



Directed Artificial Bee Colony algorithm with revamped search strategy to solve global numerical optimization problems

Kalaipriyan Thirugnanasambandam¹ · M. Rajeswari² ·
Debnath Bhattacharyya³ · Jung-yoon Kim⁴

Accepted: 25 October 2021 / Published online: 3 January 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

Artificial Bee Colony algorithm (ABC) is inspired by behavior of food foraging of honeybees to solve the NP-Hard problems using optimization model which is one among the swarm intelligence algorithms. ABC is a widespread optimization algorithm to obtain the best solution from feasible solutions in the search space and strive harder than other existing population-based algorithms. However, in diversification process ABC algorithm shows good performance but lacks in intensification process and slows to convergence towards an optimal solution because of its search equations. In this work, the authors proposed an improvised solution search strategy at employed bee phase and onlooker bee phase by considering the advantages of the local-best, neighbor-best, and iteration-best solutions. Thus, the obtained candidate solutions are closer to the best solution by providing directional information to ABC algorithms. The search radius for new candidate solutions is adjusted in scout bee phase which facilitates to move towards global convergence. Thus, the process of diversification and intensification is balanced in this work. Finally, to assess the performance of the proposed algorithm, 20 numerical benchmarks functions are used. To show the significance of the proposed methodology it has been tested with Combined Heat and Economic Power Dispatch (CHPED) problem. The empirical result exhibits that the proposed algorithm provides higher quality solutions and outperform with original ABC algorithm for solving numerical optimization problems.

Keywords Artificial Bee Colony algorithm · Directional Search · Combined Heat and Economic Power Dispatch · Evolutionary Algorithm

✉ Jung-yoon Kim
kjyoonrch@gmail.com

Extended author information available on the last page of the article

1 Introduction

Optimization problems are habitually stumbled in the field of engineering design (Rao et al. 2008; Pawar et al. 2008), computer science and information theory (Karaboga and CelalOzturk. 2011; Özbakir and AdilBaykasoğlu et al.. 2010; Kaveh and Talatahari 2009), statistical physics and economics, etc. (Pan et al. 2011; Coelho and Leandro and Viviana CoccoMariani. 2009). Popular meta-heuristic approach Swarm intelligence (SI) technique which is a sub-domain of artificial intelligence, and it is employed to unravel various engineering and optimization problems. It is inspired by the foraging behavior of the flock of birds, social insects and schools of fish and they are capable of solving complex tasks. The problems which cannot be solved by standard mathematical methods are easily solved by swarm intelligence. Use of swarm intelligence has many advantages such as modularity, parallel processing, independent and self-governing, fault tolerance and scalability (Kassabalidis et al. 2001). Optimization is the method of searching optimal solution from the large search space which is very hard and requires more computational time.

The population-based optimization (PBO) algorithm is designed to calculate the near optimal solution to global numerical problems or engineering problems based on the fitness evaluation, and thus it moves towards best solution space. PBO are classified into two major groups as SI based algorithms and Evolution based algorithms (Passino 2002). In the last decade, the research towards various optimization algorithms increased especially in the meadow of biological-inspired optimization algorithms. Particle swarm optimization (PSO) based on the communal behavior of birds flocking and fish schooling (Liu et al. 2005; Xiang et al. 2007), ant colony optimization (ACO) inspired by the behavior of ant colonies (Dorigo et al. 1996), genetic algorithm (GA) based on the Darwinian law of survival and reproductive system of living species (Golberg 1989) and ABC inspired by the natural deeds of honey bee swarm (Karaboga 2005).

ABC algorithm is a biologically inspired optimization algorithm proposed by Karaboga is used to calculate the best near solution in statistical optimization or engineering problems. ABC is instigated by the foraging deeds of honey bee swarm for seeking food resource. The recital of ABC depends on the process of exploration and exploitation method to yield uniform and doable solution. Exploration is the self-determining search to obtain optimal solution and exploitation uses acquaintance of the previous best solution to search. Many researchers have developed algorithms to amplify the process of exploration and exploitation to solve numerical optimization problems. To appraise the performance of ABC algorithm, many benchmark functions of unimodal and multimodal distributions are employed. ABC is compared with known optimization algorithms like PSO, GA, and ACO to access the performance and the obtained results show ABC produce more optimal solution than other competitive methods (Singh 2009; Karaboga 2009; Kang et al. 2009).

Researchers found ABC is good at exploration in updating search equation but poor in exploitation process (Zhu and Kwong 2010). To achieve the equipoise

between exploration and exploitation process, it is essential to progress the local search in ABC (Karaboga and BahriyeAkay. 2009). To evade local optima and accelerate the convergence speed are the primary objectives appealing in ABC. Many modified and improved algorithms (Akay and DervisKaraboga. 2012) are projected in recent years like best-so-far ABC (Banharnsakun and TiraneAchalakul et al.. 2011), Bee Swarm Optimization (BSO) (Akbari and AlirezaMohammadi and KoorushZiarati. 2010), Gbest-guided ABC (GABC) (Zhu and Kwong 2010) and I-ABC (Gao and Liu 2011) and these algorithms show better performance than original ABC. The choice of ABC over other evolutionary algorithms for improvisation is due to its explored number of applications through it. If ABC can be improved then the supported applications will get an innovative advantage with its use.

At first, a modified exploration equation is presented in the employed bee phase, the information about global best, neighbor best and iteration best are embedded in the solution search equation to balance diversification and intensification process. Then search space radius is adjusted in the phase of scout bee to move towards global convergence, and at each iteration, the perturbation is regulated dynamically. Oppositional Learning concept is imposed over ABC for reducing the deviation of search while exploitation process is on board. In this work, directional information about the food source is incorporated and the algorithm is investigated in the numerical benchmark optimization problems. And also, the proposed model is used to solve CHPED and provide significant results.

This paper is structured as pursue: The literature survey on ABC algorithm is described in Sect. 2. Section 3 discusses the impression of ABC algorithm. The proposed algorithm is conferred in Sect. 4. Section 5 describes the simulation and experimental results for the formulated problem; Sect. 6 concludes the research work.

1.1 Literature review

In the typical ABC, it has many advantages such as local search, solution update, and memory and it produced good results (Basturk and Karaboga 2006). Various modifications and improvement of the algorithm are measured and analyzed on numerical optimization problems (Karaboga 2005). Several hybridizations of ABC algorithm are found in the literature review in Karaboga et al. (2014). However, ABC trapped in local optimum which influences the performance of convergence and result in uncertainty (Luo et al. 2013). Akay et al. (2012) introduced two real-parameter optimizations in ABC algorithm Modification Rate (MR) and scaling factor (SF) to increase the convergence rate and perturbation control. Dongli et al. (Dong Li et al. 2011) proposed better performance in ABC with three modifications. The search solution is updated based on neighborhood search and proposed new search equation with respect to modifications. To balance the exploitation and exploration process, various changes have made in standard ABC.

Gao et al. (2012) proposed a modified ABC algorithm inspired by the differential evolution (DE) by updating the search solution equation. To enlarge the exploitation

process and to increase search around best solution, previous iteration value is considered. To balance exploration and exploitation selective probability is introduced to get new search equation. Thus chaotic and opposition-based learning is incorporated to improve the process of global convergence. MABC algorithm prevents the bees from entrapping to the local minimum, which increases convergence globally and efficiently increase the ability of search.

Anuar et al. (Failed 2016) monitored the behavior of scout bee and made changes to improve the performance of exploration process. The proposed algorithm is named as ABC rate of change (ABC-ROC) since the scout bee phase is based on its change rate. The limit in ABC will control exploration in scout bee phase, the author calculated slope and plotted the performance graph. The process of the rate change in performance graph decides the exploration process. Ozturk et al. (2015) proposed the genetic method for discrete ABC by replacing with Jaccard coefficient. In discrete ABC, solution generation is improved by calculating the similarities obtained genetically from inspired components clustering dynamically. Karaboga et al. (2014) proposed quick ABC algorithm (qABC) to sculpt the onlooker bee phase to obtain accurate and improved performance in exploitation process. In the qABC algorithm, the best neighbor solution is incorporated in the candidate solution equation of onlooker bee phase. To decide a best neighborhood, mean Euclidean distance is calculated between neighbor and rest of the solutions.

Yan et al. (2012) proposed an algorithm for data clustering in swarm intelligence model as Hybrid ABC algorithm (HABC). The author incorporated the genetic algorithms crossover operator in ABC to increase the process of information exchange among honey bees. The crossover operator is included in the onlooker bee phase to augment the ability of the optimization in the algorithm. Maeda et al. (Maeda and Tsuda 2015) recommended the reduction of artificial ABC for global optimization. In initialization, the number of bees and positions are updated to avoid trapping into local minima. The bees in the process are sequentially reduced to reach the pre-determined value when the highest value is obtained the objective function opts. Stanarevic et al. (Stanarevic et al. 2010) introduced ABC algorithm for the constrained problem. To improve search equation in the initialization process, Smart Bee (SB) is pioneered to use the historical memory of location and search better food sources. Constraint problems are solved by using Deb technique instead of greedy selection.

Aderhold et al. (2010) commenced with two new modifications calculating the global best solution in addition to the random reference value in employee bee phase. The distance has been the global best solution location, and potential reference is selected for probability calculation. Yang et al. (2005) developed virtual bee algorithm (VBA) to solve engineering problems with two parameters. In this algorithm virtual swarm bees are produced and randomly move in space, to find the target food source virtual bees interact with each other based on the encoded values obtained from the function. The intensity of bee's interaction based on encoded function will solve the optimization problems. Chen et al. (2014) proposed enhanced ABC (EABC) which uses self-adaptive searching strategy to avoid trapping into local minima. To balance the exploitation and exploration process artificial immune network (ai-Net) operator is introduced. This operator will accelerate the global convergence rate in search selection process, and search strategy is introduced in

initialization process to obtain a novel generation. In 2021, Author Liling Sun, et al. (2020) proposed surrogate-assisted multi-swarm artificial bee colony (SAMSABC) with improvised exploitation model using orthogonal method. The experimental results over 20 complex numerical optimization algorithms shows its significance. Authors Xiao, et al. (2021) proposed ABC with adaptive neighborhood search and Gaussian perturbation (ABCNG). In this model, the initial phase of population were generated using neighborhood selection model dynamically. Then on the scout bee phase Gaussian Perturbation method is used to improve the unimproved solutions.

