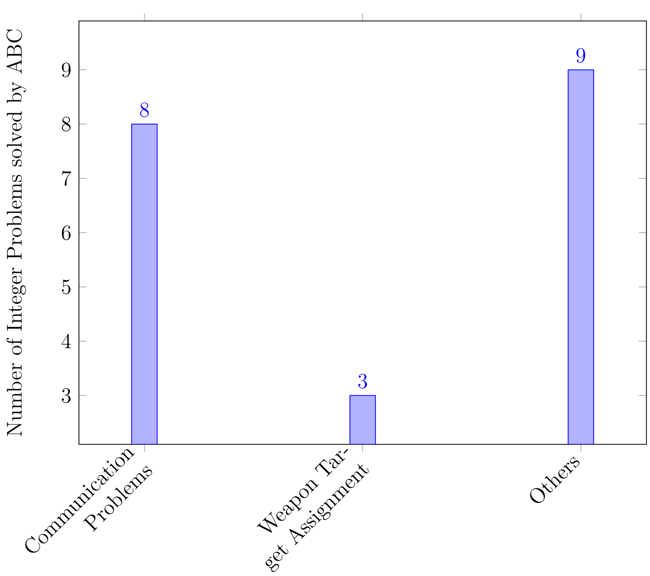

Fig. 4. The number of Integer Problems solved by ABC.

fitness evaluation. The insert neighborhood operator inserts a randomly selected stop into a randomly selected route in the solution. The remove neighborhood operator randomly removes a stop from a route in the solution. The swap neighborhood operator exchanges two nodes of the same type (in terms of terminals, stops, and destinations) between two different routes of the solution. The transfer neighborhood operator randomly moves an intermediate stop in one route to another route in the solution. For all the neighborhood operators, a checking mechanism is adopted so that a node does not appear on the same route more than once. Moreover, a node covering operation is carried out to ensure that a solution is feasible. After feasible neighbor solutions are generated, a stop-sequence improvement heuristic is used to improve the sequence of intermediate stops of each route. The proposed approach can generate better design in terms of maximum intermediate stops, total travel time, number of transfers, maximum headway, and total fuel cost. It can reduce total travel time by 4.98%, the number of transfers by 8.15%, the maximum number of intermediate stops by five, the maximum headway by 2.4%, and the total fuel cost by 5.5%, respectively.

Ekhtiari and Poursafary [209] used ABC algorithm to select efficient vendors for the companies, which can consider three aspects of multiple criteria, random factors, and reaching efficient solutions with the objective of improvement leading to the importance. The problem is converted to a mixed integer nonlinear single objective deterministic problem. The proposed approach was validated on a real problem associated with a home appliances manufacturer in Iran. The experimental results showed that ABC can find more efficient solutions to the large scale problem while PSO spends less time to solve it. Garg et al. [210] presented a two-stage approach based on ABC algorithm and an improvement heuristic to solve RRAP. The constraints are handled by the penalty approach. On four mixed integer programming RRAPs, the proposed approach can produce better results compared to the available methods in the literature.

Hu et al. [211] proposed a mixed integer linear programming formulation and an exact algorithm to solve a parallel machine scheduling problem in plastic production which involves uncertain processing time, job release time, and setup time for mold. An ABC algorithm was also presented to solve large-scale problems. Extensive computational experiments showed that the proposed algorithm surpasses the exact method in terms of objective value and computational time. Li et al. [212] proposed a mixed ABC approach to deal with the continuous prices and discrete allocations simultaneously, embedded with a transferred memory scheme to achieve the flexible and smooth tariff design with dynamic demand. The approach handles both discrete variables in discrete commitment scheduling and continuous variables in continuous economic dispatch problem into a unified chromosome structure. The approach was tested on a real-world thermal electricity company of northeast China. According to the results, the transferred memory scheme has merit in the improvement. Ozcan and Simsir [213] developed a replacement scheduling model for a rail production line to reduce the downtime to the lowest possible levels. Flexible Time-Based Replacement method was integrated with ABC algorithm. Taguchi Design of Experiment (DOE) method was used to find suitable values for the control parameters in ABC algorithm. The proposed approach gained a decrease of approximately 12% with an improvement of 246 h in downtime in addition to having advantage of being easy to apply.

The studies are summarized in Table 21.

6. Summary and discussion

As mentioned before, optimization problems can be classified into two main groups: continuous and discrete problems. The group of discrete optimization problems includes the problems with decision variables taking discrete numerical values such as binary and integer, and combinatorial type problems. Many engineering and design problems can be formulated as binary, integer or mixed integer optimization problems. The ABC algorithm has been often applied to solve these types of problems in various fields. To be able to handle with these optimization problems with discrete variables, researchers have proposed several modifications related to the representation, local search and selection operators of the ABC algorithm since it has been initially proposed for solving problems defined in continuous space. We have reviewed these studies and grouped them according to the problem type addressed. Then, by investigating how ABC algorithm was modified to be able to cope with the numeric variables in the discrete domain, the solution representation, cost and constraint functions, selection operators to guide the algorithm in the search space are reported. The main conclusions drawn from the reviewing process of the studies are given below:

- As stated above, among the papers related to discrete optimization by ABC, we only included binary, integer and mixed integer programming problems and excluded combinatorial type problems because they use different search characteristics and operators. The number of studies using ABC algorithm for these type problems is 196. As seen from Fig. 2, the number of papers is 120, 20 and 56 for binary, integer and mixed integer programming problems, respectively. From Fig. 3, it is clear that assignment, facility location, feature selection, knapsack, minimum spanning tree, portfolio selection, satisfiability, set covering, unit commitment, bike repositioning, satellite scheduling are popular binary problems, and knapsack and assignment problems are the most studied problems by using ABC. As seen from Fig. 4, weapon target assignment, communication problems and some other integer problems have been solved by ABC algorithm. Moreover, it has been also employed to determine phase vectors in various communication problems, including OFDM and CDMA systems. Fig. 5 shows that distribution systems, manufacturing production, service composition and selection, software testing, structured optimization are mixed integer programming problems solved by ABC algorithm. As seen from Fig. 6, in each year, approximately twenty papers have been published on ABC applications to numerical discrete problems.
- Representation and Operators: When a binary optimization problem having binary decision variables is handled, binary, continuous, quantum, integer, permutation, edge-set and fuzzy encoded solution representations have been used in the studies related with ABC. In some studies, continuous representation of original ABC algorithm has been retained and solutions have been mapped to binary space by using binary tournament, rounding, thresholding, angle modulation, Bernoulli process, surjection mapping, sigmoid function, flooring, modulus operation and random-key mapping process. In this case, arithmetic operations of basic ABC or some other arithmetic perturbation operators have been applied to generate new solutions in the phases of ABC. When binary encoding is used, basic logical operators (AND, OR, NOT, XOR), bit flipping mutation, exchange, one-point/two point crossover, shift/double shift, ejection-chain operator, dissimilarity measure based operators, replace/alter, swap/move, walk around search, purifying search operators and some problem-specific heuristics have been utilized to construct a new solution by ABC. When quantum encoding is used, the solution vector has been represented by q-bits corresponding to superposition of states, and dynamic mutation/multi-granularity rotation, rotation-based evolution, rotation gate and bit exchange operators have been used in the local search step of ABC. When integer encoding is preferred, the real values generated have been truncated or rounded to the closest integers. Position change, flipping, swap, modulus and N-random insertion operators have been employed to generate a new solution in the studies reviewed. When permutation encoding is used to represent solutions in ABC, combinatorial operators including swapping, insertion, partial message crossover and position based crossover operators have been employed to construct new solutions. Also, in a study, a neighborhood operator has been randomly chosen from a pool of operators including random swaps, subsequence reverse, and random swaps of reversed subsequence. To generate a new solution, knowledge-directed deletion/insertion operators are integrated to the phases of ABC.
- Control Parameter Tuning: To determine suitable values for the control parameters of ABC, some studies have used Taguchi Design of Experiment method while some have adjusted the control parameters for offspring reproduction, subpopulation sizes, and the selection pressure.
- Initialization Schemes: In some studies, ABC has initialized the solutions to distribute them evenly or generated solutions in promising areas of the search space. In the initialization, some approaches have accepted only different solutions to the population to increase diversity. For constrained problems, feasible initialization or feasibility preserving schemes have been incorporated to the algorithm. Rule-based heuristic factors, the hyperbolic tangent function, chaotic mechanisms and chaotic maps based on logistic equation have also been incorporated into the ABC algorithm. Especially for solving graph problems such as minimum spanning tree problem, edge-set encoding has been adopted in ABC and Prim's-like iterative process is executed in the initialization phase of these studies to avoid pure random initialization.

Table 21
List of the studies using ABC variants for the Other Mixed Integer Programming Problems.

No	Year	Study	Application	Representation
1	2011	Yeh and Hsieh [205]	RRAP	Mixed
2	2012	Szeto and Jiang [206]	Bus Network Design	Mixed
3	2013	Ekhtiari and Poursafary [209]	Vendor Selection	Mixed
4	2013	Garg et al. [210]	RRAP	Mixed
5	2013	Jiang et al. [207]	Bus Network Design	Mixed
6	2014	Szeto and Jiang [208]	Bus Network Design	Mixed
7	2016	Hu et al. [211]	Parallel Machine Scheduling	Mixed
8	2019	Li et al. [212]	Electricity Tariff design	Mixed
9	2019	Ozcan and Simsir [213]	Replacement Scheduling	Mixed

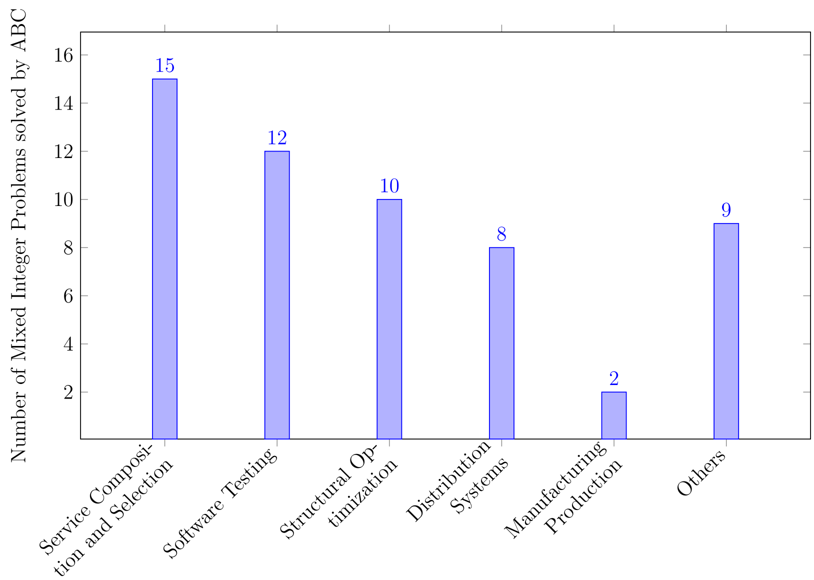

Fig. 5. The number of Mixed Integer Problems solved by ABC.

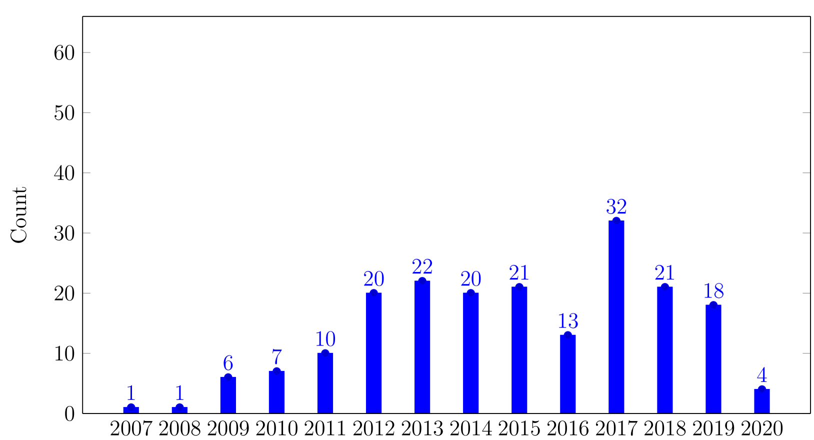

Fig. 6. Change of the number of studies by year.

- Selection Operators: Greedy selection or roulette wheel selection employed in the basic ABC has been replaced with other selection schemes, such as feasibility-based selection strategy, binary tournament selection and ranking selection schemes to guide the search through more promising regions.
- Constraint Handling: For solving problems with constraints, ABC variants have used feasible initialization, Greedy Randomized Adaptive Search Heuristic, rule-based heuristic factors and priori knowledge-based construction in the initialization phase. After producing a new solution, a mechanism such as two-phase heuristics, greedy optimization algorithm, penalization, a column state shifting method of matrix, some problem-specific repairing operators (shifting, swapping, rearranging), two-element variation technique, feasibility enforcement and infeasibility toleration procedures are integrated into ABC to preserve the feasibility of solutions.
- Diversity Preserving: To preserve the diversity in the population during search or bring diversity when it is diminished, some strategies have been used in the studies. In the initialization, only different solutions have been accepted in the population or a chaotic method has been used to generate

new sources. In new solution production, bit mutation operator, multi-dimensional local search operators based on rough set dissimilarity or Levy flight have been used in several studies to increase diversity in the population.

- Modifications Related to Problem Types: When the problem types are considered, binary logic operators and binary tournament operators have been mostly preferred, and the solutions have been repaired by two-phase heuristics to solve knapsack problems. For assignment problems, the studies have given more emphasis to initialization scheme such as adaptive search heuristic or chaotic mapping. Swap, insert, ejection-chain and replace-alter operations have been studied for assignment problems. For constraint satisfaction problems, local search enhancement operations such as greedy local search procedure or integrating with tabu search have been used to enhance convergence ability for the instances. For facility location problems, binary representation and swap, flip, logic operators have been applied successfully. In feature selection applications, global search ability has been increased to boost the performance of the algorithm by combining with decision tree and global best information. Initializing the algorithm with Prim's like iterative process or chaotic logistic mapping enhances the success for MST problems. For unit commitment problems, problem-specific reproduction operators and memory features have been integrated into the algorithm, and the infeasible solutions have been repaired when the solutions generated do not satisfy the problems constraints. For integer problems, continuous representation has been generally used, and the solutions in the continuous domain have been mapped to the integer search space by truncations or angular modulations.
- Hybridized Schemes of ABC: To enhance the convergence ability of ABC and improve the solution quality, global-best-information has been exploited, or elitist selection strategy has been integrated into the algorithm. In order to combine the advantages of different algorithms in exploration or exploitation, ABC algorithm has been hybridized with boosting decision tree, FA, GA, DE, HS, ACO, SA and TS algorithms. Integrating with a memory, Levy flight and differential self-perturbation and multiple variable-size subpopulations reproduced with distinct reproduction operators have been proposed to increase diversity and improve the performance of ABC algorithm in discrete space. In some studies, ABC and PSO algorithms have been hybridized to overcome getting trapped to local minima in PSO to solve warehouse allocation problem and t-way testing, test suite generation. ACO, PSO and ABC algorithms have been combined together to increase diversity in the population for terminal assignment problem. To increase local search ability of ABC, it has been incorporated with TS for satisfaction problems. ABC algorithm has been hybridized with ACO for feature selection purpose to avoid the getting stuck during ACO search and reduce the cost of global search in ABC algorithm. ACO has been used to determine the best solution and to produce new solutions in the scout phase, and ABC performs search on these best solutions. In feature selection, ABC has been integrated with DE to improve ABC's convergence rate by using DE operations in the neighborhood production. ABC-DE hybridization has been also used to solve PO problems. After generating the harmonies in HS, the ABC algorithm is applied followed by the steps of HS to obtain efficient frontiers in PO problems. A hybrid of the continuous ACO, ABC and GA algorithms has been obtained by employing the elitism strategy of GA and modification rate of ABC algorithm in the continuous ACO for cardinality-constrained PO problem. The ABC algorithm which replaced the scout bee phase with a guided onlooker bee phase has been merged by GA operators. In mixed integer programming problems, ABC has been used to handle with continuous variables while ACO has been used to handle with discrete variables since ACO is good at discrete optimization and ABC is powerful in continuous optimization. meanFA intensification has been incorporated into ABC algorithm to improve convergence rate to solve cardinality-constrained PO.
- Parallel Processing and Cooperative Search in Discrete ABC: A differential ABC algorithm, which employs self-adaptive parallel multi-populations, has been proposed to solve large-scale service composition and optimal selection problems in cloud manufacturing. Each subpopulation is evolved by DE strategies to overcome the slow convergence drawback due to changing single dimension of decision vector in the basic ABC. Each subpopulation uses different mutation strategy to increase the population diversity and enhance the robustness. In this approach, the sizes of subpopulations are adjusted dynamically based on the fitness improvement and the function evaluations consumed. Pareto dominance, niche preservation and decomposition-based approaches guide the elite-preserving process in a hierarchical manner.