However, ABC is used to solve engineering optimization problems and large-scale problems besides solving global numerical optimization problems. Kang et al. (2009) developed a hybrid ABC algorithm uses Nelder-Mead (NM) simplex method for selection local selection process and structural inverse analysis. The performance of this hybrid algorithm outperforms with other heuristic algorithms. Samanta et al. (2011) used ABC to search optimal combinations used in NTM (non-traditional machine) process. This NTM process solves both single and multi-objective optimization problems. Yi et al. (Yi and He 2014) developed novel ABC algorithm (NABC) for solving numerical optimization problems to enhance exploitation process by integrating current best solution in search equation. The NABC is applied in many optimization problems like flow shop, vehicle routing problem and data mining.

Yurtkuran et al. (2016) introduced Enhanced ABC with solution acceptance rule and multi-search (ABC-SA). Replaced greedy selection method solution with acceptance rule, worst condition is decreased non-linearly for the search process. Probabilistic multi-search strategies equalize the process of exploration and exploitation. Wang et al. (2014) developed multi-strategy ABC which consists of many distinct search strategies coexists to produce offspring throughout the search process. Rajasekhar et al. (2011) proposed an improved version of ABC with mutation based on Levy probability distribution and Sobol probability distribution. Further comprehensive survey on variants of ABC can be instigated in Karaboga and Akay (2009).

2 ABC algorithm

ABC is an optimization algorithm which is inspired by natural behavior of honey was proposed by Karaboga in 2005 (Karaboga 2005). The global optimum solution or near optimum solutions can be identified by simulating the intellectual behavior of honeybees. In ABC each food source replicates particular problem's solution. It treasures the global optimum of a food source using fitness function and computes the qualitative nectar in the food source. This model of algorithm is straight-forward, population-based, and robust for solving engineering and global numerical problems. The colonies of artificial bees are assembled into three groups: employed bee, onlooker bee, and scout bee. A part of the colony members is employed bee (half of the group in precise), and the other half is onlooker bee. The number of food sources available is equal to employed bees and number of onlooker bees. The employed bees search for the new food source and distribute the gathered information about

the food source to recruit onlooker bee. The onlooker bee will select the higher quality food source than low quality food source based on probability information. The employed bee whose food source is low quality are rejected and they are changed to scout bee for further searching newer food sources. ABC algorithm iteratively repeats these three units. In the first phase, employed bees search for the food source and modernize the position based on the quality of the food source. In the second phase, food source with higher probability values is chosen by onlooker bee based on the fitness. In the third phase, the low-quality food source is abandoned and reinstated with newer food source by scout bees. Figure 1 shows the flow of ABC algorithm. These three units are summarized as follows:

2.1 Population initialization

The initialization of the population is generated by the randomly distributed collection of FP solutions, where FP denotes the total number of employed or onlooker bees. The food source in the population is represented by $X_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,n}\}$ and each solution in $x_{i,j}$ is generated as:

$$x_{i,j} = x_{min,j} + rand(0, 1) * (x_{max,j} - x_{min,j}) \quad (1)$$

where $i = 1, 2, \dots, FP$ and j denotes the number of variables in the solution space ranges from 1 to S and $x_{min,j}$ and $x_{max,j}$ is the dimension parameter S with upper and lower bounds.

The food sources are dispersed to the employee bees in random and its fitness value are evaluated. $rand(0, 1)$ is an equally scattered in range $[0, 1]$.

2.2 Employed Bee phase

Every employee bee $X_{i,j}$ gives modification on the food source position. If fitness value of the new position $v_{i,j}$ is better than existing, then the bee remembers the identified food source position and discards the existing food source. The invention of candidate position $v_{i,j}$ from the existing $X_{i,j}$ in memory is generated as:

$$v_{i,j} = x_{i,j} + \emptyset_{i,j}(x_{i,j} - x_{k,j}) \quad (2)$$

where $j = 1, 2, \dots, S$,

$k = 1, 2, \dots, FP$, chose the value of k is not equal to i ,

$\emptyset_{i,j}$ is a random value in the range $[-1, 1]$.

2.2.1 Onlooker Bee phase

In this phase, employee Bee shares the information about new food source X_i in the form of fitness value with onlooker bee. Each food source is evaluated based on probability P_{ij} and the selection of best a food source X_i is made by onlooker bee. The probability evaluation of food source is calculated by using Eq. (3)

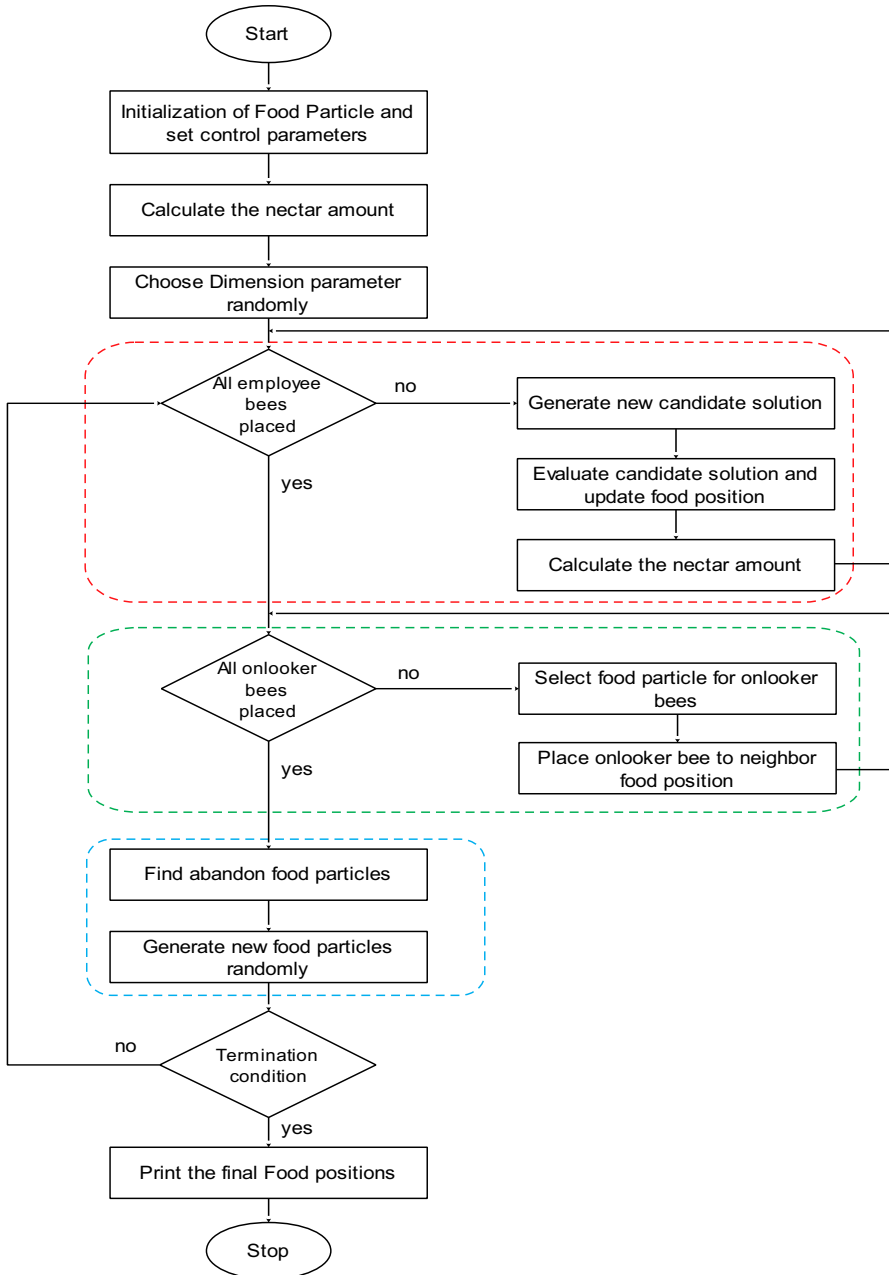


Fig. 1 Flowchart of ABC algorithm

$$P_{ij} = \frac{fit_i}{\sum_{j=1}^{FP} fit_j} \quad (3)$$

where fit_i is the fitness value of the solution i . Through this, employed bee will swap the information about food source with onlooker bees. If the candidate food source $v_{i,j}$ has better fitness than X_i , then $v_{i,j}$ food source will be replaced by X_i in the population.

2.2.2 Scout Bee phase

When the position of the food source X_i cannot enhance through a certain number of iterations, the particular food source is dilapidated. The number of iterations for the abandonment of food source is considered to be the important parameter in ABC algorithm.

Algorithm 1: ABC Algorithm

```

Initialization of population  $x_{i,j}$ , where  $i=1,2,\dots,FP, j=1,2,\dots, S$  .
Population fitness calculation
iteration =1
Repeat
{
  for  $i=1$  to  $FP$  do
    Produce new food solution for employed bees
    Greedy selection process on employee Bees
  end for
  Calculate Probability value for each solution  $x_{i,j}$ 
  if  $\text{random} < P_{ij}$  then
    Produce new food solution for onlooker bees
    Onlooker Bee Phase for  $x_{i,j}$  based on probability
  end if
  Perform Scout Bee Phase in place of abandoned food sources
  Enroll the best solution obtained so far
  iteration = iteration +1
}
Until iteration = Maximum number of Iteration

```

2.2.2.1 Revamped directed ABC algorithm (RDABC) ABC is used to solve global numerical optimization problems and engineering problems. ABC is the bio-inspired algorithm based on the behavior of ants, bees, fish, etc. ABC algorithm shows the advantage over other optimization algorithms by obtaining richness and robustness in solution. The algorithm is flexible to enhance and develop new algorithms, and the complexity is reduced since it requires fewer control parameters. To enhance the process of intensification and diversification, the authors have done three major changes in standard ABC algorithm. The pseudo code of proposed algorithm to solve global numerical optimization problems is given in Algorithm 2.

Algorithm 2: Pseudocode of RDABC

```

Initialization
Evaluate the fitness of the population
iteration = 1
Repeat
{
Employee Bee Phase
Probability Calculation Phase
Onlooker Bee Phase
Scout Bee Phase
Enroll the best solution obtained so far
iteration = iteration + 1
}
Until iteration = Maximum number of Iteration

```

At first, ABC generates randomly distributed population of food source as solutions, FP is the size of the population. In the initialization process, lower and upper bound parameters are randomly assigned to produce a feasible solution. In exploration phase, neighborhood search of 20% has been done to reconnoiter search space solution based on Eq. 4. The initialization process is explained in Algorithm 3.