Although most of these modifications improve the performance of the algorithm, they employ some problem-specific heuristics or local search operators, which make them inapplicable to other types of problems or produce inefficient results. However, they should sustain their good performance for general problem types. Although binary representation is easy to implement, it may not guide the search through the promising solutions and may cause to slow down the convergence because it contains less information. Mapping in selected encoding scheme and the feasibility preserving mechanism may increase the computation cost, and finding the best control parameter configuration might be labor-intensive. The algorithm performance is also affected by the constraint handling method and penalty coefficients when a constrained problem is handled. Some future directions to overcome these drawbacks of the modifications can be listed as below:

- Developing efficient representation and search operators suitable for discrete problems to decrease the computation cost due to the mapping between spaces
- Implementing new cost efficient selection operators which recruit the bees to the solutions in a more diversified way as in 2-ary or 3-ary selection operators
- Implementing operators exploiting the information in bit strings efficiently
- Describing new constraint handling mechanisms working also with infeasible solutions and not requiring feasibility repairing mechanisms or efficient operators ensuring the feasibility of the solutions
- Automatic tuning of the control parameters of ABC algorithm in discrete domain
- Introducing problem-independent construction heuristics to produce generally-acceptable performance for a variety of problems
- Applying the ABC algorithm to the discrete optimization problems that have not been addressed yet such as frequency planning in cellular mobile networks, cash flow matching, waste management, cutting stock problems.

7. Conclusion

This paper reviews the studies including discrete optimization problems solved by the ABC algorithm and presents the problem types, solution encodings, search operators used in the discrete domain. Since the ABC algorithm was initially proposed for solving the problems defined in the continuous domain, modifications on the algorithm to be able to handle discrete optimization problems have been reported. We discussed the advantages of the variants and pointed some drawbacks in the reviewed studies. Moreover, we offered some future directions to overcome these drawbacks. It is concluded that ABC algorithm framework is efficient for solving these problem types with suitable representation and search operators compatible with the representation. We hope this paper will be very useful for researchers working with binary, integer and mixed integer type discrete problems and the ABC algorithm.

CRediT authorship contribution statement

Bahriye Akay: Conceptualization, Methodology, Literature review, Systematic categorization, Visualization, Writing - original draft. **Dervis Karaboga:** Conceptualization, Methodology, Literature review, Systematic categorization, Visualization, Writing - original draft. **Beyza Gorkemli:** Literature review, Systematic categorization, Visualization, Writing - reviewing and editing. **Ebubekir Kaya:** Literature review, Systematic categorization, Visualization, Writing - reviewing and editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: 1995 IEEE International Conference on Neural Networks, vol. 4, 1995, pp. 1942–1948.
[2] M. Dorigo, Optimization, Learning and Natural Algorithms (Ph.D. thesis), Politecnico di Milano, 1992.
[3] D. Karaboga, An Idea Based On Honey Bee Swarm For Numerical Optimization, Tech. Rep. TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
[4] X.-S. Yang, Nature-Inspired Metaheuristic Algorithms, Luniver Press, 2010.
[5] K.M. Passino, Biomimicry of bacterial foraging for distributed optimization and control, IEEE Control Syst. Mag. 22 (3) (2002) 52–67.
[6] X.-S. Yang, S. Deb, Cuckoo search via Lévy flights, in: 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), IEEE, 2009, pp. 210–214.
[7] K.N. Krishnanand, D. Ghose, Detection of multiple source locations using a glowworm metaphor with applications to collective robotics, in: Proceedings 2005 IEEE Swarm Intelligence Symposium, 2005, SIS 2005, IEEE, 2005, pp. 84–91.
[8] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61.
[9] D. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, M. Zaidi, The Bees Algorithm, Technical Note, Manufacturing Engineering Centre, Cardiff University, UK, 2005.
[10] X.-S. Yang, A new metaheuristic bat-inspired algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Springer, 2010, pp. 65–74.
[11] A. Afshar, O.B. Haddad, M.A. Mariño, B.J. Adams, Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation, J. Franklin Inst. B 344 (5) (2007) 452–462.
[12] D. Karaboga, B. Akay, A survey: algorithms simulating bee swarm intelligence, Artif. Intell. Rev. 31 (1–4) (2009) 61.
[13] D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, A comprehensive survey: artificial bee colony (ABC) algorithm and applications, Artif. Intell. Rev. 42 (1) (2014) 21–57.
[14] B. Akay, D. Karaboga, A survey on the applications of artificial bee colony in signal, image, and video processing, Signal. Imag. Video Process. 9 (4) (2015) 967–990.
[15] M.K. Apalak, D. Karaboga, B. Akay, The artificial bee colony algorithm in layer optimization for the maximum fundamental frequency of symmetrical laminated composite plates, Eng. Optim. 46 (3) (2014) 420–437.
[16] O. Sahin, B. Akay, Comparisons of metaheuristic algorithms and fitness functions on software test data generation, Appl. Soft Comput. 49 (2016) 1202–1214.
[17] B.K. Dedeturk, B. Akay, Spam filtering using a logistic regression model trained by an artificial bee colony algorithm, Appl. Soft Comput. (2020) 106229.
[18] Y. Boudouaoui, H. Habbi, C. Ozturk, D. Karaboga, Solving differential equations with artificial bee colony programming, Soft Comput. 24 (23) (2020) 17991–18007, http://dx.doi.org/10.1007/s00500-020-05051-y.
[19] B. Akay, D. Karaboga, Solving integer programming problems by using artificial bee colony algorithm, in: Congress of the Italian Association for Artificial Intelligence, Springer, 2009, pp. 355–364.
[20] S. Pulikanti, A. Singh, An artificial bee colony algorithm for the quadratic knapsack problem, in: International Conference on Neural Information Processing, Springer, 2009, pp. 196–205.
[21] S. Sundar, A. Singh, A. Rossi, An artificial bee colony algorithm for the 0–1 multidimensional knapsack problem, in: International Conference on Contemporary Computing, Springer, 2010, pp. 141–151.
[22] S. Sundar, A. Singh, A swarm intelligence approach to the quadratic multiple knapsack problem, in: International Conference on Neural Information Processing, Springer, 2010, pp. 626–633.
[23] S. Sabet, F. Farokhi, M. Shokouhifar, A novel artificial bee colony algorithm for the knapsack problem, in: 2012 International Symposium on Innovations in Intelligent Systems and Applications, IEEE, 2012, pp. 1–5.
[24] S. Sabet, M. Shokouhifar, F. Farokhi, A discrete artificial bee colony for multiple knapsack problem, Int. J. Reason.-Based Intell. Syst. 5 (2) (2013) 88–95.
[25] L. Wei, N. Ben, C. Hanning, Binary artificial bee colony algorithm for solving 0-1 knapsack problem, Adv. Inf. Sci. Serv. Sci. (AISS) 4 (22) (2012) 464–470.
[26] T. Liu, L. Zhang, J. Zhang, Study of binary artificial bee colony algorithm based on particle swarm optimization, J. Comput. Inf. Syst. 9 (16) (2013) 6459–6466.
[27] Z. Liu, Y.A. Hu, Quantum artificial bee colony algorithm for knapsack problem, in: Advanced Materials Research, vol. 605, Trans Tech Publ, 2013, pp. 1722–1728.
[28] J. Ji, H. Wei, C. Liu, B. Yin, Artificial bee colony algorithm merged with pheromone communication mechanism for the 0-1 multidimensional knapsack problem, Math. Probl. Eng. 2013 (2013).
[29] K. Manochehri, A. Alizadegan, A hybrid quantum-inspired artificial bee colony algorithm for combinatorial optimization problem: 0-1 KNAPSACK, J. Curr. Res. Sci. 2 (6) (2014) 781.
[30] K. Manochehri, A. Alizadegan, Designing and comparing classic versus quantum artificial bee colony algorithm, J. Math. Comput. Sci. 14 (2015) 183–192.
[31] C. Ozturk, E. Hancer, D. Karaboga, A novel binary artificial bee colony algorithm based on genetic operators, Inform. Sci. 297 (2015) 154–170.
[32] F.J. Vasko, Y. Lu, K. Zyma, An empirical study of population-based metaheuristics for the multiple-choice multidimensional knapsack problem, Int. J. Metaheuristics 5 (3–4) (2016) 193–225.
[33] F. Barani, H. Nezamabadi-pour, BQIABC: A new quantum-inspired artificial bee colony algorithm for binary optimization problems, J. AI Data Min. 6 (1) (2018) 133–143.
[34] J. Cao, B. Yin, X. Lu, Y. Kang, X. Chen, A modified artificial bee colony approach for the 0-1 knapsack problem, Appl. Intell. 48 (6) (2018) 1582–1595, http://dx.doi.org/10.1007/s10489-017-1025-x.
[35] E. Hançer, An abc algorithm inspired by boolean operators for knapsack and lot sizing problems, Akademik Platf. Mühendislik ve Fen Bilimleri Dergisi 6 (2) (2018) 142–152.
[36] Y. He, H. Xie, T.-L. Wong, X. Wang, A novel binary artificial bee colony algorithm for the set-union knapsack problem, Future Gener. Comput. Syst. 78 (2018) 77–86.
[37] A.N. Hasoon, Projects selection in Knapsack problem by using artificial bee colony algorithm, Tikrit J. Pure Sci. 23 (2) (2018) 137–142.
[38] S. Zhang, S. Liu, A discrete improved artificial bee colony algorithm for 0–1 knapsack problem, IEEE Access 7 (2019) 104982–104991.
[39] M. Nouioua, Z. Li, S. Jiang, New binary artificial bee colony for the 0-1 Knapsack problem, in: International Conference on Swarm Intelligence, Springer, 2018, pp. 153–165.
[40] A. Baykasoglu, L. Ozbakir, P. Tapkan, Artificial bee colony algorithm and its application to generalized assignment problem, Itech Education and Publishing Vienna, 2007,
[41] W. Hu, Y. Wang, J. Zheng, Research on warehouse allocation problem based on the artificial bee colony inspired particle swarm optimization (ABC-PSO) algorithm, in: 2012 Fifth International Symposium on Computational Intelligence and Design, Vol. 1, IEEE, 2012, pp. 173–176.
[42] E.M. Bernardino, A.M. Bernardino, J.M. Sánchez-Pérez, J.A.G. Pulido, M.A.V. Rodríguez, Swarm optimisation algorithms applied to large balanced communication networks, J. Netw. Comput. Appl. 36 (1) (2013) 504–522.