Algorithm 3: (Initialization)

```

For i = 1, 2, ..., FP, do
  For j = 1, 2, ..., S, do
    Generate  $x_{i,j}$  solution
    
$$x_{i,j} = x_{min,j} \pm 0.2 * rand(0,1) * (x_{max,j} - x_{min,j}) \quad (4)$$

    Where  $x_{min,j}$  and  $x_{max,j}$  are the lower and upper bound of the dimension j
  End for
End for

```

After initialization process, in employed bee phase new food source is produced based on the local information extracted and evaluates the fitness value. The searchability and exploration of the solution is enhanced with the guidance of three searching mechanism; the first phase is finding the best in the neighborhood using $x_{i,j}$, $x_{k,j}$ and the perturbation on $x_{i,j}$ is decreased. The second phase is obtaining global best $x_{g,j}$ from all employed bees and evaluate the fitness on best solution fit_{best} and compared with other solutions. If fit_{best} is higher than other solution then memory is not updated else memory is updated. The third modification is in searching ability to calculate iteration best $x_{i,j}$ as iteration best food particle in dimension j and evaluate fitness on iteration best solution fit_{iter_best} . To increase the convergence rate, the food source position is directed based on random number $\xi_{i,j}$ and $\vartheta_{i,j}$. The direction information $\rho_{i,j}$ of the i th food source for j th dimension is added to achieve convergence rate. Initially the direction information $\rho_{i,j}$ is set to be 0, if the new candidate solution is higher than the old solution then $\rho_{i,j}$ is updated to value 1. If it is lesser, then the directional information $\rho_{i,j}$ is updated to -1 as shown in Eq. 5. To improve the ability of local search and convergence rate, the direction information $\rho_{i,j}$ is employed in the algorithm. Employed bee phase is illustrated in Algorithm 4.

Algorithm 4: (Employee Bee Phase)

```

For i = 1, 2, ..., FP, do
  For j = 1, 2, ..., S, do
    Produce candidate solution  $v_{i,j}$  for the employed bee of the solution  $x_{i,j}$ 

$$v_{i,j} = \begin{cases} x_{i,j} + \xi_{i,j} * \gamma_{i,j} & \text{if } \rho_{i,j} = 0 \\ x_{i,j} + \vartheta_{i,j} * \gamma_{i,j} & \text{if } \rho_{i,j} = 1 \\ x_{i,j} - \vartheta_{i,j} * \gamma_{i,j} & \text{if } \rho_{i,j} = -1 \end{cases} \quad (5)$$

    Where  $\gamma_{i,j} = (x_{i,j} - x_{k,j}) + fit_{best}(x_{g,j} - x_{i,j}) + fit_{iter\_best}(x_{t,j} - x_{i,j})$ 
    Where k is an integer chosen from 1 to FP and different from i and  $x_{best}$ .  $\xi_{i,j}$  and  $\vartheta_{i,j}$  are the uniformly distributed random real number produced in the range of [-1,1] and [0,1] respectively.  $fit_{best}$  is the fitness value of the best food particle with  $x_{g,j}$  as the global best food particle in dimension j.  $fit_{iter\_best}$  is the fitness value of the iteration best food particle with  $x_{t,j}$  as the iteration best food particle in dimension j.
    Evaluate  $fit(v_i)$ 
    if  $fit(v_i) < fit(x_i)$  then
       $x_i = v_i$ 
    else  $flag_i = flag_i + 1$ 
  End For
End For

```

Employed bees share information about the food source position with onlooker bee in the hive. Onlooker bees evaluate the fitness value based on probability calculation Pro_i based on Eq. 6. The probability of selecting the best food source is based on the fitness of the solution fit_i . The fit_i for the solution X_i can be calculated using Eq. 7.

Algorithm 5: (Probability Calculation Phase)

```

For i = 1, 2, ..., FP, do
  Calculate probability values  $Pro_i$  for the candidate solution  $v_{i,j}$ 

$$Pro_i = \frac{fit_i}{\sum_{j=1}^{FP} fit_j} \quad (6)$$


$$fit_i = \begin{cases} \frac{1}{1+fit_i}, & fit_i \geq 0 \\ 1 + abs(fit_i), & fit_i < 0 \end{cases} \quad (7)$$

End for

```

After calculating probability value Pro_i , onlooker bee randomly chooses food source with probability value from the employed bees. Food source positions were modified in onlooker bee phase upon probability value and evaluates with the random value (0, 1). If the food source has elevated probability value, then it is acceptable and proceeded to calculate candidate solution using Eq. 8. Like employed bee phase, the greedy selection of selecting the best solution among X_i and v_i is determined. The food sources which are not the virtue for exploiting are identified and abandoned through onlooker bee phase. The onlooker bee phase is ascertained by Algorithm 6.

Algorithm 6: (Onlooker Bee Phase)

```

For i = 1, 2, ..., FP, do
  For j = 1, 2, ..., S, do
    Set h=0
    While (h<=FP)
      If rand(0,1) <  $Pror_i$ 
        Produce candidate solution  $v_{i,j}$  for the onlooker bee of the food particle  $x_{i,j}$ 

$$v_{i,j} = \begin{cases} x_{i,j} + \xi_{i,j} * \gamma_{i,j} & \text{if } \rho_{i,j} = 0 \\ x_{i,j} + \vartheta_{i,j} * \gamma_{i,j} & \text{if } \rho_{i,j} = 1 \\ x_{i,j} - \vartheta_{i,j} * \gamma_{i,j} & \text{if } \rho_{i,j} = -1 \end{cases} \quad (8)$$

        Where  $\gamma_{i,j} = (x_{i,j} - x_{k,j}) + fit_{best}(x_{g,j} - x_{i,j}) + fit_{iter\_best}(x_{t,j} - x_{i,j})$ 
        evaluate  $fit(v_i)$ 
        if  $fit(v_i) < fit(x_i)$  then
           $x_i = v_i$ 
        else  $flag_i = flag_i + 1$ 
          h=h+1
        endif
      i=i+1
    end while
  End For
End for

```

However proposed algorithm increases the exploitation process by improving employee and onlooker bee phase than original ABC algorithm, the solution gets trapped in a local optimum. To avoid entrapment in local optimum, global search ability in scout bee phase is employed. The adjustable search radius to avoid getting entrapped in the local optimum is done by sigmoid function (Malik et al. 2007) in Eq. 9. The candidate solution $x_{i,j}$ of the scout bee is modified from abandoned old solution X_i . The value of $\psi_{initial}$ and ψ_{end} represents the initial and end inertia weight of range 0.9 and 0.4 respectively. The inertia values are static and desired by the experimenter. The scout bees specific search radius will decrease linearly in our proposed approach from 90 to 40% by sigmoid function at each round. For every iteration the solution will converge towards the optimal solution and the position of scout bee is adjusted in search space. The scout bee process is described in Algorithm 7.

Algorithm 7: (Scout Bee Phase)

```

For i = 1, 2, ..., FP, do
  
$$x_{i,j} = x_{min,j} + \text{rand}() * (x_{max,j} - x_{min,j})$$

  For j = 1, 2, ..., S, do
    Abandon the food source which cannot improve further

$$x_{i,j} = x_{min,j} \pm \Phi_{i,j} * \left( \frac{(\psi_{initial} - \psi_{end})}{(1 + e^{-q * (iteration - n * generation)})} + \psi_{end} \right) * (x_{max,j} - x_{min,j}) \quad (9)$$

    Where  $q = 10(\log(generation) - 2)$ 
    Where  $\psi_{initial}$  and  $\psi_{end}$  are the start and end weight of a given run with values 0.9 and 0.4 respectively.  $n$  is the sigmoid constant with value 0.75 to set partition of the function
    Compare old and new solution, replace old solution with new solution
  End for
End for

```

Thus, our proposed algorithm employs diversity in providing the new solution, and the convergence speed is increased by enhancing the process of both exploitation and exploration. Using proposed algorithm, the solutions are globally optimized compared to standard ABC algorithm. The authors have done three modifications,

at first in initialization process 20% of neighborhood search is done to enhance the exploitation process. Secondly, directed and search strategy is improved by considering neighbor best, global best and iteration best to produce new candidate solution. Through this, the process of exploitation is enhanced and convergence speed is increased. At last, to avoid entrapment in local optima, the exploration process is revamped by decreasing the search space radius from 90 to 40% using the sigmoid function. On imposing two different models namely ABC and Oppositional Learning would increase the computation time on every iteration. However, by fusing these two models, the proposed model achieves better results within the stipulated number of generations. Hence, by increasing the computational time of every iteration, the overall number of iterations gets reduced.

3 Experimental studies

3.1 Environmental setup and benchmark functions

The performance of DABC is investigated to minimize a set of 20 scalable benchmark functions of various unimodal/multimodal, separable/non-separable. The benchmark functions with various dimensions of 30, 60 and 100. In the first experiment, a set of 20 benchmark functions is demeanor to access the performance of DABC algorithm. In the second experiment, the performance of the proposed algorithm is investigated on a real-world engineering design optimization problem.

3.1.1 Mathematical Benchmark functions and parameter settings

The performance of the proposed RDABC algorithm is compared with ABC algorithm and its variants like SAMSABC (Sun et al. 2020), ABCNG (Xiao et al. 2021). RDABC is coded in MATLAB 12.0 platform under windows on an Intel 2 GHz core 2 quad processor with 2 GB of RAM. The proposed algorithm is to minimize 20 benchmark functions with dimensions D of 30, 60 and 100. For the lawful comparison, the maximum number of evaluations for all the algorithm is $100000 * D$. RDABC is run for 50 times for each mathematical benchmark functions. The algorithm aborts when the maximum numbers of evaluations are reached or when the corresponding global minimum value is reached. Table 1 summarizes 20 scalable benchmark functions with its characteristic and range. Results obtained by RDABC are compared with SAMSABC, ABCNG, and ABC concerning the optimal global solution, mean, median and standard deviation. The benchmark functions $f_1 - f_4$, f_7 , f_9 and f_{10} are unimodal functions, f_5 , f_6 , f_8 , $f_{11} - f_{20}$ are multimodal functions, f_{15} , f_{16} , are penalized functions, $f_1 - f_{15}$ and $f_{18} - f_{20}$ are separable functions and f_{16} and f_{17} are non-separable functions. Table 2.

On comparing the results of RDABC with the existing algorithms such as SAMSABC, ABCNG and conventional ABC, the proposed model outperforms the other algorithms in terms of achieving best results. On comparing the mean results, the proposed model achieves near values to best which in turn indicates that the final results include more number of optimal solutions. The SD shows the exploration

Table 1 Mathematical Benchmark Functions

Label	Name	Function	Range	Accept
F1	Sphere	$f_1(X) = \sum_{i=1}^n x_i^2$	$[-100, 100]^n$	1×10^{-8}
F2	Elliptic	$f_2(X) = \sum_{i=1}^n (10^{6i-1}) x_i^2$	$[-100, 100]^n$	1×10^{-8}
F3	SumSquare	$f_3(X) = \sum_{i=1}^n i x_i^2$	$[-10, 10]^n$	1×10^{-8}
F4	SumPower	$f_4(X) = \sum_{i=1}^n x_i ^{(i+1)}$	$[-1, 1]^n$	1×10^{-8}
F5	Schweffel 2.22	$f_5(X) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	$[-10, 10]^n$	1×10^{-8}
F6	Schweffel 2.21	$f_6(X) = \max \{ x_i , 1 \leq i \leq n\}$	$[-100, 100]^n$	1.5×10^1
F7	Step	$f_7(X) = \sum_{i=1}^n (x_i + 0.5)^2$	$[-100, 100]^n$	1×10^{-8}
F8	Exponential	$f_8(X) = \exp(0.5 * \sum_{i=1}^n x_i) - 1$	$[-1.28, 1.28]^n$	1×10^{-8}
F9	Quartic	$f_9(X) = \sum_{i=1}^n x_i^4 + \text{random}[0, 1)$	$[-1.28, 1.28]^n$	1×10^{-2}
F10	Rosenbrock	$f_{10}(X) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-5, 10]^n$	5×10^0
F11	Rastrigin	$f_{11}(X) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	$[-5.12, 5.12]^n$	1×10^{-8}
F12	NCRastrigin	$y_i = \begin{cases} x_i, & x_i < \frac{1}{2} \\ \text{round}(2x_i), & x_i \geq \frac{1}{2} \end{cases}$	$[-5.12, 5.12]^n$	1×10^{-8}
F13	Griewank	$f_{13}(X) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]^n$	1×10^{-8}

Table 1 (continued)

Label	Name	Function	Range	Accept
F14	Schwefel 2.26	$f_{14}(X) = 418.98288727243369 * n - \sum_{i=1}^n x_i \sin(\sqrt{ x_i })$	$[-500, 500]^n$	1×10^{-8}
F15	Ackley	$f_{15}(X) = 20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$ $f_{16}(X) = \frac{\pi}{n} \left\{ 10 \sin^2(\pi y_1) + \sum_{i=1}^{n-1} [(y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2] + \sum_{i=1}^n u(x_i, 10, 100, 4) \right\}$ $y_1 = 1 + \frac{1}{4}(x_1 + 1)$ $u_{x_i, a, k, m} = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & a \leq x_i \leq a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$	$[-32, 32]^n$	1×10^{-8}
F16	Penalized 1	$f_{17}(X) = \frac{1}{10} \left\{ \sin^2(\pi x_1) + \sum_{i=1}^{n-1} [(x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)^2 [1 + \sin^2(2\pi x_{i+1})] \right\} + \sum_{i=1}^n u(x_i, 5, 100, 4)$	$[-50, 50]^n$	1×10^{-8}
F17	Penalized 2	$f_{18}(X) = \sum_{i=1}^n x_i \sin(x_i) + 0.1 x_i $	$[-10, 10]^n$	1×10^{-8}
F18	Alpine	$f_{19}(X) = \sum_{i=1}^{n-1} (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + \sin^2(3\pi x_1) + x_n - 1 [1 + \sin^2(3\pi x_n)]$	$[-10, 10]^n$	1×10^{-8}
F19	Levy	$f_{20}(X) = \sum_{i=1}^D \left(\sum_{k=0}^{k_{max}} [a^k \cos(2\pi b^k (x_i + 0.5))] \right) - D \sum_{k=0}^{k_{max}} [d^k \cos(2\pi b^k 0.5)]$ $a = 0.5, b = 3, k_{max} = 20$	$[-0.5, 0.5]^n$	1×10^{-8}

Table 2 Simulation Results of Benchmark Functions with respect to Best, Median, Mean and Standard Deviation

Function	Dimension	RDABC			SAMSABC			ABCNG			ABC		
		30	60	100	30	60	100	30	60	100	30	60	100
F1	Best	1.31E-32	8.79E-30	4.17E-28	1.81E-32	1.03E-29	5.15E-28	8.50E-32	3.59E-29	6.24E-28	2.32E-10	3.64E-10	3.98E-10
	Median	1.37E-31	6.64E-29	1.90E-27	1.08E-31	5.67E-29	1.45E-27	1.47E-31	1.27E-28	2.13E-27	8.25E-10	1.53E-09	1.95E-09
	Mean	1.60E-31	7.13E-29	1.31E-27	1.29E-31	8.32E-29	1.87E-27	2.95E-32	2.38E-29	1.59E-28	6.42E-10	1.67E-09	1.39E-09
	SD	8.07E-32	5.25E-29	6.43E-28	8.49E-32	4.94E-29	6.62E-28	3.70E-32	3.15E-29	2.23E-28	3.04E-10	1.04E-09	8.19E-10
F2	Best	3.21E-29	3.32E-26	2.93E-24	2.17E-29	3.91E-26	4.01E-24	7.91E-28	2.24E-25	3.91E-24	3.25E-07	2.40E-07	1.40E-06
	Median	9.41E-28	3.93E-25	3.33E-24	1.16E-27	4.54E-25	3.95E-24	1.66E-27	7.59E-25	3.84E-24	9.95E-06	2.79E-06	4.42E-06
	Mean	1.07E-27	4.67E-25	3.31E-24	1.26E-27	3.88E-25	3.89E-24	1.18E-28	7.28E-26	6.22E-26	6.60E-06	3.11E-06	4.18E-06
	SD	6.73E-28	2.78E-25	2.49E-25	6.93E-28	2.53E-25	8.36E-26	1.62E-28	9.04E-26	6.85E-26	4.48E-06	1.86E-06	1.69E-06
F3	Best	3.17E-33	6.11E-30	1.26E-28	3.83E-33	5.29E-30	2.17E-28	8.52E-33	1.53E-29	5.39E-28	9.43E-12	8.76E-11	3.18E-10
	Median	3.59E-32	1.84E-29	5.82E-28	3.45E-32	2.21E-29	6.46E-28	2.36E-32	2.84E-29	7.71E-28	3.12E-11	1.76E-10	1.56E-09
	Mean	2.98E-32	1.91E-29	5.61E-28	3.33E-32	2.30E-29	5.56E-28	9.35E-33	3.13E-30	5.14E-29	3.39E-11	1.84E-10	1.81E-09
	SD	1.67E-32	9.97E-30	2.61E-28	1.36E-32	8.52E-30	2.44E-28	1.11E-32	3.99E-30	6.78E-29	1.26E-11	6.85E-11	8.25E-10
F4	Best	8.10E-75	7.36E-65	6.29E-52	3.68E-74	4.92E-65	9.50E-52	8.94E-69	1.31E-62	7.96E-49	1.39E-18	4.97E-11	3.16E-08
	Median	4.18E-69	3.59E-62	4.26E-48	4.19E-69	4.09E-62	4.43E-48	1.21E-68	6.46E-62	5.69E-48	2.90E-16	2.62E-10	1.35E-06
	Mean	6.85E-69	4.80E-62	4.64E-48	4.86E-69	4.73E-62	4.61E-48	6.52E-70	7.59E-63	2.10E-48	1.73E-16	4.26E-10	1.04E-06
	SD	4.65E-69	3.48E-62	2.41E-48	3.51E-69	3.51E-62	2.70E-48	8.83E-70	1.02E-62	2.43E-48	1.39E-16	1.86E-10	5.86E-07
F5	Best	7.16E-18	1.10E-16	1.09E-15	9.36E-18	1.27E-15	2.92E-15	1.96E-17	1.15E-15	6.47E-15	1.43E-06	5.98E-06	1.56E-06
	Median	2.70E-17	6.92E-16	3.01E-15	1.98E-17	1.12E-15	4.57E-15	3.58E-17	2.94E-14	6.54E-15	2.28E-06	8.23E-06	7.21E-06
	Mean	2.73E-17	4.74E-16	2.92E-15	2.00E-17	1.08E-15	4.71E-15	4.74E-18	5.77E-15	2.20E-17	1.94E-06	7.58E-06	1.08E-05
	SD	1.18E-17	2.60E-16	1.62E-15	8.74E-18	1.45E-16	1.12E-15	6.35E-18	8.57E-15	3.27E-17	4.21E-07	1.16E-06	3.63E-06
F6	Best	8.01E+00	1.20E-02	9.10E-02	9.52E+00	3.12E-01	7.10E-01	1.76E+00	3.70E-01	7.01E-01	2.00E-01	3.84E-01	5.90E-01
	Median	4.43E+00	1.47E-01	4.12E-01	5.33E+00	3.64E-01	6.58E-01	6.57E-01	3.92E-01	6.71E-01	2.04E-01	4.46E-01	6.05E-01
	Mean	3.49E+00	2.01E-01	3.88E-01	4.91E+00	3.68E-01	6.62E-01	8.83E-01	7.50E-03	1.37E-02	2.03E-01	4.23E-01	6.04E-01
	SD	2.62E+00	1.23E-01	1.82E-01	3.26E+00	2.54E-02	3.05E-02	4.48E-01	9.37E-03	1.65E-02	2.42E-03	2.50E-02	7.85E-03

Table 2 (continued)