[43] J. Banda, A. Singh, A hybrid artificial bee colony algorithm for the terminal assignment problem, in: International Conference on Swarm, Evolutionary, and Memetic Computing, Springer, 2014, pp. 134–144.
[44] G. Behzadi, B. Sundarakani, Practical ABC intelligence solution for Quadratic Assignment Problems, in: Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management, 2014, pp. 959–966.
[45] J. Shao, L. Yang, L. Peng, T. Chi, X. Wang, An improved artificial bee colony-based approach for zoning protected ecological areas, PLoS One 10 (9) (2015).
[46] M. Metlicka, D. Davendra, Chaos driven discrete artificial bee algorithm for location and assignment optimisation problems, Swarm Evol. Comput. 25 (2015) 15–28.
[47] J.A. Sultan, D.A. Matrood, Z.M. Khaleel, Artificial bee colony for quadratic assignment problem: A hospital case study, J. Univ. Hum. Dev. 2 (3) (2016) 502–508.
[48] J.-Q. Li, J.-d. Wang, Q.-k. Pan, P.-y. Duan, H.-y. Sang, K.-z. Gao, Y. Xue, A hybrid artificial bee colony for optimizing a reverse logistics network system, Soft Comput. 21 (20) (2017) 6001–6018.
[49] J. Li, R. Zhang, Y. Yang, Multi-AUV autonomous task planning based on the scroll time domain quantum bee colony optimization algorithm in uncertain environment, PLoS One 12 (11) (2017).
[50] J. Li, R.B. Zhang, Multi-AUV distributed task allocation based on the differential evolution quantum bee colony optimization algorithm, Pol. Marit. Res. 24 (s3) (2017) 65–71.
[51] H. Wang, F. Zhu, W. Liao, X. Sun, Optimal solution of robots task assignment problem based on improved artificial bee colony algorithm, in: 2017 International Conference on Advanced Mechatronic Systems (ICAMechS), IEEE, 2017, pp. 398–402.
[52] L. Yang, A. Zhu, J. Shao, T. Chi, A knowledge-informed and pareto-based artificial bee colony optimization algorithm for multi-objective land-use allocation, ISPRS Int. J. Geo-Inf. 7 (2) (2018) 63.
[53] Z. Yilmaz, F. Basciftci, Binary artificial bee colony algorithm to solve single objective resource allocation problem, Int. J. Future Comput. Commun. 7 (1) (2018).
[54] S. Samanta, D. Philip, S. Chakraborty, Bi-objective dependent location quadratic assignment problem: Formulation and solution using a modified artificial bee colony algorithm, Comput. Ind. Eng. 121 (2018) 8–26.
[55] S. Samanta, D. Philip, S. Chakraborty, A quick convergent artificial bee colony algorithm for solving quadratic assignment problems, Comput. Ind. Eng. 137 (2019) 106070.
[56] Y. Aratsu, K. Mizuno, H. Sasaki, S. Nishihara, Artificial bee colony for constraint satisfaction problems, SCIS-ISIS2012, 2012, pp. 2283–2286.
[57] M.H. Kashan, N. Nahavandi, A.H. Kashan, Disabc: A new artificial bee colony algorithm for binary optimization, Appl. Soft Comput. 12 (1) (2012) 342–352.
[58] Y. Aratsu, K. Mizuno, H. Sasaki, S. Nishihara, Experimental evaluation of artificial bee colony with greedy scouts for constraint satisfaction problems, in: 2013 Conference on Technologies and Applications of Artificial Intelligence, IEEE, 2013, pp. 134–139.
[59] Y. Aratsu, K. Mizuno, H. Sasaki, S. Nishihara, Solving constraint satisfaction problems by artificial bee colony with greedy scouts, Proc. World Congr. Eng. Comput. Sci. 1 (1) (2013) 1–6.
[60] M.S.M. Kasihmuddin, M.A. Mansor, S. Sathasivam, Artificial bee colony in the hopfield network for maximum k-satisfiability problem, J. Inf. Math. Sci. 8 (5) (2016) 317–334.
[61] M.S.M. Kasihmuddin, M. Mansor, S. Sathasivam, et al., Robust artificial bee colony in the hopfield network for 2-satisfiability problem., Pertanika J. Sci. Technol. 25 (2) (2017).
[62] Y. Guo, C. Zhang, A hybrid artificial bee colony algorithm for satisfiability problems based on tabu search, in: 2017 3rd IEEE International Conference on Computer and Communications (ICCC), IEEE, 2017, pp. 2226–2230.
[63] Y. Guo, C. Zhang, Research on neighborhood search strategy of artificial bee colony algorithm for satisfiability problems, in: 2017 10th International Symposium on Computational Intelligence and Design (ISCID), Vol. 1, IEEE, 2017, pp. 123–126.
[64] N. Tuncbilek, F. Tasgetiren, S. Esnaf, Artificial bee colony optimization algorithm for uncapacitated facility location problems, J. Econ. Soc. Res. 14 (1) (2012).
[65] M.S. Kiran, M. Gündüz, XOR-based artificial bee colony algorithm for binary optimization, Turk. J. Electr. Eng. Comput. Sci. 21 (Suppl. 2) (2013) 2307–2328.
[66] A. Yurtkuran, E. Emel, A modified artificial bee colony algorithm forp-center problems, Sci. World J. 2014 (2014) 1–9, http://dx.doi.org/10.1155/2014/824196.
[67] M.S. Kiran, The continuous artificial bee colony algorithm for binary optimization, Appl. Soft Comput. 33 (2015) 15–23.
[68] M. Basti, M. Sevkli, An artificial bee colony algorithm for the p-median facility location problem, Int. J. Metaheuristics 4 (1) (2015) 91–113.
[69] Y. Watanabe, M. Takaya, A. Yamamura, Fitness function in ABC algorithm for uncapacitated facility location problem, in: Information and Communication Technology-EurAsia Conference, Springer, 2015, pp. 129–138.
[70] V. Chahkandi, M. Yaghoobi, G. Veisi, Feature selection with chaotic hybrid artificial bee colony algorithm based on fuzzy (CHABCF), J. Soft Comput. Appl. 2013 (1) (2013) 1–8.
[71] J. Cai, W. Zhu, H. Ding, F. Min, An improved artificial bee colony algorithm for minimal time cost reduction, Int. J. Mach. Learn. Cybern. 5 (5) (2014) 743–752.
[72] E. Hancer, B. Xue, D. Karaboga, M. Zhang, A binary ABC algorithm based on advanced similarity scheme for feature selection, Appl. Soft Comput. 36 (2015) 334–348.
[73] D. Ye, Z. Chen, A new approach to minimum attribute reduction based on discrete artificial bee colony, Soft Comput. 19 (7) (2015) 1893–1903.
[74] Z.B. Özger, B. Bolat, B. Dırı, A comparative study on binary artificial bee colony optimization methods for feature selection, in: 2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), IEEE, 2016, pp. 1–4.
[75] P. Shunmugapriya, S. Kanmani, A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC hybrid), Swarm Evol. Comput. 36 (2017) 27–36.
[76] A. Zabidi, N. Md Tahir, I. Mohd Yassin, Z.I. Rizman, The performance of binary artificial bee colony (BABC) in structure selection of polynomial NARX and NARMAX models, Int. J. Adv. Sci. Eng. Inf. Technol. 7 (2) (2017) 373–379.
[77] A. Zabidi, I. Yassin, N. Tahir, Z. Rizman, M. Karbasi, Comparison between binary particles swarm optimization (BPSO) and binary artificial bee colony (BABC) for nonlinear autoregressive model structure selection of chaotic data, J. Fundam. Appl. Sci. 9 (3S) (2017) 730–754.
[78] E. Hancer, B. Xue, M. Zhang, D. Karaboga, B. Akay, Pareto Front feature selection based on artificial bee colony optimization, Inform. Sci. 422 (2018) 462–479.
[79] H. Rao, X. Shi, A.K. Rodrigue, J. Feng, Y. Xia, M. Elhoseny, X. Yuan, L. Gu, Feature selection based on artificial bee colony and gradient boosting decision tree, Appl. Soft Comput. 74 (2019) 634–642.
[80] C.J. Santana Jr., M. Macedo, H. Siqueira, A. Gokhale, C.J. Bastos-Filho, A novel binary artificial bee colony algorithm, Future Gener. Comput. Syst. 98 (2019) 180–196.
[81] E. Zorarpacı, S.A. Özel, A hybrid approach of differential evolution and artificial bee colony for feature selection, Expert Syst. Appl. 62 (2016) 91–103.
[82] A. Singh, An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem, Appl. Soft Comput. 9 (2) (2009) 625–631.
[83] X. Zhang, X. Zhang, A binary artificial bee colony algorithm for constructing spanning trees in vehicular ad hoc networks, Ad Hoc Netw. 58 (2017) 198–204.
[84] Y. Li, Y. Zhao, Y. Zhang, A spanning tree construction algorithm for industrial wireless sensor networks based on quantum artificial bee colony, EURASIP J. Wireless Commun. Networking 2019 (1) (2019) 176.
[85] S. Sundar, A. Singh, A swarm intelligence approach to the quadratic minimum spanning tree problem, Inform. Sci. 180 (17) (2010) 3182–3191.
[86] A. Singh, S. Sundar, An artificial bee colony algorithm for the minimum routing cost spanning tree problem, Soft Comput. 15 (12) (2011) 2489–2499.
[87] K. Singh, S. Sundar, Artifical bee colony algorithm using problem-specific neighborhood strategies for the tree t-spanner problem, Appl. Soft Comput. 62 (2018) 110–118.
[88] L. Hong-mei, W. Zhuo-fu, L. Hui-min, Artificial bee colony algorithm for real estate portfolio optimization based on risk preference coefficient, in: 2010 International Conference on Management Science & Engineering 17th Annual Conference Proceedings, IEEE, 2010, pp. 1682–1687.
[89] Z. Wang, S. Liu, X. Kong, Artificial bee colony algorithm for portfolio optimization problems, Int. J. Adv. Comput. Technol. 4 (4) (2012) 8–16.
[90] Z. Wang, R. Ouyang, X. Kong, A hybrid artificial bee colony algorithm for portfolio optimization problem 1, J. Theoret. Appl. Inf. Technol. 9 (1) (2013) 94–100.
[91] A.H. Chen, Y.-C. Liang, C.-C. Liu, An artificial bee colony algorithm for the cardinality-constrained portfolio optimization problems, in: 2012 IEEE Congress on Evolutionary Computation, IEEE, 2012, pp. 1–8.
[92] A.H. Chen, Y.-C. Liang, C.-C. Liu, Portfolio optimization using improved artificial bee colony approach, in: 2013 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr), IEEE, 2013, pp. 60–67.
[93] M. Ge, Artificial bee colony algorithm for portfolio optimization, in: Fifth International Conference on Intelligent Control and Information Processing, IEEE, 2014, pp. 449–453.
[94] M. Tuba, N. Bacanin, Artificial bee colony algorithm hybridized with firefly algorithm for cardinality constrained mean-variance portfolio selection problem, Appl. Math. Inf. Sci. 8 (6) (2014) 2831.