Function	Dimension	RDABC			SAMSABC			ABCNG			ABC		
		30	60	100	30	60	100	30	60	100	30	60	100
F7	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Median	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F8	SD	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	Best	6.93E-69	3.18E-64	1.82E-61	9.28E-69	4.90E-64	4.16E-61	3.54E-67	8.20E-62	3.30E-60	1.39E-29	5.60E-28	1.72E-26
	Median	1.53E-67	1.40E-61	9.66E-60	4.16E-67	1.13E-61	6.95E-60	5.16E-67	2.06E-61	8.63E-60	1.04E-28	1.06E-26	6.40E-26
F9	Mean	3.22E-67	1.22E-61	7.57E-60	2.42E-67	1.33E-61	7.87E-60	6.23E-68	3.05E-62	2.88E-60	6.76E-29	1.00E-26	7.57E-26
	SD	1.97E-67	7.51E-62	4.34E-60	1.86E-67	1.32E-61	7.86E-62	4.91E-60	3.73E-68	3.66E-62	3.44E-60	5.04E-29	5.58E-27
	Best	2.90E-03	1.50E-02	9.20E-02	2.57E-02	1.32E-01	1.89E-01	2.61E-02	1.32E-01	1.33E-01	6.49E-02	1.94E-01	4.03E-01
F10	Median	2.42E-02	7.80E-02	1.60E-01	3.98E-02	1.33E-01	2.33E-01	3.16E-02	1.33E-01	2.55E-01	1.07E-01	2.21E-01	4.64E-01
	Mean	2.34E-02	6.65E-02	1.67E-01	3.12E-02	1.33E-01	2.27E-01	3.07E-03	8.03E-05	8.74E-04	8.91E-02	2.32E-01	4.53E-01
	SD	1.51E-02	3.77E-02	4.80E-02	7.03E-03	3.30E-04	2.40E-02	3.43E-03	9.98E-05	1.19E-03	1.96E-02	2.00E-02	3.15E-02
F11	Best	2.51E-02	2.17E-01	3.82E-01	4.59E-02	2.72E-01	5.79E-01	1.27E+00	4.67E+00	5.68E+00	2.34E-02	1.97E-01	5.60E-01
	Median	1.18E+00	2.96E+00	3.77E+00	9.43E-01	3.20E+00	4.68E+00	1.57E+00	5.09E+00	6.32E+00	1.61E+00	3.45E+00	1.94E+00
	Mean	9.93E-01	3.02E+00	3.02E+00	7.63E-01	2.72E+00	4.29E+00	1.77E-01	9.62E-02	3.48E-01	7.38E-01	2.67E+00	2.14E+00
F12	SD	5.75E-01	1.57E+00	2.40E+00	6.19E-01	1.79E+00	2.45E+00	2.24E-01	1.18E-01	5.23E-01	7.75E-01	1.87E+00	1.02E+00
	Best	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	6.63E-01	9.03E-01	3.63E-10	3.34E-10	7.50E-09
	Median	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.01E-01	1.19E+00	1.42E+00	9.49E-02	1.11E+00	8.47E-01
F12	Mean	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	2.57E-02	2.50E-01	1.69E-01	8.10E-02	1.37E+00	9.25E-01
	SD	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	2.95E-02	3.04E-01	2.22E-01	5.02E-02	7.61E-01	7.16E-01
	Best	0.00E+00	0.00E+00	1.24E-01	0.00E+00	0.00E+00	0.00E+00	9.12E-01	2.59E+00	6.79E+00	9.03E-10	2.32E-07	2.24E+00
F12	Median	0.00E+00	0.00E+00	2.04E+00	0.00E+00	0.00E+00	0.00E+00	9.50E-01	2.95E+00	7.00E+00	5.78E-01	1.50E+00	5.35E+00
	Mean	0.00E+00	0.00E+00	3.05E+00	0.00E+00	0.00E+00	0.00E+00	1.73E-02	6.40E-02	8.20E-02	5.72E-01	9.07E-01	4.42E+00
	SD	0.00E+00	0.00E+00	1.59E+00	0.00E+00	0.00E+00	0.00E+00	1.99E-02	7.68E-02	9.74E-02	2.83E-01	9.60E-01	1.35E+00

Table 2 (continued)

Function	Dimension	RDABC			SAMSABC			ABCNG			ABC			
		30	60	100	30	60	100	30	60	100	30	60	100	
F13	Best	1.90E-13	7.40E-13	1.30E-14	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.13E-08	2.10E-10	2.31E-08	4.94E-12	8.04E-12	6.43E-13
	Median	1.77E-08	3.76E-10	5.39E-09	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.11E-07	7.08E-10	2.64E-08	3.71E-08	5.82E-10	2.32E-08
	Mean	2.12E-08	5.91E-10	4.20E-09	0.00E+00	0.00E+00	0.00E+00	0.00E+00	3.23E-08	2.29E-10	1.59E-09	7.09E-08	4.29E-10	1.38E-08
F14	SD	1.06E-08	2.41E-10	3.31E-09	0.00E+00	0.00E+00	0.00E+00	0.00E+00	3.74E-08	2.74E-10	1.92E-09	4.81E-08	3.02E-10	1.20E-08
	Best	9.10E-13	1.75E-11	8.90E-11	2.08E-12	3.07E-11	1.62E-10	1.62E-10	8.98E-13	3.47E-11	1.56E-10	1.70E-06	3.61E-02	7.92E-02
	Median	1.29E-03	2.49E-11	1.02E-10	1.08E-12	3.29E-11	1.35E-10	1.35E-10	9.54E-11	3.49E-11	1.49E-10	1.38E-02	5.17E-02	4.02E-02
F15	Mean	1.66E-03	2.67E-11	1.06E-10	1.16E-12	3.39E-11	1.45E-10	1.45E-10	2.76E-11	4.55E-13	8.82E-12	1.00E-02	5.10E-02	4.94E-02
	SD	1.11E-03	6.81E-12	1.17E-11	6.36E-13	1.88E-12	1.34E-11	1.34E-11	3.54E-11	5.77E-13	9.57E-12	7.73E-03	1.31E-02	2.34E-02
	Best	2.50E-14	9.30E-14	2.17E-13	4.11E-14	1.93E-13	3.84E-13	3.84E-13	4.28E-14	1.61E-13	3.96E-13	2.40E-06	2.52E-06	5.21E-06
F16	Median	3.12E-14	1.27E-13	3.22E-13	4.20E-14	1.74E-13	3.88E-13	3.88E-13	4.34E-14	1.59E-13	3.97E-13	5.59E-06	1.08E-05	8.93E-06
	Mean	3.53E-14	1.23E-13	3.32E-13	4.19E-14	1.76E-13	3.92E-13	3.92E-13	1.20E-16	6.63E-16	5.38E-16	5.66E-06	9.11E-06	9.59E-06
	SD	6.49E-15	1.86E-14	6.45E-14	6.66E-16	1.12E-14	4.34E-15	4.34E-15	1.78E-16	7.61E-16	6.50E-16	2.33E-06	4.45E-06	3.20E-06
F17	Best	6.90E-33	7.50E-32	7.80E-31	1.72E-32	1.86E-31	1.82E-30	1.82E-30	2.11E-32	9.67E-31	2.39E-30	7.90E-12	2.81E-11	3.84E-11
	Median	2.06E-32	5.04E-31	2.25E-30	1.91E-32	6.03E-31	2.58E-30	2.58E-30	2.31E-32	1.06E-30	2.81E-30	1.37E-11	4.90E-11	8.96E-11
	Mean	1.89E-32	8.30E-31	2.08E-30	2.23E-32	6.86E-31	2.50E-30	2.50E-30	1.08E-33	3.06E-32	1.22E-31	1.34E-11	6.59E-11	1.38E-10
F18	SD	6.86E-33	3.32E-31	8.18E-31	3.23E-33	3.48E-31	4.25E-31	4.25E-31	1.36E-33	3.84E-32	1.48E-31	3.43E-12	2.02E-11	4.89E-11
	Best	4.84E-32	4.30E-30	9.20E-29	6.31E-32	1.98E-29	1.26E-28	1.26E-28	2.46E-31	3.55E-29	2.90E-28	3.91E-10	9.29E-10	2.51E-11
	Median	2.44E-31	5.08E-29	2.16E-28	1.59E-31	4.67E-29	1.76E-28	1.76E-28	3.71E-31	5.19E-29	2.93E-28	8.83E-10	3.72E-09	1.47E-10
F18	Mean	1.98E-31	3.31E-29	1.57E-28	2.53E-31	4.31E-29	2.11E-28	2.11E-28	3.77E-32	7.29E-30	4.56E-31	9.10E-10	4.81E-09	1.23E-10
	SD	1.16E-31	2.05E-29	5.92E-29	1.25E-31	2.27E-29	5.41E-29	5.41E-29	5.04E-32	8.93E-30	5.45E-31	3.98E-10	2.39E-09	5.55E-11
	Best	2.26E-18	3.01E-16	8.80E-16	4.01E-18	9.50E-17	2.89E-15	2.89E-15	2.40E-16	1.28E-15	4.77E-15	3.09E-05	2.27E-04	7.26E-04
F18	Median	2.96E-16	1.12E-15	5.64E-15	1.39E-16	9.18E-16	6.36E-15	6.36E-15	5.72E-16	1.23E-16	5.96E-16	7.23E-05	5.80E-04	8.58E-03
	Mean	3.91E-16	9.23E-16	5.21E-15	2.89E-16	1.14E-15	6.68E-15	6.68E-15	6.20E-17	1.23E-16	5.96E-16	7.17E-05	7.19E-04	6.79E-03
	SD	2.40E-16	6.09E-16	3.13E-15	2.19E-16	5.17E-16	2.21E-15	2.21E-15	8.07E-17	1.57E-16	7.04E-16	2.44E-05	3.20E-04	3.82E-03

Table 2 (continued)

Function	Dimension	RDABC			SAMSABC			ABCNG			ABC		
		30	60	100	30	60	100	30	60	100	30	60	100
F19	Best	9.30E-32	8.32E-31	3.16E-29	1.84E-31	1.08E-30	4.31E-29	2.06E-31	6.38E-30	8.99E-29	1.34E-10	1.70E-10	1.77E-10
	Median	1.45E-31	3.76E-30	6.31E-29	1.98E-31	4.53E-30	7.54E-29	2.07E-31	7.58E-30	1.30E-28	5.02E-10	2.80E-09	2.08E-09
	Mean	1.53E-31	6.17E-30	7.83E-29	1.94E-31	5.12E-30	8.30E-29	4.49E-34	6.64E-31	1.42E-29	4.33E-10	1.84E-09	1.85E-09
	SD	2.72E-32	2.21E-30	4.29E-29	8.29E-33	2.23E-30	3.76E-29	5.73E-34	7.70E-31	1.74E-29	2.53E-10	1.45E-09	1.16E-09
F20	Best	0.00E+00	0.00E+00	5.68E-14	0.00E+00	0.00E+00	6.21E-14	0.00E+00	0.00E+00	5.75E-14	1.45E-01	1.93E-01	6.87E+00
	Median	5.57E-10	4.13E-11	4.85E-14	0.00E+00	0.00E+00	5.18E-14	4.17E-03	2.94E-01	5.28E-14	1.52E-01	2.56E-01	7.13E+00
	Mean	6.28E-10	3.02E-11	4.98E-14	0.00E+00	0.00E+00	5.46E-14	1.35E-03	8.92E-02	3.52E-15	1.51E-01	2.76E-01	7.29E+00
	SD	4.25E-10	1.99E-11	4.34E-15	0.00E+00	0.00E+00	7.06E-15	1.78E-03	1.06E-01	3.98E-15	5.14E-03	5.10E-02	1.70E-01

capability of all algorithms. And in terms of computational time as it is mentioned in Sect. 4, the model uses less iterations and thus the computational time is lesser when compared with ABC, SAMSABC and ABCNG Table 3.