[95] W. Chen, Artificial bee colony algorithm for constrained possibilistic portfolio optimization problem, Physica A 429 (2015) 125–139.
[96] S.M. Seyedhosseini, M.J. Esfahani, M. Ghaffari, A novel hybrid algorithm based on a harmony search and artificial bee colony for solving a portfolio optimization problem using a mean-semi variance approach, J. Central South Univ. 23 (1) (2016) 181–188.
[97] C.B. Kalayci, O. Ertenlice, H. Akyer, H. Aygoren, An artificial bee colony algorithm with feasibility enforcement and infeasibility toleration procedures for cardinality constrained portfolio optimization, Expert Syst. Appl. 85 (2017) 61–75.
[98] C.B. Kalayci, O. Polat, M.A. Akbay, An efficient hybrid metaheuristic algorithm for cardinality constrained portfolio optimization, Swarm Evol. Comput. 54 (2020) 100662.
[99] M. Mansourinia, A. Momeni, Cardinality constrained portfolio optimization using bee colony algorithm (case study: Tehran stock exchange), Electron. Bus. 17 (11) (2018).
[100] I. Strumberger, E. Tuba, N. Bacanin, M. Beko, M. Tuba, Hybridized artificial bee colony algorithm for constrained portfolio optimization problem, in: 2018 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2018, pp. 1–8.
[101] B. Crawford, R. Soto, R. Cuesta, F. Paredes, Application of the artificial bee colony algorithm for solving the set covering problem, Sci. World J. 2014 (2014).
[102] R. Cuesta, B. Crawford, R. Soto, F. Paredes, An artificial bee colony algorithm for the set covering problem, in: Modern Trends and Techniques in Computer Science, Springer, 2014, pp. 53–63.
[103] B. Crawford, R. Soto, W. Palma, F. Johnson, F. Paredes, E. Olguín, A 2-level approach for the set covering problem: Parameter tuning of artificial bee colony algorithm by using genetic algorithm, in: International Conference in Swarm Intelligence, Springer, 2014, pp. 189–196.
[104] B. Crawford, R. Soto, E. Monfroy, G. Astorga, J. García, E. Cortes, A meta-optimization approach for covering problems in facility location, in: Workshop on Engineering Applications, Springer, 2017, pp. 565–578.
[105] B. Crawford, R. Soto, R. Cuesta, M. Olivares-Suárez, F. Johnson, E. Olguín, Two swarm intelligence algorithms for the set covering problem, in: 2014 9th International Conference on Software Engineering and Applications (ICSOFT-EA), IEEE, 2014, pp. 60–69.
[106] K. Chandrasekaran, S. Hemamalini, S.P. Simon, N.P. Padhy, Thermal unit commitment using binary/real coded artificial bee colony algorithm, Electr. Power Syst. Res. 84 (1) (2012) 109–119.
[107] K. Chandrasekaran, S.P. Simon, Multi-objective unit commitment problem with reliability function using fuzzified binary real coded artificial bee colony algorithm, IET gener. Transm. Dist. 6 (10) (2012) 1060–1073.
[108] M. Govardhan, R. Roy, Profit based unit commitment using gbest artificial bee colony algorithm, in: 2013 3rd International Conference on Electric Power and Energy Conversion Systems, IEEE, 2013, pp. 1–6.
[109] P.K. Singhal, R. Naresh, V. Sharma, N.G. Kumar, A new strategy based artificial bee colony algorithm for unit commitment problem, in: 2015 International Conference on Recent Developments in Control, Automation and Power Engineering (RDCAPE), IEEE, 2015, pp. 55–60.
[110] P.K. Singhal, R. Naresh, V. Sharma, A modified binary artificial bee colony algorithm for ramp rate constrained unit commitment problem, Int. Trans. Electr. Energy Syst. 25 (12) (2015) 3472–3491.
[111] P.K. Singhal, R. Naresh, V. Sharma, A novel strategy-based hybrid binary artificial bee colony algorithm for unit commitment problem, Arab. J. Sci. Eng. 40 (5) (2015) 1455–1469.
[112] P. Singhal, R. Naresh, V. Sharma, Comparative performance of artificial bee colony based algorithms for wind-thermal unit commitment, Int. J. Electr. Comput. Eng. 10 (8) (2016) 1087–1098.
[113] C. Shui, W. Szeto, An artificial bee colony algorithm for the public bike repositioning problem, in: 37th Australasian Transport Research Forum, ATRF 2015, 2015, pp. 1–12.
[114] Y.-j. Tian, Q.-h. Xie, Dynamic scheduling of public bicycles based on artificial bee colony algorithm, in: Proceedings of the 23rd International Conference on Industrial Engineering and Engineering Management 2016, Springer, 2017, pp. 245–249.
[115] C.S. Shui, W. Szeto, Dynamic green bike repositioning problem–A hybrid rolling horizon artificial bee colony algorithm approach, Transp. Res. D 60 (2018) 119–136.
[116] P. Zhao, X. Sun, P. Chen, An improved binary bee colony algorithm for satellite resource scheduling method, in: International Conference on Signal and Information Processing, Networking and Computers, Springer, 2017, pp. 181–188.
[117] K. Luo, A hybrid binary artificial bee colony algorithm for the satellite photograph scheduling problem, Eng. Optim. (2019) 1–20.
[118] I. Oliveira, R. Schirru, et al., Identifying nuclear power plant transients using the Discrete Binary Artificial Bee Colony (DBABC) algorith, in: International Conference on Mathematics and Computational Methods Applied To Nuclear Science and Engineering (M&C 2011), Latin American Section (LAS) / American Nuclear Society (ANS), ISBN: 978-85-63688-00-2, 2011, pp. 1–9.
[119] G. Pampará, A.P. Engelbrecht, Binary artificial bee colony optimization, in: 2011 IEEE Symposium on Swarm Intelligence, IEEE, 2011, pp. 1–8.
[120] W. Qi, H. Shen, H. Chen, A discrete artificial bee colony algorithm for RFID network scheduling, Int. J. Adv. Comput. Technol. 4 (14) (2012) 324–332.
[121] M. Kaur, K. Singh, An algorithm for min cut partitioning for digital circuit layout based on evolutionary approach, IRACST–Eng. Sci. Technol. Int. J. 2 (4) (2012).
[122] 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. Pers. Commun. 65 (2) (2012) 425–442.
[123] L. Wei, C. Hanning, BABC: a binary version of artificial bee colony algorithm for discrete optimization, Int. J. Adv. Comput. Technol. 4 (14) (2012) 307–314.
[124] W. Liu, B. Niu, H. Chen, Y. Zhu, Artificial bee colony algorithm for reader collision problem in radio frequency identification network, J. Comput. Theoret. Nanosci. 10 (12) (2013) 2877–2883.
[125] T. Bayraktar, M.E. Aydin, M. Dugenci, A memory-integrated artificial bee algorithm for 1-D bin packing problems, in: Proc. CIE IMSS, 2014, pp. 1023–1034.
[126] D. Jia, X. Duan, M.K. Khan, Binary artificial bee colony optimization using bitwise operation, Comput. Ind. Eng. 76 (2014) 360–365.
[127] M. Yahya, M. Saka, Construction site layout planning using multi-objective artificial bee colony algorithm with Levy flights, Autom. Constr. 38 (2014) 14–29.
[128] M. Mandala, C. Gupta, Binary artificial bee colony optimization for GENCOs' profit maximization under pool electricity market, Int. J. Comput. Appl. 90 (19) (2014).
[129] H. Zhang, D.-Y. Ye, Key-node-based local search discrete artificial bee colony algorithm for obstacle-avoiding rectilinear steiner tree construction, Neural Comput. Appl. 26 (4) (2015) 875–898.
[130] M. Al-Salamah, Constrained binary artificial bee colony to minimize the makespan for single machine batch processing with non-identical job sizes, Appl. Soft Comput. 29 (2015) 379–385.
[131] M. Rigakis, D. Trachanatzi, M. Marinaki, Y. Marinakis, Artificial bee colony optimization approach to develop strategies for the iterated prisoner's dilemma, in: 7th International Conference of Bionspired Methods and their Applications, Bled, Slovenia, 2016, pp. 18–20.
[132] K. Arul Jeyaraj, V. Rajasekaran, S. Nandha Kumar, K. Chandrasekaran, A multi-objective placement of phasor measurement units using fuzzified artificial bee colony algorithm, considering system observability and voltage stability, J. Exp. Theor. Artif. Intell. 28 (1–2) (2016) 113–136.
[133] S.-S. Kim, S. McLoone, J.-H. Byeon, S. Lee, H. Liu, Cognitively inspired artificial bee colony clustering for cognitive wireless sensor networks, Cogn. Comput. 9 (2) (2017) 207–224.
[134] K. Gao, Y. Zhang, A. Sadollah, R. Su, Improved artificial bee colony algorithm for solving urban traffic light scheduling problem, in: 2017 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2017, pp. 395–402.
[135] K. Li, Z. Yuan, X. Jiang, A novel damaged ship righting plan optimization method based on artificial bee colony algorithm, J. Mar. Sci. Technol. 26 (5) (2018) 667–677.
[136] D. Kong, T. Chang, J. Pan, N. Hao, K. Xu, L. Zhang, G. Yang, A decision variable-based combinatorial optimization approach for interval-valued intuitionistic fuzzy MAGDM, Inform. Sci. 484 (2019) 197–218.
[137] X. Chen, G. Lin, M. Xu, Applying a binary artificial bee colony algorithm to the max-cut problem, in: 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), IEEE, 2019, pp. 1–4.
[138] G. Chen, P. Sun, J. Zhang, Repair strategy of military communication network based on discrete artificial bee colony algorithm, IEEE Access 8 (2020) 73051–73060.
[139] A. Saad, S.A. Khan, A. Mahmood, A multi-objective evolutionary artificial bee colony algorithm for optimizing network topology design, Swarm Evol. Comput. 38 (2018) 187–201.
[140] Y. Wang, W. Chen, C. Tellambura, A PAPR reduction method based on artificial bee colony algorithm for OFDM signals, IEEE Trans. Wirel. Commun. 9 (10) (2010) 2994–2999.
[141] N. Taspinar, D. Karaboga, M. Yildirim, B. Akay, Partial transmit sequences based on artificial bee colony algorithm for peak-to-average power ratio reduction in multicarrier code division multiple access systems, IET Commun. 5 (8) (2011) 1155–1162.
[142] N. Taspinar, D. Karaboga, M. Yildirim, B. Akay, PAPR reduction using artificial bee colony algorithm in OFDM systems, Turk. J. Electr. Eng. Comput. Sci. 19 (1) (2011) 47–58.
[143] H.V.T. Dung, H. Bang, C.-S. Park, S. Myong, PAPR Reduction using PTS with low computational complexity in coherent optical OFDM systems, in: 2012 18th Asia-Pacific Conference on Communications (APCC), IEEE, 2012, pp. 629–634.
[144] X. Yu, S. Li, Z. Zhu, T. Zhang, An improved artificial bee colony-partial transmit sequence algorithm for PAPR reduction in OFDM systems, Int. J. Wirel. Mob. Comput. 6 (5) (2013) 473–480.