3.1.2 Statistical test

Wilcoxon signed-rank nonparametric statistical test is used to find the significant difference and behavior of experimental methods. This tool is used to measure the statistical performance of algorithm in pair-wise. The test results are shown in Tables 4, 5 and 6 and the result is performed with significance value $\alpha = 0.05$. There is no difference between first algorithms median and next algorithm for the same functions. Wilcoxon Signed rank test provide sizes for the ranks like $T+$, $T-$ to know the best rank among two algorithms (Civicioglu 2013). The first column in the table shows benchmark functions to evaluate the performance of experimental methods. The second column indicates p-value. The third and the fourth column indicate the rank value of the pair algorithm. The sign + indicates the proposed RDABC is superior in Wilcoxon signed-Rank test and null hypothesis is rejected. The sign - indicates the proposed RDABC is inferior performance in Wilcoxon test and null hypothesis is rejected. The sign = shows no significant difference among two algorithms. The Wilcoxon Signed-Rank test is carried out with comparative algorithms for smaller, medium and larger dimensions.

Table 4, 5, and 6 shows statistical result of Wilcoxon Signed-Rank Test compared with SAMSABC, ABCNG, and ABC for small, medium, and large dimensions through 50 runs for each mathematical benchmark functions. '+' sign indicates that null hypothesis is rejected and RDABC shows superior performance over other algorithms. '-' sign indicates that null hypothesis is rejected and RDABC shows inferior performance over other algorithms. '=' sign indicates that there is no significant different among the two algorithms in solving benchmark problems. Table 7 shows the count of statistical significance of RDABC with other algorithms.

Table 7 shows the comparison of the proposed algorithm with SAMSABC, ABCNG and ABC algorithm for smaller, medium, and larger dimension problems. Pair-wise comparison of proposed RDABC with SAMSABC for 30-dimensional benchmark functions rejects the null hypothesis with superior performance for 60% of functions, 15% of functions with no significant difference and 25% of functions with inferior performance. For 60-dimensional benchmark problems, RDABC rejects the null hypothesis with a same performance like 30-dimensional functions. For larger dimension, RDABC rejects the null hypothesis with 75% of superior performance, 10% of functions with no significant difference and 15% of functions with inferior performance.

Pair-wise comparison of RDABC with ABCNG rejects the null hypothesis for all dimensional problem with no significant differences. For 30-dimension problem, 85% of functions possess superior performance, 15% of functions with inferior performance than ABCNG. All the medium sized functions possess superior performance with null hypothesis rejection. For 100-dimension function, 95% of the functions achieved superior performance and 5% of the functions with inferior performance. Pair-wise comparison of the proposed algorithm with standard ABC

Table 3 Computational CPU Time in Seconds

Function	Dimension = 30				Dimension = 60				Dimension = 100			
	RDABC	SAMSABC	ABCNG	ABC	RDABC	SAMSABC	ABCNG	ABC	RDABC	SAMSABC	ABCNG	ABC
	F1	0.13	0.15	0.23	0.18	0.22	0.47	0.4	0.31	0.62	0.87	0.8
F2	0.45	0.5	0.6	0.51	1.21	1.3	1.41	1.32	2.25	2.4	2.45	2.36
F3	0.1	0.13	0.21	0.16	0.21	0.25	0.4	0.27	0.61	0.65	0.8	0.67
F4	0.41	0.49	0.57	0.5	1.1	1.15	1.33	1.14	2.4	2.71	2.63	2.44
F5	0.12	0.15	0.23	0.16	0.25	0.47	0.43	0.4	0.85	1.07	1.03	1
F6	0.1	0.16	0.23	0.17	0.21	0.41	0.41	0.49	0.91	1.11	1.26	1.19
F7	0.1	0.15	0.23	0.18	0.22	0.35	0.46	0.4	1.02	1.15	1.26	1.2
F8	0.1	0.14	0.22	0.16	0.23	0.26	0.48	0.26	0.95	0.99	1.2	0.98
F9	0.42	0.5	0.58	0.51	1.16	1.23	1.35	1.25	2.13	2.3	2.32	2.22
F10	0.11	0.16	0.23	0.17	0.23	0.32	0.39	0.43	0.79	0.88	0.95	0.99
F11	0.28	0.3	0.39	0.34	0.65	0.71	0.89	0.73	1.28	1.34	1.52	1.36
F12	0.21	0.25	0.34	0.32	0.59	0.54	0.7	0.65	1.28	1.23	1.39	1.34
F13	0.05	0.38	0.47	0.41	0.73	0.86	1.02	0.87	1.46	1.59	1.75	1.6
F14	0.33	0.34	0.43	0.36	0.78	0.82	1.06	0.83	1.69	1.73	1.97	1.74
F15	0.28	0.32	0.42	0.35	0.63	0.73	0.92	0.74	1.25	1.35	1.54	1.36
F16	0.28	0.36	0.46	0.39	0.68	0.83	1.03	0.85	1.51	1.66	1.86	1.68
F17	0.29	0.34	0.44	0.4	0.7	0.84	1.06	0.87	1.57	1.71	1.93	1.74
F18	0.22	0.28	0.37	0.31	0.52	0.62	0.81	0.65	1.21	1.31	1.5	1.34
F19	0.28	0.35	0.43	0.38	0.7	0.79	0.99	0.78	1.53	1.62	1.82	1.61
F20	28.27	34.17	31.07	33.27	88.25	93.46	94.86	91.26	139.2	144.41	145.81	142.21

Table 4 Wilcoxon Signed-Rank test based on mean value of benchmark functions for various dimension RDABC VS SAMSABC

Functions	30					60					100				
	p-value	T+	T-	Winner		p-value	T+	T-	Winner		p-value	T+	T-	Winner	
	F1	1.99E-01	170	295	+		9.75E-01	231	234	+		8.45E-01	242	223	-
F2	1.02E-01	153	312	+		1.31E-01	159	306	+		1.73E-06	0	465	+	
F3	8.45E-01	223	242	+		7.34E-01	216	249	+		1.71E-01	166	299	+	
F4	8.61E-01	224	241	+		7.34E-01	216	249	+		2.54E-01	288	177	-	
F5	7.81E-01	219	246	+		1.73E-06	0	465	+		9.63E-04	72	393	+	
F6	3.39E-01	186	279	+		2.60E-05	28	437	+		1.73E-06	0	465	+	
F7	1	0	0	=		1	0	0	=		1	0	0	=	
F8	3.09E-01	183	282	+		1.85E-01	297	168	-		7.81E-01	219	246	+	
F9	2.22E-04	53	412	+		1.92E-06	1	464	+		1.11E-03	74	391	+	
F10	3.49E-01	278	187	-		3.93E-01	274	191	-		6.87E-02	144	321	+	
F11	1	0	0	=		1	0	0	=		1	0	0	=	
F12	1	0	0	=		1	0	0	=		1.73E-06	465	0	-	
F13	1.73E-06	465	0	-		1.73E-06	465	0	-		1.73E-06	465	0	-	
F14	1.73E-06	465	0	-		1.60E-04	49	416	+		1.73E-06	0	465	+	
F15	4.73E-06	10	455	+		1.73E-06	0	465	+		1.73E-06	0	465	+	
F16	1.74E-04	50	415	+		5.72E-01	260	205	-		1.32E-02	112	353	+	
F17	8.13E-01	244	221	-		3.82E-01	180	275	+		2.29E-01	174	291	+	
F18	3.82E-01	190	275	+		7.19E-01	215	250	+		1.11E-01	155	310	+	
F19	2.13E-06	2	463	+		6.14E-01	208	257	+		5.44E-01	203	262	+	
F20	1.73E-06	465	0	-		1.73E-06	465	0	-		4.68E-03	95	370	+	

Table 5 Wilcoxon Signed-Rank test based on mean value of benchmark functions for various dimension RDABC VS ABCNG

Functions	30						60						100					
	p-value		T+	T-	Winner	Winner	p-value		T+	T-	Winner	Winner	p-value		T+	T-	Winner	
F1	4.39E-03	94	371	+	+	+	2.29E-01	174	291	+	+	+	8.61E-01	224	241	+		
F2	2.16E-05	26	439	+	+	+	4.90E-04	63	402	+	+	+	1.73E-06	0	465	+		
F3	4.17E-01	272	193	-	-	-	5.32E-03	97	368	+	+	+	4.07E-05	33	432	+		
F4	1.15E-04	45	420	+	+	+	6.04E-03	99	366	+	+	+	2.70E-02	340	125	-		
F5	6.04E-03	99	366	+	+	+	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+		
F6	3.41E-05	434	31	-	-	-	2.13E-06	2	463	+	+	+	1.73E-06	0	465	+		
F7	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+		
F8	5.79E-05	37	428	+	+	+	8.22E-02	148	317	+	+	+	5.19E-02	138	327	+		
F9	6.32E-05	38	427	+	+	+	1.92E-06	1	464	+	+	+	1.92E-06	1	464	+		
F10	1.36E-04	47	418	+	+	+	1.73E-06	0	465	+	+	+	3.52E-06	7	458	+		
F11	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+		
F12	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+		
F13	1.49E-05	22	443	+	+	+	7.86E-02	147	318	+	+	+	1.73E-06	0	465	+		
F14	1.73E-06	465	0	-	-	-	3.18E-06	6	459	+	+	+	1.73E-06	0	465	+		
F15	3.18E-06	6	459	+	+	+	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+		
F16	1.02E-05	18	447	+	+	+	1.97E-05	25	440	+	+	+	1.24E-05	20	445	+		
F17	9.63E-04	72	393	+	+	+	1.59E-03	79	386	+	+	+	1.73E-06	0	465	+		
F18	1.11E-03	74	391	+	+	+	4.73E-06	10	455	+	+	+	3.33E-02	129	336	+		
F19	1.73E-06	0	465	+	+	+	8.19E-05	41	424	+	+	+	9.71E-05	43	422	+		
F20	1.73E-06	0	465	+	+	+	1.73E-06	0	465	+	+	+	5.04E-01	200	265	+		

Table 6 Wilcoxon Signed-Rank test based on mean value of benchmark functions for various dimension RDABC VS ABC

Functions	30						60						100					
	p-value		T+	T-	Winner		p-value		T+	T-	Winner		p-value		T+	T-	Winner	
F1	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F2	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F3	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F4	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F5	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F6	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F7	1	0	0	0	=		1	0	0	0	=		1	0	0	0	=	
F8	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F9	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F10	6.29E-01	209	256	256	+		2.37E-01	290	175	175	-		2.18E-02	344	121	121	-	
F11	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F12	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F13	1.80E-05	24	441	441	+		9.59E-01	235	230	230	-		1.06E-04	44	421	421	+	
F14	6.98E-06	14	451	451	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F15	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F16	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F17	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F18	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F19	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	
F20	1.73E-06	0	465	465	+		1.73E-06	0	465	465	+		1.73E-06	0	465	465	+	

Table 7 Pair-wise comparison of proposed algorithm with other algorithms for various dimensions

Algorithm comparison	+ / = / -		
	30	60	100
RDABC Vs. SAMSABC	12/3/5	12/3/5	15/2/3
RDABC Vs. ABCNG	17/0/3	20/0/0	19/0/1
RDABC Vs ABC	19/1/0	17/1/2	18/1/1

algorithm rejects the null hypothesis for all functions. For 30-dimensional problem, 95% of the benchmark functions attained superior performance, 5% of functions with significant different and no inferior performance. For medium sized functions, 85% of functions attained superior performance, significant difference for 5% of functions and 10% of functions with inferior performance. For 100-dimension functions, 90% of functions achieved superior performance, 5% of functions attained inferior performance and 5% of functions had significant difference.

3.2 CHPED

CHPED problem is to minimize the production cost of the system by satisfying heat, power demands and constraints. The CHPED is to determine the generation of unit power system and production of heat system. The objective function is to minimize the cost, which can be represented mathematically by,

$$F(p, h) = \text{Minimize} \sum_{i=1}^{n_p} C_i(P_i) + \sum_{j=1}^{n_c} C_j(P_j, H_j) + \sum_{k=1}^{n_h} C_k(H_k)$$

Subject to.

Equality constraints

$$\sum_{i=1}^{n_p} P_i + \sum_{j=1}^{n_c} P_j = P_d + P_L$$

$$\sum_{k=1}^{n_h} H_k + \sum_{j=1}^{n_c} H_j = H_d$$

$$P_i^{min} \leq P_i \leq P_i^{max}, i = 1, 2, \dots, n_p$$

$$P_j^{min}(H_j) \leq P_j \leq P_j^{max}(H_j), j = 1, 2, \dots, n_c$$

$$H_j^{min}(P_j) \leq H_j \leq H_j^{max}(P_j), j = 1, 2, \dots, n_c$$

$$H_k^{\min} \leq H_k \leq H_k^{\max}, k = 1, 2, \dots, n_h$$

The total heat and power production of the problem $F(p, h)$ is to minimize the cost. P is the unit power generation, h is the unit heat production; n_p , n_c and n_h are the number of conventional power unit, co-generation unit and heat-only units respectively. $C_i(P_i)$, $C_j(P_j, H_j)$ and $C_k(H_k)$ are the cost functions of the i th conventional power unit, j th co-generation unit and k th heat-only unit. P^{\min} , P^{\max} , H^{\min} and H^{\max} are the limits of the unit power capacity and unit heat capacity. The active power transmission loss P_L , can be calculated using Eq ():

$$P_L = \sum_{i=1}^{n_c} \sum_{j=1}^{n_c} P_i P C_{i,j} P_j$$

where $PC_{i,j}$ is the loss coefficient connected between i and j . In CHPED problem, the power generations P_i , P_j , and heat productions H_j , H_k are taken as decision variables. The cost function for the power generations is represented by a quadratic function.

Cost function of each unit

Power-only units

$$F_{t1}(P_1) = 25 + 2P_1 + 0.008P_1^2 + \left| 100 \sin \{0.042(P_1^{\min} - P_1)\} \right|$$

where $10 \leq P_1 \leq 75$

$$F_{t2}(P_2) = 60 + 1.8P_2 + 0.003P_2^2 + \left| 140 \sin \{0.04(P_2^{\min} - P_2)\} \right|$$

where $20 \leq P_2 \leq 125$

$$F_{t3}(P_3) = 100 + 2.1P_3 + 0.0012P_3^2 + \left| 160 \sin \{0.038(P_3^{\min} - P_3)\} \right|$$

where $30 \leq P_3 \leq 175$

$$F_{t4}(P_4) = 120 + 2P_4 + 0.001P_4^2 + \left| 180 \sin \{0.037(P_4^{\min} - P_4)\} \right|$$

where $40 \leq P_4 \leq 250$.

Cogeneration units

$$F_{c5}(P_5, H_5) = 2650 + 14.5P_5 + 0.0345P_5^2 + 4.2H_5 + 0.03H_5^2 + 0.031P_5H_5$$

$$F_{c6}(P_6, H_6) = 1250 + 36P_6 + 0.0435P_6^2 + 0.6H_6 + 0.027H_6^2 + 0.11P_6H_6$$

Heat-only unit

$$F_{h7}(H_7) = 950 + 2.0109H_7 + 0.038H_7^2$$

where $0 \leq H_7 \leq 2695.2$

Table 8 refers the results of combined head and power dispatch with recent algorithms that are used to solve the problem. From the table it is evident that the proposed RDABC outperforms other existing algorithms in terms of overall cost within the given constrained of heat and power. However, on comparing the loss obtained RDABC significance is lower. Network loss coefficients and the feasible region are cited in the Appendix section.

4 Conclusion

In this paper, Revamped Directed ABC Algorithm is proposed for addressing non-linear continuous problems with improved search strategy in the phases of exploration and exploitation. The proposed algorithm has been tested in two different folds namely 1. Standard mathematical benchmark instances and 2. CHPED problem. The test beds are chosen in such a way to prove the significance of the algorithm in both unconstrained and constrained optimization problems. On comparing the results of the proposed algorithm, it shows the significance in terms of convergence towards optimal solution with less computational time. Statistical test namely Wilcoxon Signed Rank Test also concludes that the proposed method wins the pairwise comparison in most of the benchmark instances. On comparing the results of constrained optimization problem namely CHPED, it chooses 5 power values and three heat values in a efficient manner that reduces the overall cost without violating the heat demand constraint. The overall significance of the proposed algorithm are compared with most recent and also conventional algorithms to prove its significance and thus the results shows the significance of proposed algorithm. The future work of the paper is to solve more Engineering problems which are with constraint non-linear search space.

Table 8 Result comparison of rdabc vs other existing algorithms

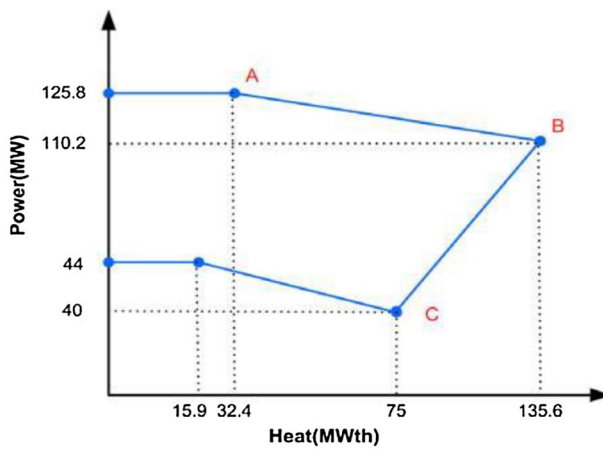
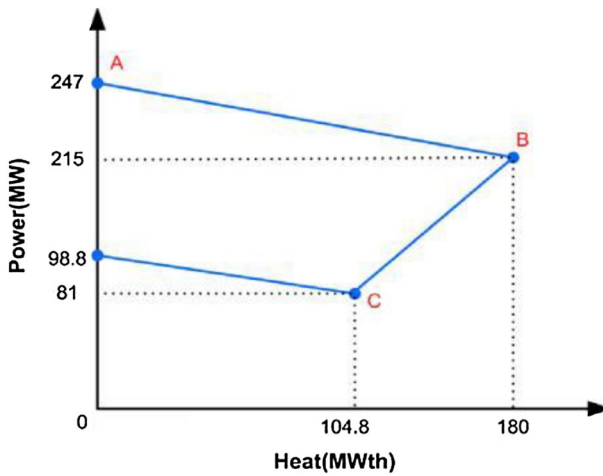
	CPSO (Mohammadi Ivatloo et al. 2013)	TVAC-PSO (Mohammadi Ivatloo et al. 2013)	RCGA (Basu 2011)	BCO (Basu 2011)	EP (Basu 2010)	DE (Basu 2010)	PSO (Basu 2010)	GSA (Beigvand et al. 2016)	MPSO (Neyestani et al. 2019)	AMPSO (Neyestani et al. 2019)	RDABC
P1	75	47.3383	74.6834	43.9457	61.3610	44.2118	18.4626	48.7638	75	45.6843	35.1502
P2	112.3800	98.5398	97.9578	98.5888	95.1205	98.5383	124.2602	98.7469	98.5063	98.5398	103.6527
P3	30	112.6735	167.2308	112.9320	99.9427	122.6913	112.7794	112	112.7066	112.6735	144.4703
P4	250	209.8158	124.9079	209.7719	208.7319	209.7741	209.8158	208.5113	209.8598	209.8158	213.9846
P5	93.2701	92.3718	98.8008	98.8000	98.8000	98.8217	98.8140	92.6909	93.4525	93.9903	63.3112
P6	40.1585	40.0000	44.0001	44	44	44	44.0107	40	40.0442	40	46.8526
H5	32.5655	37.8467	58.0965	12.0974	18.0713	12.5379	57.9236	35.9704	31.3642	28.3178	5.0192
H6	72.6738	74.9999	32.4116	78.0236	77.5548	78.3481	32.7603	75	74.3473	75	64.6829
H7	44.7606	37.1532	59.4919	59.8790	54.3739	59.1139	59.3161	39	44.2891	46.6822	80.2979
Loss	0.6395	0.7033	0.7258	0.7213	0.7132	0.7486	0.7284	0.7030	0.7674	0.7037	7.4217
Power	600.8086	600.7392	607.5808	608.0384	607.9561	618.0372	608.1427	600.7129	629.5693	600.7037	607.4217
Heat	149.9999	149.9998	150	150	150	149.9999	150	149.9704	150.0006	150	150
Cost	10.3255.3314	10.100.3147	10.667.4027	10.317.4995	10.419.9101	10.399.8284	10.612.9247	10.112.4514	10.122.8073	10.094.1098	10.010.6939

Appendix

Network Loss coefficients

$$B = \begin{bmatrix} 49 & 14 & 15 & 15 & 20 & 25 \\ 14 & 45 & 16 & 20 & 18 & 19 \\ 15 & 16 & 39 & 10 & 12 & 15 \\ 15 & 20 & 10 & 40 & 14 & 11 \\ 20 & 18 & 12 & 14 & 35 & 17 \\ 25 & 19 & 15 & 11 & 17 & 39 \end{bmatrix} \times 10^{-7}$$

See Figs. 2, 3.