[145] X. Cheng, D. Liu, S. Feng, Q. Pan, H. Fang, PTS Based on disabc algorithm for PAPR reduction in OFDM systems, Electron. Lett. 54 (6) (2018) 397–398.
[146] Z. Yin, X. Liu, Z. Wu, A multiuser detector based on artificial bee colony algorithm for DS-UWB systems, Sci. World J. 2013 (2013).
[147] S. Ashrafinia, M. Naeem, D. Lee, Discrete artificial bee colony for computationally efficient symbol detection in multidevice STBC MIMO systems, Adv. Artif. Intell. 2013 (2013) 1–14, http://dx.doi.org/10.1155/2013/578710.
[148] R. Durgut, H. Kutucu, S. Akleylek, An Artificial Bee Colony Algorithm for Solving the Weapon Target Assignment Problem, in: Proceedings of the 7th International Conference on Information Communication and Management, 2017, pp. 28–31.
[149] T. Chang, D. Kong, N. Hao, K. Xu, G. Yang, Solving the dynamic weapon target assignment problem by an improved artificial bee colony algorithm with heuristic factor initialization, Appl. Soft Comput. 70 (2018) 845–863.
[150] D. Guo, X. Dong, Q. Li, Z. Ren, Weapon target assignment method with grouping constraints for interception based on artificial bee colony algorithm, in: 2019 IEEE 15th International Conference on Control and Automation (ICCA), IEEE, 2019, pp. 1385–1390.
[151] J.A. Pacurib, G.M.M. Seno, J.P.T. Yusiong, Solving sudoku puzzles using improved artificial bee colony algorithm, in: 2009 Fourth International Conference on Innovative Computing, Information and Control (ICICIC), IEEE, 2009, pp. 885–888.
[152] J.P.T. Yusiong, J.A. Pacurib, G.M.M. Seno, SudokuBee: An artificial bee colony-based approach in solving Sudoku puzzles, Int. J. Adv. Res. Comput. Sci. 1 (3) (2010).
[153] B. Akay, I. Kirmizi, Structural optimization of wavelet packets using swarm algorithms, in: 2012 IEEE Congress on Evolutionary Computation, IEEE, 2012, pp. 1–5.
[154] A. Draa, A. Bouaziz, An artificial bee colony algorithm for image contrast enhancement, Swarm Evol. Comput. 16 (2014) 69–84.
[155] K. Guo, Q. Zhang, A discrete artificial bee colony algorithm for the reverse logistics location and routing problem, Int. J. Inf. Technol. Decis. Mak. 16 (05) (2017) 1339–1357.
[156] B. Arun, T.V. Kumar, Materialized view selection using artificial bee colony optimization, Int. J. Intell. Inf. Technol.(IJIIT) 13 (1) (2017) 26–49.
[157] R.F.O. Adebiyi, K.A. Abubilal, A.M.S. Tekanyi, B.H. Adebiyi, Management of vehicular traffic system using artificial bee colony algorithm, Int. J. Image Graph. Signal Process. 9 (11) (2017) 18.
[158] Amarjeet, J.K. Chhabra, et al., Many-objective artificial bee colony algorithm for large-scale software module clustering problem, Soft Comput. 22 (19) (2018) 6341–6361.
[159] 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.
[160] F. Abu-Mouti, M. El-Hawary, Modified artificial bee colony algorithm for optimal distributed generation sizing and allocation in distribution systems, in: 2009 IEEE Electrical Power & Energy Conference (EPEC), IEEE, 2009, pp. 1–9.
[161] H. Nasiraghdam, S. Jadid, Optimal hybrid PV/WT/FC sizing and distribution system reconfiguration using multi-objective artificial bee colony (MOABC) algorithm, Sol. Energy 86 (10) (2012) 3057–3071.
[162] O. Abedinia, E.S. Barazandeh, Interactive artificial bee colony based on distribution planning with renewable energy units, in: 2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT), IEEE, 2013, pp. 1–6.
[163] A. El-Zonkoly, Optimal placement and schedule of multiple grid connected hybrid energy systems, Int. J. Electr. Power Energy Syst. 61 (2014) 239–247.
[164] M. Kefayat, A.L. Ara, S.N. Niaki, A hybrid of ant colony optimization and artificial bee colony algorithm for probabilistic optimal placement and sizing of distributed energy resources, Energy Convers. Manage. 92 (2015) 149–161.
[165] S. Das, A. Verma, P.R. Bijwe, Transmission network expansion planning using a modified artificial bee colony algorithm, Int. Trans. Electr. Energy Syst. 27 (9) (2017) e2372.
[166] C.K. Das, O. Bass, G. Kothapalli, T.S. Mahmoud, D. Habibi, Optimal placement of distributed energy storage systems in distribution networks using artificial bee colony algorithm, Appl. Energy 232 (2018) 212–228.
[167] S. Ajorlou, I. Shams, M.G. Aryanezhad, Optimization of a multiproduct conwip-based manufacturing system using artificial bee colony approach, in: World Congress on Engineering 2012. July 4-6, 2012. London, UK., Vol. 2189, International Association of Engineers, 2011, pp. 1385–1388.
[168] S. Ajorlou, I. Shams, Artificial bee colony algorithm for CONWIP production control system in a multi-product multi-machine manufacturing environment, J. Intell. Manuf. 24 (6) (2013) 1145–1156.
[169] X. Wang, Z. Wang, X. Xu, An improved artificial bee colony approach to QoS-aware service selection, in: 2013 IEEE 20th International Conference on Web Services, IEEE, 2013, pp. 395–402.