Acknowledgements This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2019R1G1A110034111).

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

- Aderhold, A., Diwold, K., Scheidler, A., Middendorf, M.: Artificial bee colony optimization: a new selection scheme and its performance. In: *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, pp. 283–294. Springer, Berlin, Heidelberg (2010)
- Akay, B., Karaboga, D.: A modified artificial bee colony algorithm for real-parameter optimization. *Inform Sci* **192**, 120–142 (2012)
- Akbari, R., Mohammadi, A., Ziarati, K.: A novel bee swarm optimization algorithm for numerical function optimization. *Commun. Nonlinear Sci. Num. Simul.* **15**(10), 3142–3155 (2010)
- Anuar, S., Selamat, A., Sallehuddin, R.: A modified scout bee for artificial bee colony algorithm and its performance on optimization problems. *J. King Saud Univ-Comput. Inform. Sci.* **28**(4), 395–406 (2016)
- Banharsakun, A., Achalakul, T., Sirinaovakul, B.: The best-so-far selection in artificial bee colony algorithm. *Appl. Soft. Comput.* **11**(2), 2888–2901 (2011)
- Basturk, B., and D. Karaboga. (2006) An artificial bee colony (abc) algorithm for numeric function optimization. In: *IEEE Swarm Intelligence Symposium*. Indianapolis, Indiana, USA
- Basu, M.: Combined heat and power economic dispatch by using differential evolution. *Electr. Power Compon. Syst.* **38**, 996–1004 (2010)
- Basu, M.: Bee colony optimization for combined heat and power economic dispatch. *Expert Syst. Appl.* **38**, 13527–13531 (2011)
- Beigvand, S.D., Abdi, H., La Scala, M.: Combined heat and power economic dispatch problem using gravitational search algorithm. *Electr. Power Syst. Res.* **133**, 160–172 (2016)
- Chen, Tinggui, and Renbin Xiao. (2014) Enhancing artificial bee colony algorithm with self-adaptive searching strategy and artificial immune network operators for global optimization. *The Scientific World Journal* 2014.
- Civicioglu, P.: Backtracking search optimization algorithm for numerical optimization problems. *Appl. Math. Comput.* **219**(15), 8121–8144 (2013)
- Dorigo, M., Maniezzo, V., Coloni, A.: Ant system: optimization by a colony of cooperating agents. *IEEE Trans Syst, Man, Cybern, Part B Cybern* **26**(1), 29–41 (1996)
- Gao, W., Liu, S.: Improved artificial bee colony algorithm for global optimization. *Inf. Process. Lett.* **111**(17), 871–882 (2011)
- Gao, W.-F., Liu, S.-Y.: A modified artificial bee colony algorithm. *Comput. Oper. Res.* **39**(3), 687–697 (2012)
- Golberg, D.E.: *Genetic algorithms in search, optimization, and machine learning*. Addisonwesley **1989**, 102 (1989)
- Kang, F., Li, J., Qing, Xu.: Structural inverse analysis by hybrid simplex artificial bee colony algorithms. *Comput. Struct.* **87**(13), 861–870 (2009)
- Karaboga, N.: A new design method based on artificial bee colony algorithm for digital IIR filters. *J. Franklin Inst.* **346**(4), 328–348 (2009)
- Karaboga, D., Akay, B.: A survey: algorithms simulating bee swarm intelligence. *Artif. Intell. Rev.* **31**(1–4), 61–85 (2009)
- Karaboga, D., Akay, B.: A comparative study of artificial bee colony algorithm. *Appl Math Comput* **214**(1), 108–132 (2009)
- Karaboga, D., Gorkemli, B.: A quick artificial bee colony (qABC) algorithm and its performance on optimization problems. *Appl. Soft Comput.* **23**, 227–238 (2014)
- Karaboga, D., Gorkemli, B., Ozturk, C., Karaboga, N.: A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artif. Intell. Rev.* **42**(1), 21–57 (2014)

- Karaboga, D., Ozturk, C.: A novel clustering approach: artificial bee colony (ABC) algorithm. *Appl Soft Comput* **11**(1), 652–657 (2011)
- Karaboga, Dervis. (2005) An idea based on honey bee swarm for numerical optimization. Technical report-tr06, Erciyesuniversity, engineering faculty, computer engineering department. Vol. 200
- Kassabalidis I, El-Sharkawi MA, Marks RJ, Arabshahi P, Gray AA (2001) Swarm intelligence for routing in communication networks. IEEE Global Telecommunications Conference. GLOBE-COM'01. 6: 3613-3617
- Kaveh, A., Talatahari, S.: Size optimization of space trusses using Big Bang-Big Crunch algorithm. *Comput. Struct.* **87**(17), 1129–1140 (2009)
- Liu, B., Wang, L., Jin, Y.H., Tang, F., Huang, D.X.: Improved particle swarm optimization combined with chaos. *Chaos, Solitons Fractals* **25**(5), 1261–1271 (2005)
- Luo, J., Wang, Q., Xiao, X.: A modified artificial bee colony algorithm based on converge-onlookers approach for global optimization. *Appl. Math. Comput.* **219**(20), 10253–10262 (2013)
- Maeda, M., Tsuda, S.: Reduction of artificial bee colony algorithm for global optimization. *Neurocomput* **148**, 70–74 (2015)
- Malik, R.F., Rahman, T.A., Hashim, S.Z., Ngah, R.: New particle swarm optimizer with sigmoid increasing inertia weight. *Int J Comput Sci Security* **1**(2), 35–44 (2007)
- Mohammadi-Ivatloo, B., Moradi-Dalvand, M., Rabiee, A.: Combined heat and power economic dispatch problem solution using particle swarm optimization with time varying acceleration coefficients. *Electr. Power Syst. Res* **95**, 9–18 (2013)
- Neyestani, M., Hatami, M., Hesari, S.: Combined heat and power economic dispatch problem using advanced modified particle swarm optimization. *J. Renew. Sustain. Energy.* **11**(1), 015302 (2019)
- Ozturk, C., Hancer, E., Karaboga, D.: Dynamic clustering with improved binary artificial bee colony algorithm. *Appl. Soft Comput.* **28**, 69–80 (2015)
- Pawar, P., Rao, R., Davim, J.: Optimization of process parameters of milling process using particle swarm optimization and artificial bee colony algorithm. In: International Conference on Advances in Mechanical engineering (2018).
- Pan, Q.K., Tasgetiren, M.F., Suganthan, P.N., Chua, T.J.: A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. *Inform. Sci.* **181**(12), 2455–2468 (2011)
- Passino, K.M.: Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control. Syst.* **22**(3), 52–67 (2002)
- Rajasekhar, Anguluri, Ajith Abraham, and Millie Pant. (2011) Levy mutated artificial bee colony algorithm for global optimization. Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on. IEEE.
- Rao, R.S., Narasimham, S.V., Ramalingaraju, M.: Optimization of distribution network configuration for loss reduction using artificial bee colony algorithm. *Int J Electr Power Energy Syst Eng* **1**(2), 116–122 (2008)
- Samanta, S., Chakraborty, S.: Parametric optimization of some non-traditional machining processes using artificial bee colony algorithm. *Eng. Appl. Artif. Intell.* **24**(6), 946–957 (2011)
- dos Santos, C.L., Mariani, V.C.: A novel chaotic particle swarm optimization approach using Hénon map and implicit filtering local search for economic load dispatch. *Chaos, Solitons Fractals* **39**(2), 510–518 (2009)
- Singh, A.: An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem. *Appl. Soft Comput.* **9**(2), 625–631 (2009)
- Stanarevic, Nadezda, Milan Tuba, and Nebojsa Bacanin. (2010) Enhanced artificial bee colony algorithm performance. In: Proceedings of the 14th WSEAS international conference on computers: part of the 14th WSEAS CSCC multiconference. 2: 440-445
- Sun, L., Sun, W., Liang, X., He, M., Chen, H.: A modified surrogate-assisted multi-swarm artificial bee colony for complex numerical optimization problems. *Microprocess Microsyst* **76**, 103050 (2020)
- Wang, H., et al.: Multi-strategy ensemble artificial bee colony algorithm. *Inform Sci* **279**, 587–603 (2014)
- Xiang, T., Liao, X., Wong, K.-w.: An improved particle swarm optimization algorithm combined with piecewise linear chaotic map. *Appl. Math. Comput.* **190**(2), 1637–1645 (2007)
- Xiao, S., Wang, H., Huang, Z., Zhou, X., Xu, M.: Artificial bee colony algorithm based on adaptive neighborhood search and Gaussian perturbation. *Appl. Soft Comput.* **100**, 106955 (2021)
- Yan, X., et al.: A new approach for data clustering using hybrid artificial bee colony algorithm. *Neurocomput* **97**, 241–250 (2012)

- Yang, Xin-She. (2005) Engineering optimizations via nature-inspired virtual bee algorithms. International Work-Conference on the Interplay between Natural and Artificial Computation. Springer Berlin Heidelberg
- Yi Y, and He R (2014) A novel artificial bee colony algorithm. Intelligent human-machine systems and cybernetics (IHMSC), 2014 Sixth International Conference on I IEEE
- Yurtkuran, A., Emel, E.: An enhanced artificial bee colony algorithm with solution acceptance rule and probabilistic multisearch. *Comput. Intell. Neurosci.* **2016**, 41 (2016)
- Zhang D, Guan X, Tang Y, Tang Y. (2011) Modified artificial bee colony algorithms for numerical optimization. In: Proc. of 3rd International Workshop on Intelligent Systems and Applications.
- Zhu, G., Kwong, S.: Gbest-guided artificial bee colony algorithm for numerical function optimization. *Appl. Math. Comput.* **217**(7), 3166–3173 (2010)
- Özbakir, L., Baykasoğlu, A., Tapkan, P.: Bees algorithm for generalized assignment problem. *Appl Math Comput* **215**(11), 3782–3795 (2010)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Kalaipriyan Thirugnanasambandam¹ · M. Rajeswari² ·
Debnath Bhattacharyya³ · Jung-yoon Kim⁴

Kalaipriyan Thirugnanasambandam
kalaip27@gmail.com

M. Rajeswari
raji.rajeswari18@gmail.com

Debnath Bhattacharyya
debnathb@gmail.com

- ¹ School of Computer Science and Engineering, Vellore Institute of Technology, Chennai Campus, Chennai, India
- ² Sri Manakula Vinayagar Engineering College, Madhagadipet, Puducherry, India
- ³ Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vijayawada, Andhra Pradesh, India
- ⁴ Department of Game Media, College of Future Industry, Gachon University, Seongnam-si 13120, Gyeonggi-do, Republic of Korea