[170] X. Xu, Z. Liu, S-ABC-A service-oriented artificial bee colony algorithm for global optimal services selection in concurrent requests environment, in: 2014 IEEE International Conference on Web Services, IEEE, 2014, pp. 503–509.
[171] Y. Huo, Y. Zhuang, J. Gu, S. Ni, Y. Xue, Discrete gbest-guided artificial bee colony algorithm for cloud service composition, Appl. Intell. 42 (4) (2015) 661–678.
[172] J. Lartigau, X. Xu, L. Nie, D. Zhan, Cloud manufacturing service composition based on QoS with geo-perspective transportation using an improved Artificial Bee Colony optimisation algorithm, Int. J. Prod. Res. 53 (14) (2015) 4380–4404.
[173] X. Xu, Z. Liu, Z. Wang, Q.Z. Sheng, J. Yu, X. Wang, S-ABC: A paradigm of service domain-oriented artificial bee colony algorithms for service selection and composition, Future Gener. Comput. Syst. 68 (2017) 304–319.
[174] A.F. Deepa, J. Sathiaseelan, Multi-Objective QoS-based Reusable Service Selection enhanced with Artificial Bee Colony (MQRSABC), 2017.
[175] Y. Cheng, C. Ding, Optimization of web services composition using artificial bee colony algorithm, in: 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), IEEE, 2017, pp. 1–6.
[176] J. Zhou, X. Yao, A hybrid artificial bee colony algorithm for optimal selection of QoS-based cloud manufacturing service composition, Int. J. Adv. Manuf. Technol. 88 (9–12) (2017) 3371–3387.
[177] J. Zhou, X. Yao, A hybrid approach combining modified artificial bee colony and cuckoo search algorithms for multi-objective cloud manufacturing service composition, Int. J. Prod. Res. 55 (16) (2017) 4765–4784, http://dx.doi.org/10.1080/00207543.2017.1292064.
[178] J. Zhou, X. Yao, Multi-population parallel self-adaptive differential artificial bee colony algorithm with application in large-scale service composition for cloud manufacturing, Appl. Soft Comput. 56 (2017) 379–397.
[179] J. Zhou, X. Yao, DE-caABC: differential evolution enhanced context-aware artificial bee colony algorithm for service composition and optimal selection in cloud manufacturing, Int. J. Adv. Manuf. Technol. 90 (1–4) (2017) 1085–1103.
[180] J. Zhou, X. Yao, Y. Lin, F.T. Chan, Y. Li, An adaptive multi-population differential artificial bee colony algorithm for many-objective service composition in cloud manufacturing, Inform. Sci. 456 (2018) 50–82.
[181] M. Chandra, R. Niyogi, Web service selection using modified artificial bee colony algorithm, IEEE Access 7 (2019) 88673–88684.
[182] F. Dahan, H. Mathkour, M. Arafah, Two-step artificial bee colony algorithm enhancement for QoS-aware Web service selection problem, IEEE Access 7 (2019) 21787–21794.
[183] X. Wang, X. Xu, Q.Z. Sheng, Z. Wang, L. Yao, Novel artificial bee colony algorithms for QoS-aware service selection, IEEE Trans. Serv. Comput. 12 (2) (2019) 247–261, http://dx.doi.org/10.1109/tsc.2016.2612663.
[184] D.J. Mala, V. Mohan, ABC tester-artificial bee colony based software test suite optimization approach, Int. J. Softw. Eng. 2 (2) (2009) 15–43.
[185] A. Srikanth, N.J. Kulkarni, K.V. Naveen, P. Singh, P.R. Srivastava, et al., Test case optimization using artificial bee colony algorithm, in: International Conference on Advances in Computing and Communications, Springer, 2011, pp. 570–579.
[186] A.K. Joseph, G. Radhamani, A hybrid model of particle swarm optimization (PSO) and artificial bee colony (ABC) algorithm for test case optimization, Int. J. Comput. Sci. Eng.(IJCSE) 3 (5) (2011).
[187] P. Bansal, S. Sabharwal, N. Mittal, S. Arora, ABC-CAG: covering array generator for pair-wise testing using artificial bee colony algorithm, e-Informatica Softw. Eng. J. 10 (1) (2016).
[188] A.A. Alsewari, A.K. Alazzawi, T.H. Rassem, M.N. Kabir, A.A.B. Homaid, Y.A. Alsariera, N.M. Tairan, K.Z. Zamli, ABC Algorithm for combinatorial testing problem, J. Telecommun. Electron. Comput. Eng.(JTEC) 9 (3–3) (2017) 85–88.
[189] A.K. Alazzawi, A.A.B. Homaid, A.A. Alomoush, A.A. Alsewari, Artificial bee colony algorithm for pairwise test generation, J. Telecommun. Electron. Comput. Eng.(JTEC) 9 (1–2) (2017) 103–108.
[190] A.K. Alazzawi, H.M. Rais, S. Basri, ABCVS: An artificial bee colony for generating variable t-way test sets, Int. J. Adv. Comput. Sci. Appl. 10 (4) (2019) 259–274.
[191] A.K. Alazzawi, H.M. Rais, S. Basri, Parameters tuning of hybrid artificial bee colony search based strategy for t-way testing, Int. J. Innov. Technol. Exploring Eng. 8 (5S) (2019) 204–212.
[192] A.K. Alazzawi, H.M. Rais, S. Basri, Hybrid artificial bee colony algorithm for t-way interaction test suite generation, in: Computer Science on-Line Conference, Springer, 2019, pp. 192–199.
[193] A.K. Alazzawi, H. Rais, S. Basri, HABC: Hybrid artificial bee colony for generating variable t-way test sets, J. Eng. Sci. Technol. 15 (2) (2020) 746–767.
[194] M.S.A.R. Ali, R.R. Othman, Z.R. Yahya, M.Z. Zahir, A modified artificial bee colony based test suite generation strategy for uniform T-way testing, IOP Conf. Ser. Mater. Sci. Eng. 767 (2020) 012020.

[195] H. Garg, Solving structural engineering design optimization problems using an artificial bee colony algorithm, J. Ind. Manag. Optim. 10 (3) (2014) 777–794.
[196] A. Hadidi, S.K. Azad, S.K. Azad, Structural optimization using artificial bee colony algorithm, in: 2nd International Conference on Engineering Optimization, 2010, pp. 6–9.
[197] S. Omkar, J. Senthilnath, R. Khandelwal, G.N. Naik, S. Gopalakrishnan, Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures, Appl. Soft Comput. 11 (1) (2011) 489–499.
[198] B. Akay, D. Karaboga, Artificial bee colony algorithm for large-scale problems and engineering design optimization, J. Intell. Manuf. 23 (4) (2012) 1001–1014.
[199] I. Aydogdu, A. Akın, M. Saka, Optimum Design of Steel Space Frames by Artificial Bee Colony Algorithm, ACE, 2012.
[200] O. Sevim, M. Sönmez, B. Toprak, Optimum design of truss structure using artificial bee colony algorithm, Int. J. Soft Comput. Artif. Intell. 4 (2016) 17–20.
[201] S. Carbas, M. Saka, Discrete design optimization of cold-formed thin-walled open sections subjected to various external loading, in: International Conference o Advances in Applied and Computational Mechanics, İZmir, TÜRkiye, 2015, pp. 5–7.
[202] S. Carbas, M.P. Saka, Optimum design of cold-formed thin-walled sections subjected to axial and bi-axial bending using artificial bee colony algorithm, Res. Eng. Struct. Mater. (2016) http://dx.doi.org/10.17515/resm2015.13st0816.
[203] L. Yancang, W. Jie, L. Libo, Z. Jie, Improved artificial bee algorithm for reliability-based optimization of truss structures, Open Civ. Eng. J. 11 (1) (2017).
[204] C. Dong, Z. Xiong, X. Liu, Y. Ye, Y. Yang, W. Guo, Dual-search artificial bee colony algorithm for engineering optimization, IEEE Access 7 (2019) 24571–24584.
[205] 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.
[206] W.Y. Szeto, Y. Jiang, Hybrid artificial bee colony algorithm for transit network design, Transp. Res. Rec. 2284 (1) (2012) 47–56.
[207] Y. Jiang, W. Szeto, T. Ng, Transit network design: A hybrid enhanced artificial bee colony approach and a case study, Int. J. Transp. Sci. Technol. 2 (3) (2013) 243–260.
[208] W.Y. Szeto, Y. Jiang, Transit route and frequency design: Bi-level modeling and hybrid artificial bee colony algorithm approach, Transp. Res. B 67 (2014) 235–263.
[209] M. Ekhtiari, S. Poursafary, Multiobjective stochastic programming for mixed integer vendor selection problem using artificial bee colony algorithm, ISRN Artif. Intell. 2013 (2013) 1–13.
[210] H. Garg, M. Rani, S. Sharma, An efficient two phase approach for solving reliability–redundancy allocation problem using artificial bee colony technique, Comput. Oper. Res. 40 (12) (2013) 2961–2969.
[211] H. Hu, K.K.H. Ng, Y. Qin, Robust parallel machine scheduling problem with uncertainties and sequence-dependent setup time, Sci. Program. 2016 (2016) 1–13, http://dx.doi.org/10.1155/2016/5127253.
[212] X. Li, H. Yang, M. Yang, G. Yang, Flexible time-of-use tariff with dynamic demand using artificial bee colony with transferred memory scheme, Swarm Evol. Comput. 46 (2019) 235–251.
[213] S. Ozcan, F. Simsir, A new model based on Artificial Bee Colony algorithm for preventive maintenance with replacement scheduling in continuous production lines, Eng. Sci. Technol. Int. J. 22 (6) (2019) 1175–1186.